GENERATING UNIT IMPULSE AND UNIT STEP SEQUENCES

LAB # 07

CSE301L Signals & Systems Lab

Submitted by: **Shah Raza**

Registration No: **18PWCSE1658**

Class Section: **B**

"On my honor, as a student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: ______________

Submitted to: **Engr. Durr-e-Nayab**

Wednesday, July 15th, 2020

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Lab Objectives:

Objectives of this lab are as follows:

- Generating unit impulse and unit step sequences.
- Basic signal operations.

Task # 1:

Using ones function, plot the signum sequence over interval -10≤n≤10. It can be defined as:

$$sign(n) = \begin{cases} 1, & for\, n > 0 \\ -1, & for\, n < 0 \\ 0, & for\, n = 0 \end{cases}$$

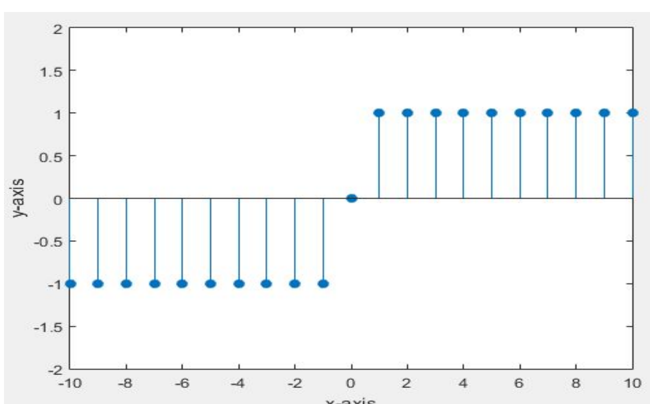
Problem Analysis:

Use ones and zeros functions to obtain the given signal.

Code:

```
n = -10:10;
x1=[-ones(1,10) zeros(1,1) ones(1,10)];
stem(n,x1,'filled');
axis([-10 10 -2 2]);
```

Output:



Task # 2:

Prove the following:

$$\delta[n] = u[n] - u[n-1]$$

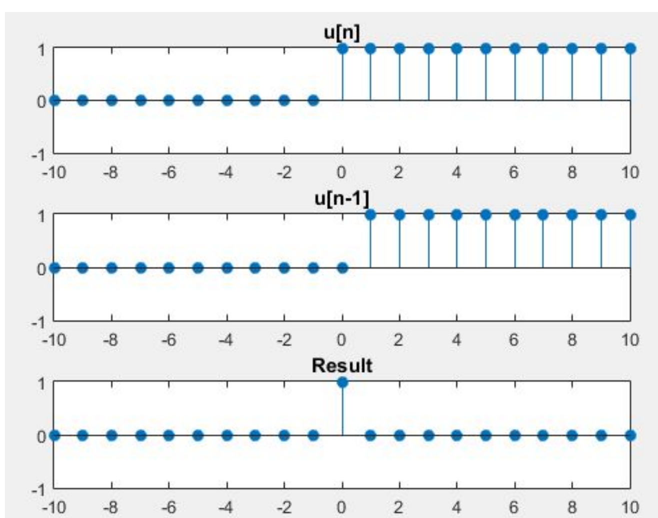
Problem Analysis:

To prove the given equation first generate a unit step signal and then subtract a shifted unit step signal from it.

Code:

```
n=-10:10;
u0=[zeros(1,10) ones(1,11)];
subplot(3,1,1);
stem(n,u0,'filled');
axis([-10 10 -1 1]);
title('u[n]');
u1=[zeros(1,11) ones(1,10)];
subplot(3,1,2);
stem(n,u1,'filled');
axis([-10 10 -1 1]);
title('u[n-1]');
u=u0-u1;
subplot(3,1,3);
stem(n,u,'filled');
axis([-10 10 -1 1]);
title('Result');
```

Output:



Task # 3:

Delay the original signal given in above example by 1 sec. Plot both the delayed & original signal on the same figure.

