DA-IICT

CT215 LAB1

Name: Patel Raj Kamleshbhai

ID: 201901306

Date: 31 January 2021

EXRCISES

[A] Few Basic LabView Commands:

(1) sort(a, order): It sorts the matrix column wise. Here, "a" is a matrix or a vector. And "order" specifies the ascending/ descending order of sorting in which we want to sort the given vector. If we don't specify the order then by default sorting order is ascending order.

```
>>a = [ 3 2 4 1 0 1 9 9 5]

a =

     3     2     4     1     0     1     9     9     5


>>b = sort(a)

b =

     0     1     1     2     3     4     5     9     9


>>b = sort(a, 'descend')

b =

     9     9     5     4     3     2     1     1     0
```

(2) conv(a, b): It computes convolution of two matrices or vectors. And gives result of convolution theory of two signals. If "b" is not specified then it gives the auto-convolution i.e., conv(a, a).

```
>>a = [1 2 3 4]

a =

         1         2         3         4


>>b = [5 6 7 8]

b =

         5         6         7         8


>>c = conv(a, b)

c =

         5        16        34        60        61        52        32
```

(3) ones(a, b): It returns a matrix of ones of size a x b

zeros(a, b): it returns a matrix of zeros of size a x b.
(#Rows = a & #Columns = b)

```
>>a = ones(2, 3)

a =

         1         1         1
         1         1         1


>>b = ones(4, 3)

b =

         1         1         1
         1         1         1
         1         1         1
         1         1         1
```

(4) bitand(a, b): It computes bitwise AND operation between a and b.

bitor(a, b): It computes bitwise OR operation between a and b.

bitxor(a, b): It computes bitwise AND operation between a and b.
(All of these bitwise functions return respective answer in the decimal.)

```
>>a = bitand(6, 14)

a =

         6


>>b = bitor(9, 7)

b =

        15


>>c = bitxor(3, 8)

c =

        11
```

(5) cat(type, a, b): It concatenates two matrices or vectors a and b. If we want to concatenate more than two matrix then use cat(1,a,b,c, ……). Here, type "1" specifies column wise concatenation and type "2" specifies row wise concatenation.

```
>>a = [1 2 3 4]

a =

     1     2     3     4


>>b = [5 6 7 8]

b =

     5     6     7     8


>>c = cat(1, a, b)

c =

     1     2     3     4
     5     6     7     8


>>c = cat(2, a, b)

c =

     1     2     3     4     5     6     7     8
```

[B] Compute the Fourier transform (FT) of the signal g(t) = exp(−2t)u(t), where u(.) is the unit step function.

- $Fourier\ transform\ is\ given\ by,$

$$G(\omega) = \int_{-\infty}^{+\infty} g(t)e^{-j\omega t}dt$$

$$And\ g(t) = \ e^{-2t}u(t)\ is\ given,$$

$$So, G(\omega) = \int_{-\infty}^{+\infty} e^{-2t}u(t)e^{-j\omega t}dt$$

$$Here\ unit\ step\ function\ u(t) = \begin{cases} 0; & t\ < 0 \\ 1; & t\ \geq 0 \end{cases}$$

$$\therefore G(\omega) = \int_{0}^{+\infty} e^{-(2\ +j\omega)t}dt$$

$$\therefore G(\omega) = \frac{[e^{-(2\ +j\omega)t}]_0^{+\infty}}{-(2\ +j\omega)}$$

$$\therefore G(\omega) = \frac{[e^{-\infty}\ -\ e^{0}]}{-(2\ +j\omega)}$$

$$\boxed{\therefore G(\omega) = \frac{1}{2\ +j\omega}}$$

[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):

