DSP

LAB # 02

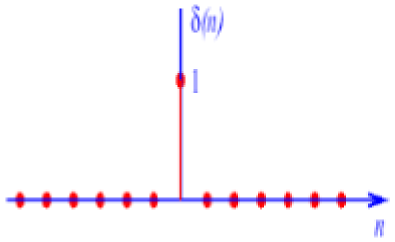
Generating Basic sequences in MATLAB

Lab Objective:

To generate the basic functions of the mathematics.

The Unit Delta (Impulse) function: is often called the discrete time impulse or the unit impulse. It is denoted by δ[n].

$$\delta[n] = \begin{cases} 1 & ,n = 0 \\ 0 & ,n \neq 0 \end{cases}$$



Discrete Shifted Unit Impulse:

$$\delta[n-k] = \begin{cases} 1 & ,n = k \\ 0 & ,n \neq k \end{cases}$$

Properties of Unit Impulse Function:

$$\delta[n] = u[n] - u[n-1]$$
$$u[n] = \sum_{k=-\infty}^{n} \delta[k]$$
$$x[n]\delta[n] = x[0]\delta[n]$$
$$x[n]\delta[n-k] = x[k]\delta[n-k]$$
$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

Creation of Unit Impulse and Unit Step sequences:
A unit impulse sequence I[n] of length N can be generated using the MATLAB command

I= [1 zeros (1, N-1)];

A unit impulse sequence Id of length N delayed by d samples, where d<N, can be generated using the MATLAB command

Id= [zeros (1, d) 1 zeros (1, N-d-1)];

Similarly, a unit step sequence S[n] of length N can be generated using the MATLAB command

S= [ones (1, N)]

A delayed unit step sequence can be generated in a way similar to that used in the generation of a delayed unit step sequence.

Program P1: can be used to generate and plot a unit impulse sequence between -5 and 10

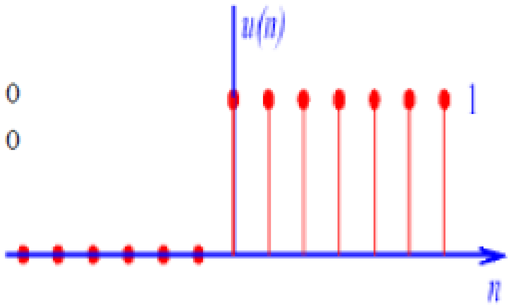
```
%Generate a vector from -5 to 10

clf; %clear old graph

n= -5:10

%Generate the unit impulse sequence

I= [zeros (1, 5) 1 zeros (1, 10)];

%Plot the unit impulse sequence

stem (n,I);

title (‘Unit Impulse Sequence’)

xlabel(‘Time index n’);

ylabel(‘Amplitude’)

axis([-5 10 0 2])
```

Lab Task 1:

- Modify the program P1 to generate a delayed and scaled unit impulse sequence Id[n] with a delay of 7 samples and scaled by a factor of 3. Run the modified program and display the sequence generated.
- Modify the program P1 to generate a unit step sequence S[n]. Run the modified program and display the sequence generated.
- Modify the program P1 to generate a unit step sequence Sd[n] with an advance of 3 samples. Run the modified program and display the sequence generated.

The Unit step function: The unit step, denoted by u(n), is defined by

$$\mathbf{U(n)} = \begin{cases} 1 & ,n \geq 0 \\ 0 & ,n < 0 \end{cases}$$

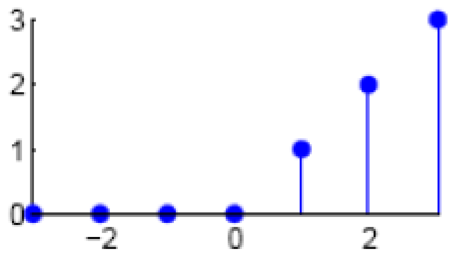


and is related to the unit sample by

$$u(n) = \sum_{k=-\infty}^{n} \delta(k)$$

The Unit Ramp Function:

$$r(n) = \begin{cases} n & 0 \leq n \leq \infty \\ 0 & \text{elsewhere} \end{cases}$$