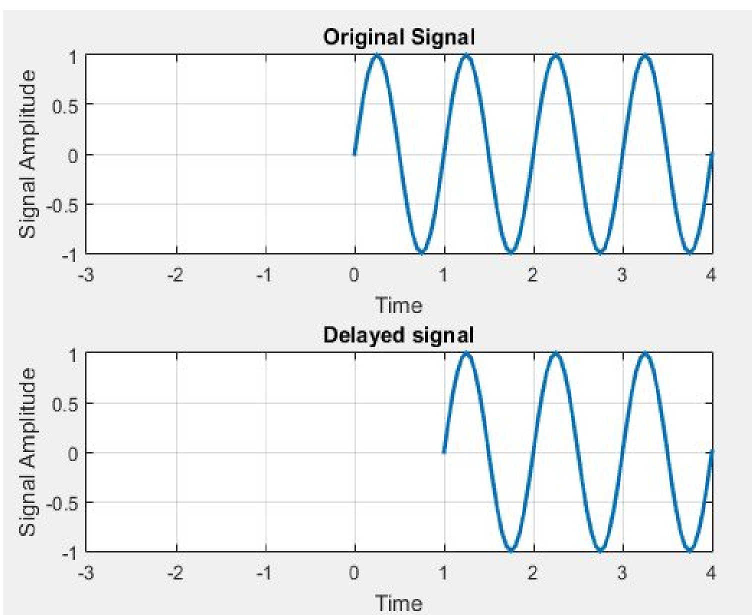
Problem Analysis:

To delay the signal by one unit, add one to the x-axis of the signal.

Code:

```
n=0:0.05:4;
x=sin(2*pi*1*n);
subplot(2,1,1);
plot(n,x,'linewidth',2);
title('Original Signal');
xlabel('Time');
ylabel('Signal Amplitude');
axis([-3 4 -1 1]);
grid;
subplot(2,1,2);
plot(n+1,x,'linewidth',2);
title('Delayed signal');
xlabel('Time');
ylabel('Signal Amplitude');
axis([-3 4 -1 1]);
grid;
```

Output:



Task # 4:

Flip the following signal:

$$y = 5\exp\left(i * n * \frac{\pi}{4}\right)$$

Plot the original signal as well as the flipped one in the same figure.

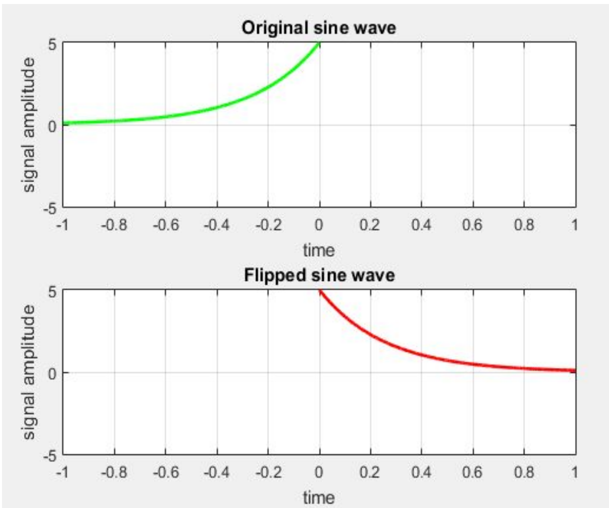
Problem Analysis:

To flip the given signal, multiply the x-axis of the signal to by minus.

Code:

```
n=-1:1/2000:1;
x1=5*exp(5*n*pi/4);
subplot(2,1,1);
plot(n,x1, 'g', 'linewidth',2);
axis([-1 1 -5 5]);
xlabel('time');
ylabel('signal amplitude');
title('Original sine wave');
grid;
subplot(2,1,2);
plot(-n,x1, 'r', 'linewidth',2);
axis([-1 1 -5 5]);
xlabel('time');
ylabel('signal amplitude');
title('Flipped sine wave');
grid;
```

Output:



Task # 5:

Flip the following signal:

$$x[n]= 2\,\delta[n]+ 5\,\delta[n-1] + 8\,\delta[n-2] + 4\,\delta[n-3] + 3\,\delta[n-4]$$

Plot the original signal as well as the flipped one in the same figure.