CODE:

```
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')
```

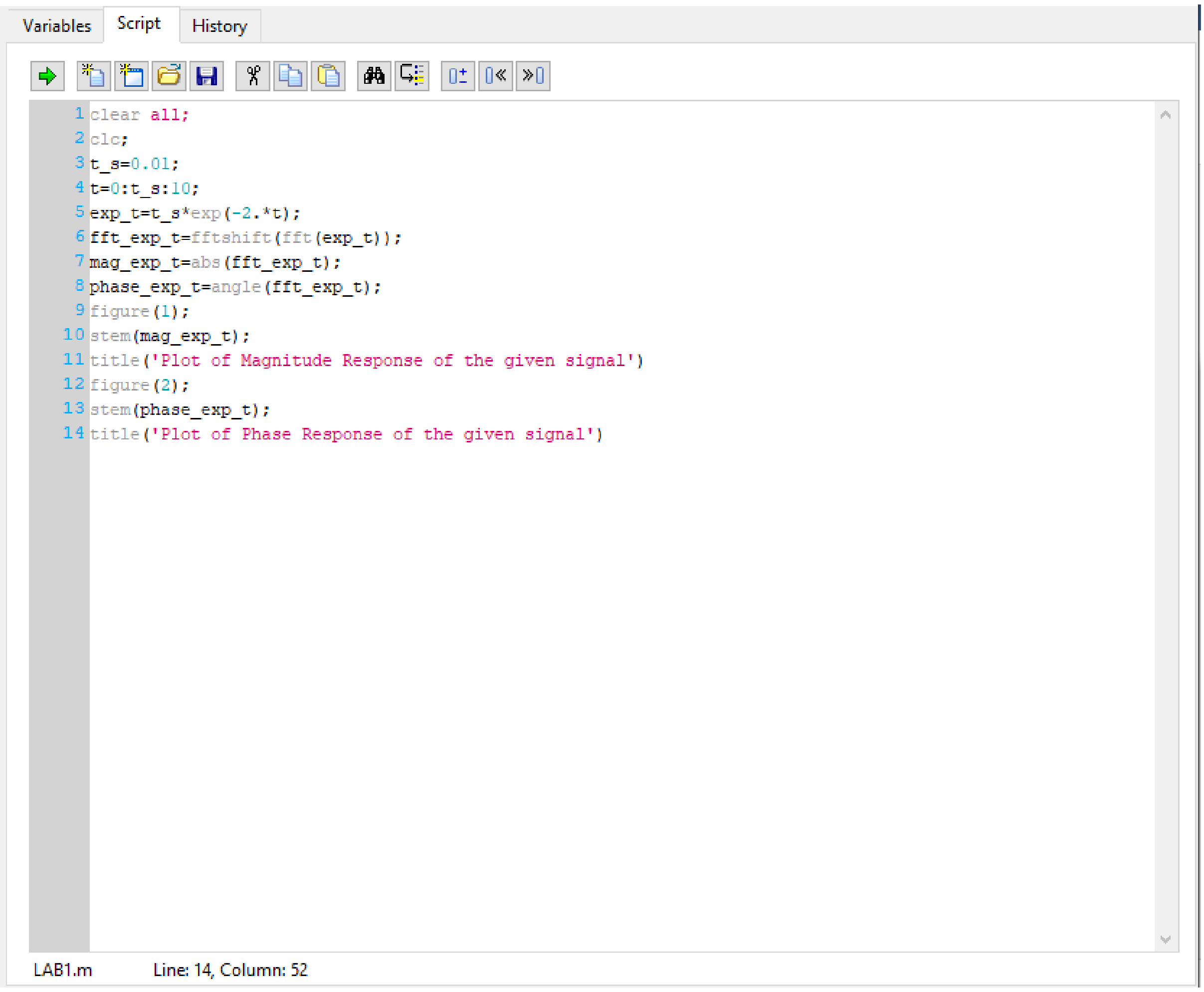
In the above code,

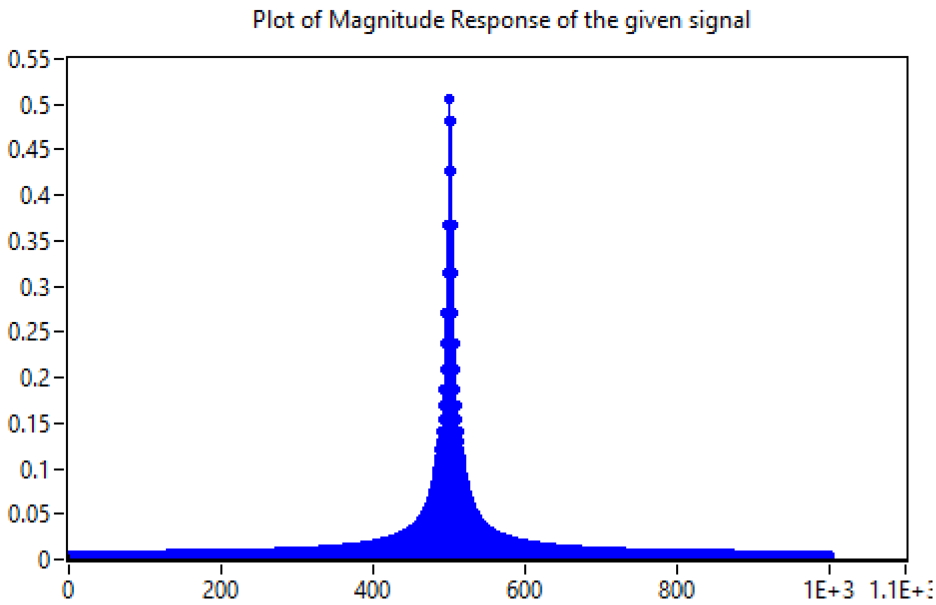
• t_s is the sampling interval, because g(t) is the continuous signal and it can not be stored. So, we have stored its samples. And t is the total time.

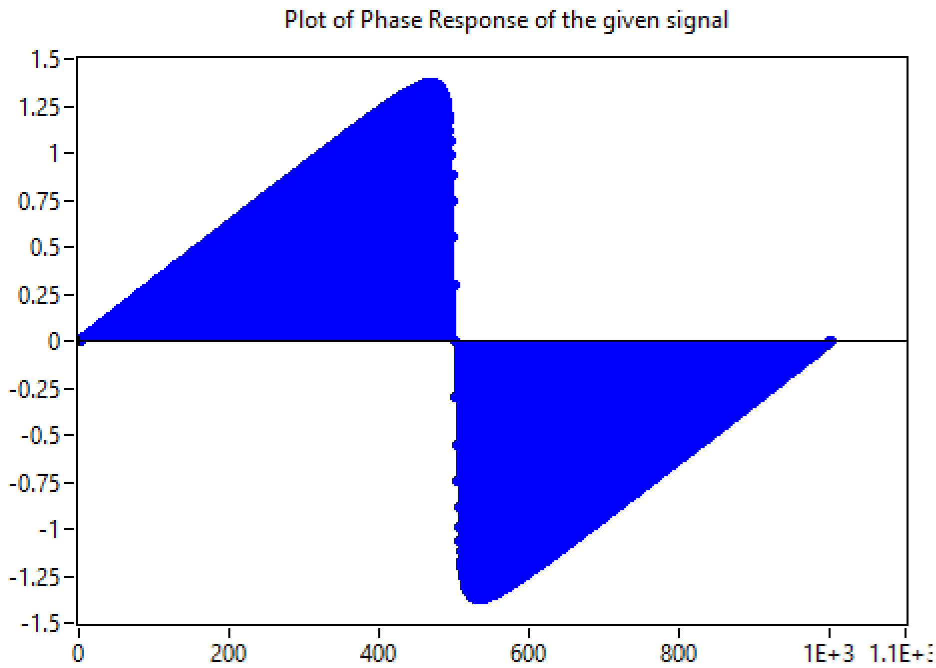
• fft() function (Fast Fourier Transform) computes the DFT (Discrete Fourier Transform) of the given signal g(t). And fftshift() function rearranges the array obtained from the fft() function such that 0 frequency sample is at the center of the array.

• abs() and angle() functions compute the magnitude and the phase respectively.

• figure(), stem() and title() functions are for plotting.







And from the above two plots,

• g(t) is real valued continuous time signal, as the magnitude response of g(t) is an even function and the phase response of g(t) is an odd function.