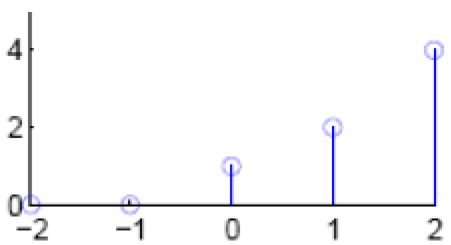


Lab Task 2:

Write a Matlab code that generates Unit Rampfunction

The Exponential Function: Finally, an *exponential* sequence is defined by

$$\text{Exponential}(n) = \begin{cases} a^n u(n) & -\infty \leq n \leq \infty \\ 0 & \text{elsewhere} \end{cases}$$



Creation of Real Exponentials

Another basic discrete-time sequence is the exponential sequence. Such a sequence can be generated using the MATLAB

Program P2: given below can be used to generate a real-valued exponential sequence.
%Generation of a real exponential sequence

```
clf;
n=0:20;
a=2;
k=0.2;
x=k*a.^n;
stem(n,x);
title('Sinusoidal Sequence');
xlabel('Time index n');
ylabel('Amplitude');
```

Lab Task 3:

- Which parameter controls the rate of growth or decay of this sequence and which parameter controls the amplitude of this sequence? Run program P2 again with parameter 'k' changed to 20.
- What will happen if the parameter a is less than 1? Run Program P2 again with parameter 'a' changed to 0.9.

Creation of Complex Exponentials:

Program P3: below can be used to generate a complex valued-exponential sequence.

```
%Generation of a complex exponential sequence
clf;
n=0:20
w=pi/6;
x=exp (j*w*n)
subplot(2,1,1);
stem(n,real(x));
xlabel('Time index n');
ylabel('Amplitude');
title('Real part');
subplot (2,1,2);
stem (n,imag(x));
xlabel('Time index n');
ylabel('Amplitude');
title ('Imaginary part');
```

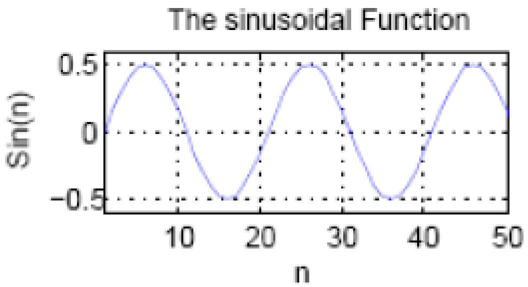
Lab Task 4:

- Create x[n] = exp (j*n/3) for some range and see its real and imaginary parts separately
- Make the following signals; plot their real and imaginary parts on the same figure. What difference do you notice between the two signals?
  - $X1[n]=e^{(-0.1+j0.3)n}$ $-10 \le n \le 10$
  - $X2[n]= e^{(0.1+j0.3)n}$ $-10 \le n \le 10$

The sinusoidal Function:

$$Y(n) = A \sin(W_c n + \varphi)$$

A=Amplitude, W_c= Angular frequency, phi (φ) = phase shift
$W_c = 2\pi f$ where f =frequency
f =1/T where T = Time Period.



Program P4: is a simple example that generates a sinusoidal signal

```
%Generation of a sinusoidal sequence

n=0:40; %length of sequence

f=0.1; %Frequency

phase = 0;

A=1.5; %Amplitude

x=A*cos (2*pi*f*n-phase);

stem(n,x);

title ('Sinusoidal Sequence')

xlabel ('Time index n');

ylabel ('Amplitude')

axis ([0 40 -2 2]);

grid;
```

Lab Task 5:

- Modify the program P4 to generate a sinusoidal sequence of length 50, frequency 0.08, amplitude 2.5 and phase shift 90 degrees and display it. Compare it with sequence of P4.

- Generate the following signals

$$x[n] = \sin\left(\frac{\pi}{17} n\right), \qquad 0 \le n \le 34$$

$$x[n] = \sin\left(\frac{\pi}{17} n\right), \qquad -17 \le n \le 17$$

Lab Task 6:

- Write a program in Matlab to generate Saw tooth Waveform.

Lab Task 7:

- Write a program in Matlab to generate a Triangular Pulse