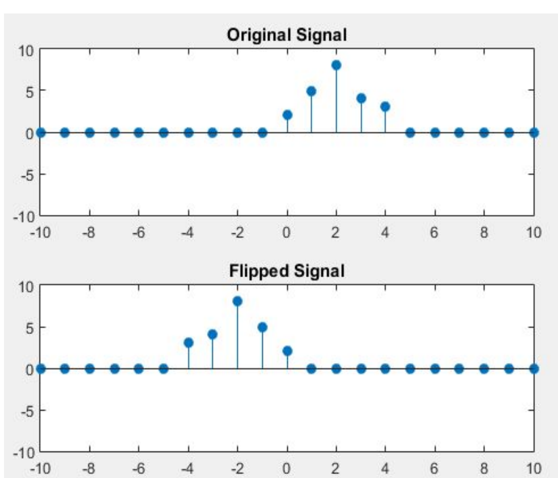
Problem Analysis:

Generate the signal as described above and then flip it.

Code:

```
n=-10:10;
x=[zeros(1,10) ones(1,1) zeros(1,10)];
x0=2*x;
x=[zeros(1,11) ones(1,1) zeros(1,9)];
x1=5*x;
x=[zeros(1,12) ones(1,1) zeros(1,8)];
x2=8*x;
x=[zeros(1,13) ones(1,1) zeros(1,7)];
x3=4*x;
x=[zeros(1,14) ones(1,1) zeros(1,6)];
x4=3*x;
xr=x0+x1+x2+x3+x4;
subplot(2,1,1);
stem(n,xr,'filled');
axis([-10 10 -10 10]);
title('Original Signal');
subplot(2,1,2);
stem(-n,xr,'filled');
axis([-10 10 -10 10]);
title('Flipped Signal');
```

Output:



Task # 6:

Scale the continuous-time sinusoid used in the signal shifting example by a factor of 2.

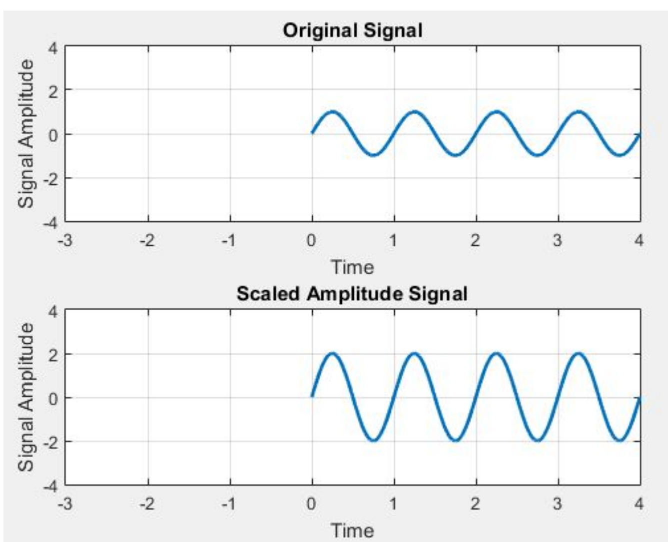
Problem Analysis:

To scale the amplitude of the signal, multiply the y-axis of the signal by the given factor.

Code:

```
n=0:0.002:4;
x=sin(2*pi*1*n);
subplot(2,1,1);
plot(n,x,'linewidth',2);
title('Original Signal');
xlabel('Time');
ylabel('Signal Amplitude');
axis([-3 4 -4 4]);
grid;
subplot(2,1,2);
S=2;
plot(n,S*x,'linewidth',2);
title('Scaled Amplitude Signal');
xlabel('Time');
ylabel('Signal Amplitude');
axis([-3 4 -4 4]);
grid;
```

Output:



Task # 7:

Interpolate (up-sample) the signal by a factor of 2.

Problem Analysis:

To interpolate the given signal use interp command.

Code:

```
n=-2:1/1000:2;
x1=sin(2*pi*2*n);
x2=interp(x1,2);
subplot(2,1,1);
plot(x1);
title('Original signal');
xlabel('Sample Number');
ylabel('Signal Amplitude');
axis([0 4000 -1 1]);
grid;
subplot(2,1,2);
plot(x2);
title('Interpolated Signal');
xlabel('Sample Number');
ylabel('Signal Amplitude');
axis([0 2000 -1 1]);
grid;
```

Output: