

Signals and Systems

Lab Manual

CNET-221



Department of Computer Engineering & Networks

COLLEGE OF COMPUTER SCIENCE & INFORMATION SYSTEMS
JAZAN UNIVERSITY
KINGDOM OF SAUDI ARABIA

Mr. Riad M. I. Khabakho
Mr. Asif Ali Khan
Department of Computer Networks and Engineering
2014-2015

LIST OF EXPERIMENTS

No.	Name of Experiment / Program
Section-I	
1	Simulation Program for a Discrete-Time signal
2.	Simulation Program for a Continuous-Time Signal
3	Simulation program for a Periodic Signal
4	Simulation program for a Aperiodic Signal
5	Simulation program for Time Shifting-I
6	Simulation program for Time Shifting-II
7	Simulation program for Time Reversal
8	Simulation program for Time Scaling
9	Simulation program for the Unit Impulse Signal
10	Simulation program for the Unit Step Sequins Signal
11	Simulation program for the Unit Step Function-I
12	Simulation program for the Unit Step Function-II
13	Simulation program for the Ramp Sequence Signal
Section-II	
14	Introduction to the Function generator and Oscilloscope (Hardware)
15	Waveform generation and there analysis using Fourier Transform (Hardware)
16	Frequency Manipulation (Hardware)

Note: The lab has been broadly divided into two sections. In the first section, simulation has to be done using MATLAB. Another section (marked as Hardware) has to be performed using Oscilloscope and other required hardware units. The final result is to be recorded from the CRO.

MATLAB is a flexible, extensible software package that allows for the quick, efficient manipulation and visualization of datasets, among its many uses. It is widely used in the oceanographic community, and will serve you well regardless of what field you go into, when it comes to dealing with data.

General Purpose Commands

Operators and Special Characters	
+	Plus; addition operator.
-	Minus; subtraction operator.
*	Scalar and matrix multiplication operator.
.*	Array multiplication operator.
^	Scalar and matrix exponentiation operator.
.^	Array exponentiation operator.
\	Left-division operator.
/	Right-division operator.
.\	Array left-division operator.
./	Array right-division operator.
:	Colon; generates regularly spaced elements and represents an entire row or column.
()	Parentheses; encloses function arguments and array indices; overrides precedence.
[]	Brackets; enclosures array elements.
.	Decimal point.
...	Ellipsis; line-continuation operator.
,	Comma; separates statements and elements in a row.
;	Semicolon; separates columns and suppresses display.
%	Percent sign; designates a comment and specifies formatting.
'	Quote sign and transpose operator.
.'	Nonconjugated transpose operator.
=	Assignment (replacement) operator.

Commands for Managing a Session

clc	Clears Command window.
clear	Removes variables from memory.
exist	Checks for existence of file or variable.
global	Declares variables to be global.
help	Searches for a help topic.
lookfor	Searches help entries for a keyword.
quit	Stops MATLAB.
who	Lists current variables.
whos	Lists current variables (long display).

Special Variables and Constants

ans	Most recent answer.
eps	Accuracy of floating-point precision.
i, j	The imaginary unit $\sqrt{-1}$.
Inf	Infinity.
NaN	Undefined numerical result (not a number).
pi	The number π .

System and File Commands

cd	Changes current directory.
date	Displays current date.
delete	Deletes a file.
diary	Switches on/off diary file recording.
dir	Lists all files in current directory.
load	Loads workspace variables from a file.
path	Displays search path.
pwd	Displays current directory.
save	Saves workspace variables in a file.
type	Displays contents of a file.
what	Lists all MATLAB files in the current directory.
wklread	Reads .wkl spreadsheet file.

Input/Output Commands	
disp	Displays contents of an array or string.
fscanf	Read formatted data from a file.
format	Controls screen-display format.
fprintf	Performs formatted writes to screen or file.
input	Displays prompts and waits for input.
;	Suppresses screen printing.

Plotting commands

Basic xy Plotting Commands	
axis	Sets axis limits.
fplot	Intelligent plotting of functions.
grid	Displays gridlines.
plot	Generates xy plot.
print	Prints plot or saves plot to a file
title	Puts text at top of plot.
xlabel	Adds text label to x-axis.
ylabel	Adds text label to y-axis.

Plot Enhancement Commands	
axes	Creates axes objects.
close	Closes the current plot.
close all	Closes all plots.
figure	Opens a new figure window.
gtext	Enables label placement by mouse.
hold	Freezes current plot.
legend	Legend placement by mouse.
refresh	Redraws current figure window.
set	Specifies properties of objects such as axes.
subplot	Creates plots in subwindows.
text	Places string in figure.

Specialized Plot Commands

<code>bar</code>	Creates bar chart.
<code>loglog</code>	Creates log-log plot.
<code>polar</code>	Creates polar plot.
<code>semilogx</code>	Creates semilog plot (logarithmic abscissa).
<code>semilogy</code>	Creates semilog plot (logarithmic ordinate).
<code>stairs</code>	Creates stairs plot.
<code>stem</code>	Creates stem plot.

Programming

Logical and Relational Operators

<code>==</code>	Relational operator: equal to.
<code>~=</code>	Relational operator: not equal to.
<code><</code>	Relational operator: less than.
<code><=</code>	Relational operator: less than or equal to.
<code>></code>	Relational operator: greater than.
<code>>=</code>	Relational operator: greater than or equal to.
<code>&</code>	Logical operator: AND.
<code> </code>	Logical operator: OR.
<code>~</code>	Logical operator: NOT.
<code>xor</code>	Logical operator: EXCLUSIVE OR.

Mathematical Functions

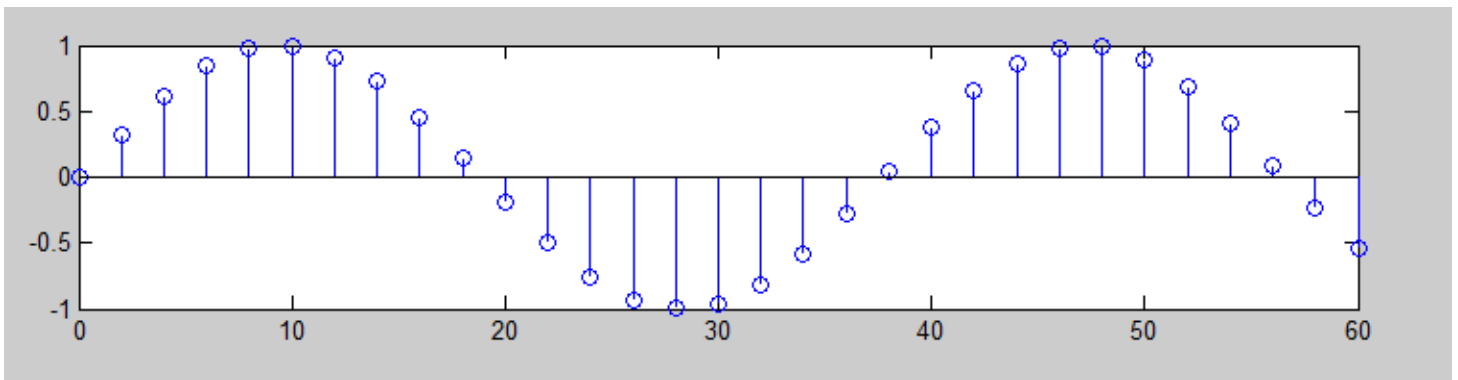
Exponential and Logarithmic Functions	
$\exp(x)$	Exponential; e^x .
$\log(x)$	Natural logarithm; $\ln(x)$.
$\log_{10}(x)$	Common (base 10) logarithm; $\log(x) = \log_{10}(x)$.
$\text{sqrt}(x)$	Square root; \sqrt{x} .

Trigonometric Functions	
$\text{acos}(x)$	Inverse cosine; $\arccos x = \cos^{-1}(x)$.
$\text{acot}(x)$	Inverse cotangent; $\text{arccot } x = \cot^{-1}(x)$.
$\text{acsc}(x)$	Inverse cosecant; $\text{arcs } x = \csc^{-1}(x)$.
$\text{asec}(x)$	Inverse secant; $\text{arcsec } x = \sec^{-1}(x)$.
$\text{asin}(x)$	Inverse sine; $\arcsin x = \sin^{-1}(x)$.
$\text{atan}(x)$	Inverse tangent; $\arctan x = \tan^{-1}(x)$.
$\text{atan2}(y, x)$	Four-quadrant inverse tangent.
$\cos(x)$	Cosine; $\cos(x)$.
$\cot(x)$	Cotangent; $\cot(x)$.
$\csc(x)$	Cosecant; $\csc(x)$.
$\sec(x)$	Secant; $\sec(x)$.
$\sin(x)$	Sine; $\sin(x)$.
$\tan(x)$	Tangent; $\tan(x)$.

1- Discrete-Time signal

Write a program to generate a discrete sinusoidal signal and compare the generated output.

```
n = 0:2:60;  
y = sin(n/6);  
subplot(3,1,1)  
stem(n,y)
```

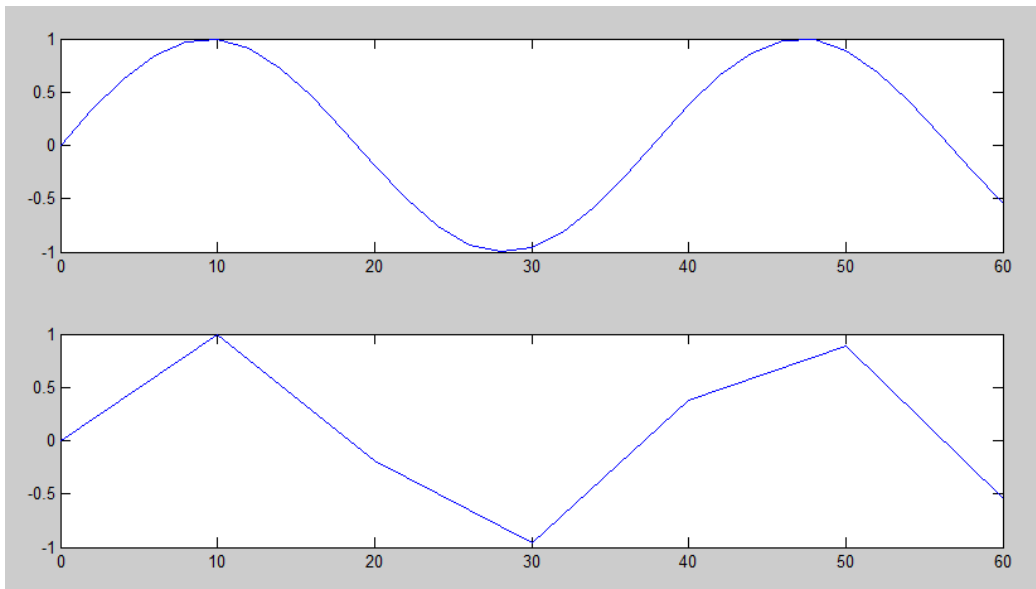


This plot shows the **discrete-time signal** formed by computing the values of the function $\sin(t/6)$ at points which are uniformly spaced at intervals of size 2. Notice that while $\sin(t/6)$ is a continuous-time function, the sampled version of the signal, $\sin(n/6)$, is a discrete-time function.

2- Continuous-Time Signal

Write a program to generate a Continuous sinusoidal signal and compare the generated output.
Verify the result for various grid values

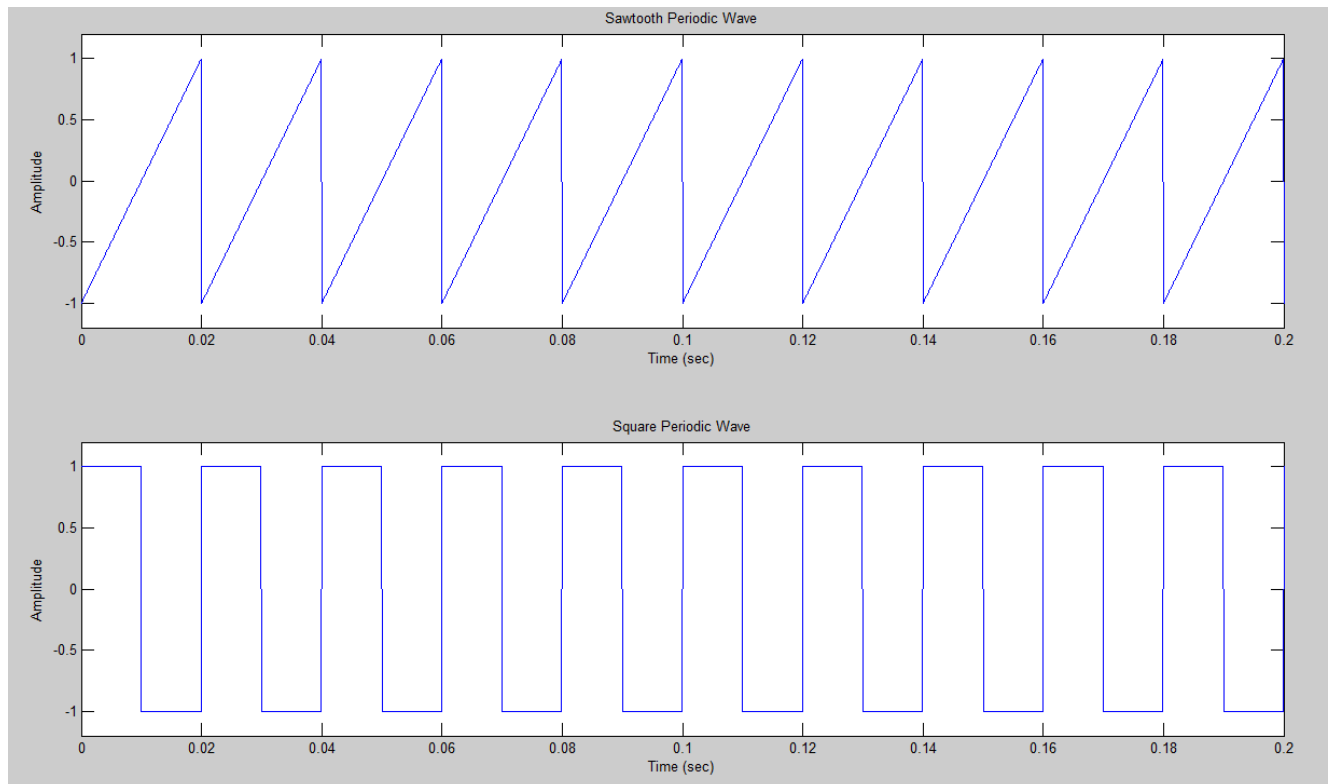
```
n1 = 0:2:60;  
z = sin(n1/6);  
subplot(3,1,2)  
plot(n1,z)  
n2 = 0:10:60;  
w = sin(n2/6);  
subplot(3,1,3)  
plot(n2,w)
```



3- Periodic Signal

Write a program to generate periodic saw tooth and square wave and compare the generated output.

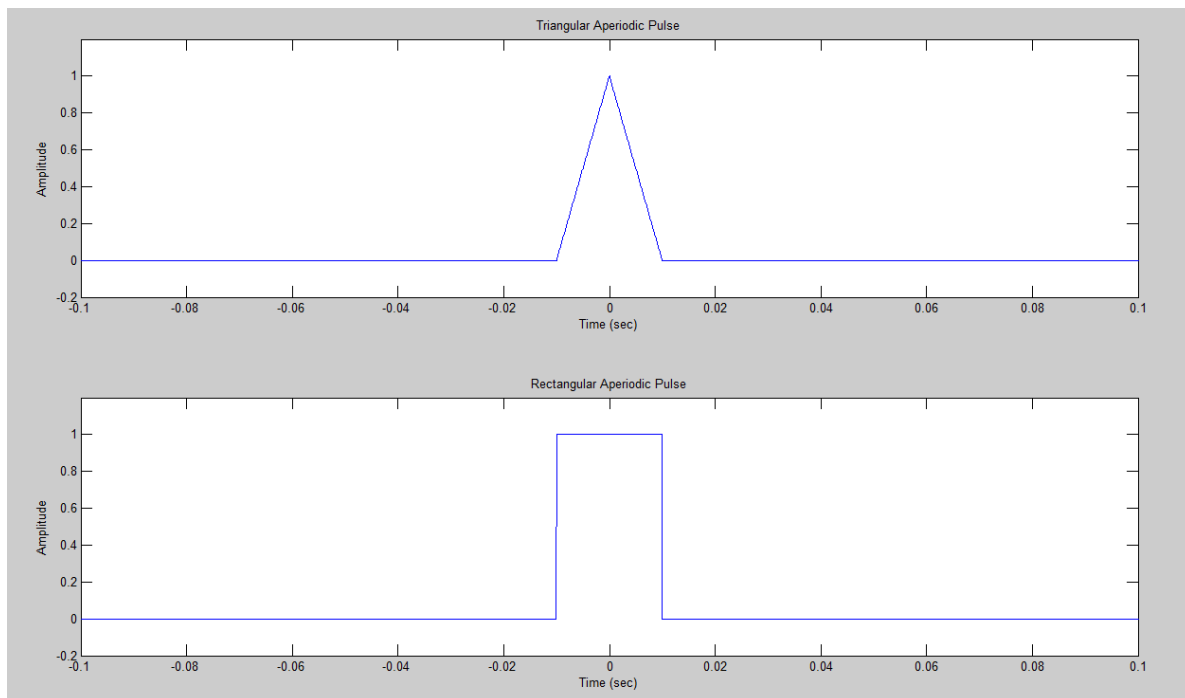
```
fs = 10000;  
t = 0:1/fs:1.5;  
x1 = sawtooth(2*pi*50*t);  
x2 = square(2*pi*50*t);  
subplot(211),plot(t,x1), axis([0 0.2 -1.2 1.2])  
xlabel('Time (sec)');  
ylabel('Amplitude');  
title('Sawtooth Periodic Wave');  
subplot(212)  
plot(t,x2)  
axis([0 0.2 -1.2 1.2]);  
xlabel('Time (sec)');  
ylabel('Amplitude');  
title('Square Periodic Wave');
```



4- Aperiodic Signal

Write a program to generate aperiodic Triangular and Rectangular wave and compare the generated output.

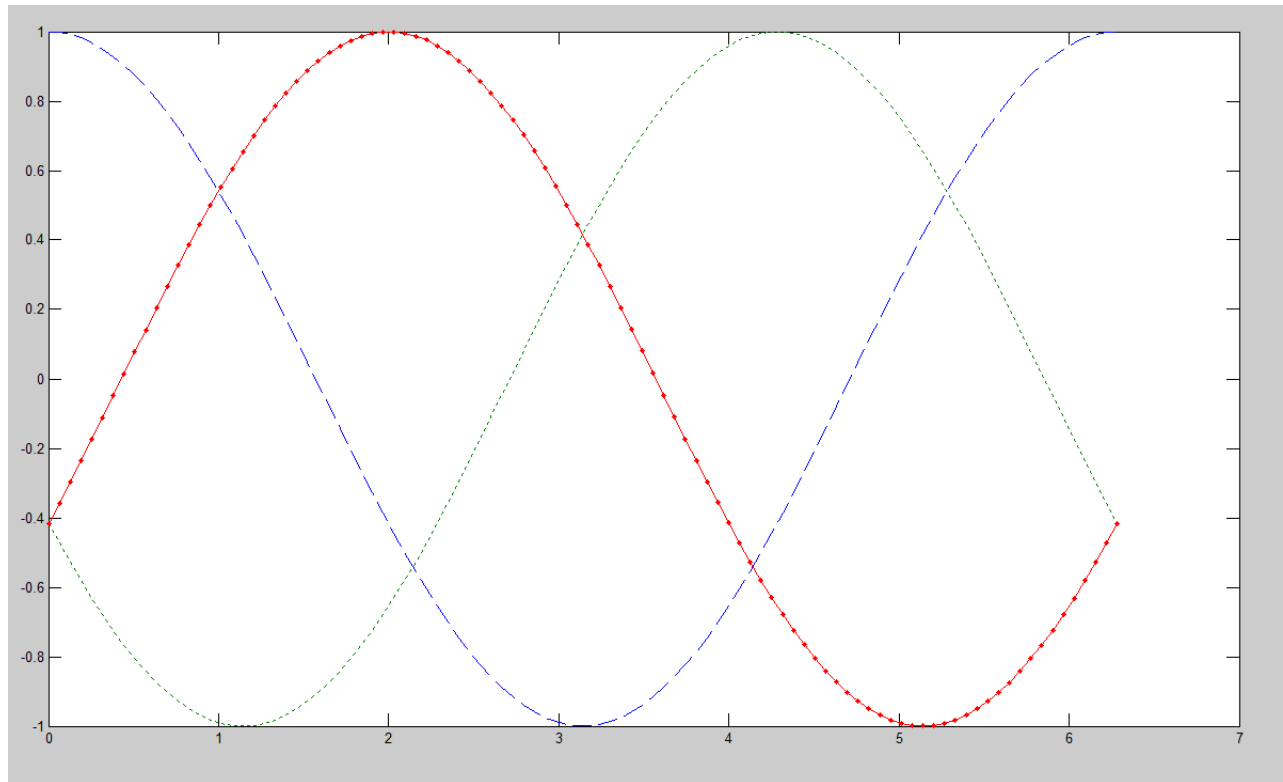
```
fs = 10000;
t = -1:1/fs:1;
x1 = tripuls(t,20e-3);
x2 = rectpuls(t,20e-3);
subplot(211)
plot(t,x1)
axis([-0.1 0.1 -0.2 1.2])
xlabel('Time (sec)');
ylabel('Amplitude');
title('Triangular Aperiodic Pulse');
subplot(212)
plot(t,x2)
axis([-0.1 0.1 -0.2 1.2])
xlabel('Time (sec)');
ylabel('Amplitude');
title('Rectangular Aperiodic Pulse');
h(gcf) =(gcf);
h(gcf).Color = [1,1,1];
```



5- Time Shifting-I

Write a program to generate a periodic signal, and perform positive and negative shift and compare the generated output.

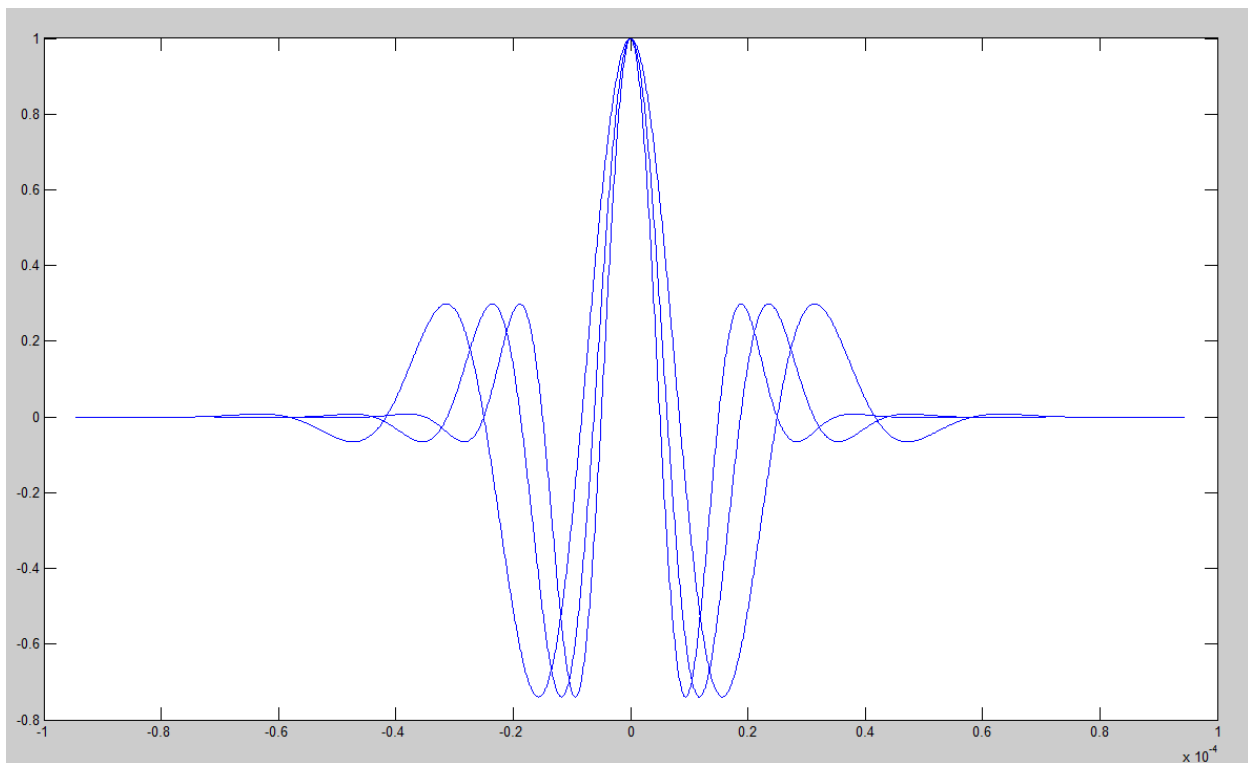
```
x = linspace( 0, 2*pi, 100);  
y1 = cos(x);  
y2 = cos(x+2);  
y3 = cos(x-2);  
plot(x, y1, '--',x,y2,':',x, y3,'-')
```



6- Time Shifting-II

Write a program to verify the time shifting and compare the generated output.

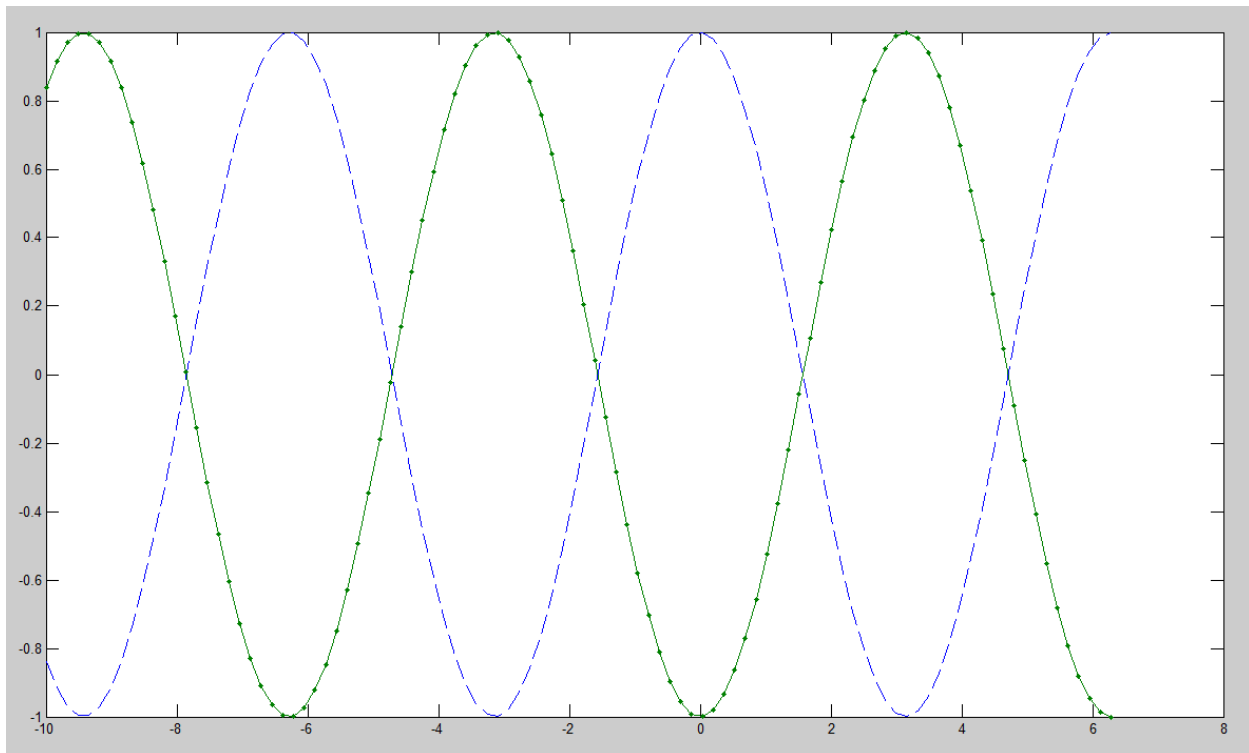
```
clear all; close all; clc;
fc=30e3;
for n=1:3;
tc = gauspuls('cutoff',fc,0.4,[],-40);
t = -tc : 1e-7 : tc;
yi = gauspuls(t,fc,0.6);
plot(t,yi)
hold on
fc=fc+10e3;
end
```



7- Time Reversal

Write a program to generate sinusoidal wave, it's time reversal and compare the generated output.

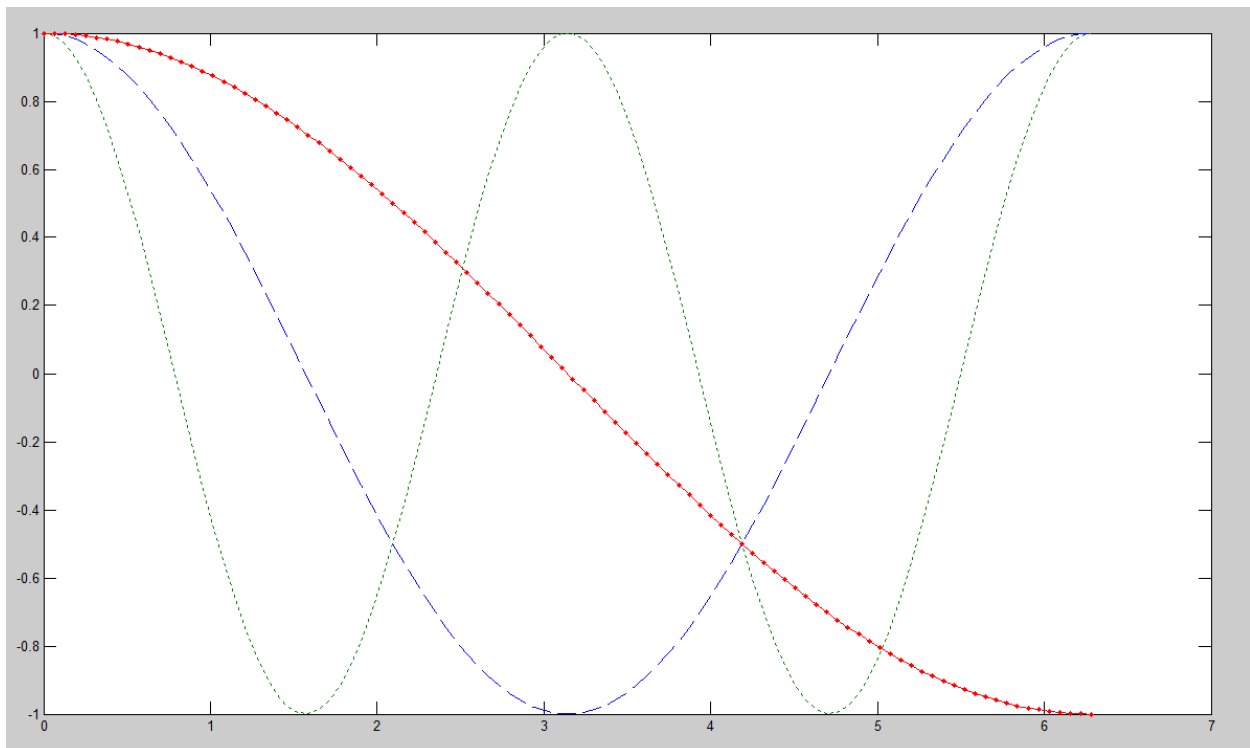
```
x = linspace(-10, 2*pi, 100);  
y1 = cos(x);  
y2 = -cos(x);  
plot(x, y1, '-o', x, y2, '-o')
```



8- Time Scaling

Write a program to generate a sinusoidal signal, time scaling and compare the generated output.

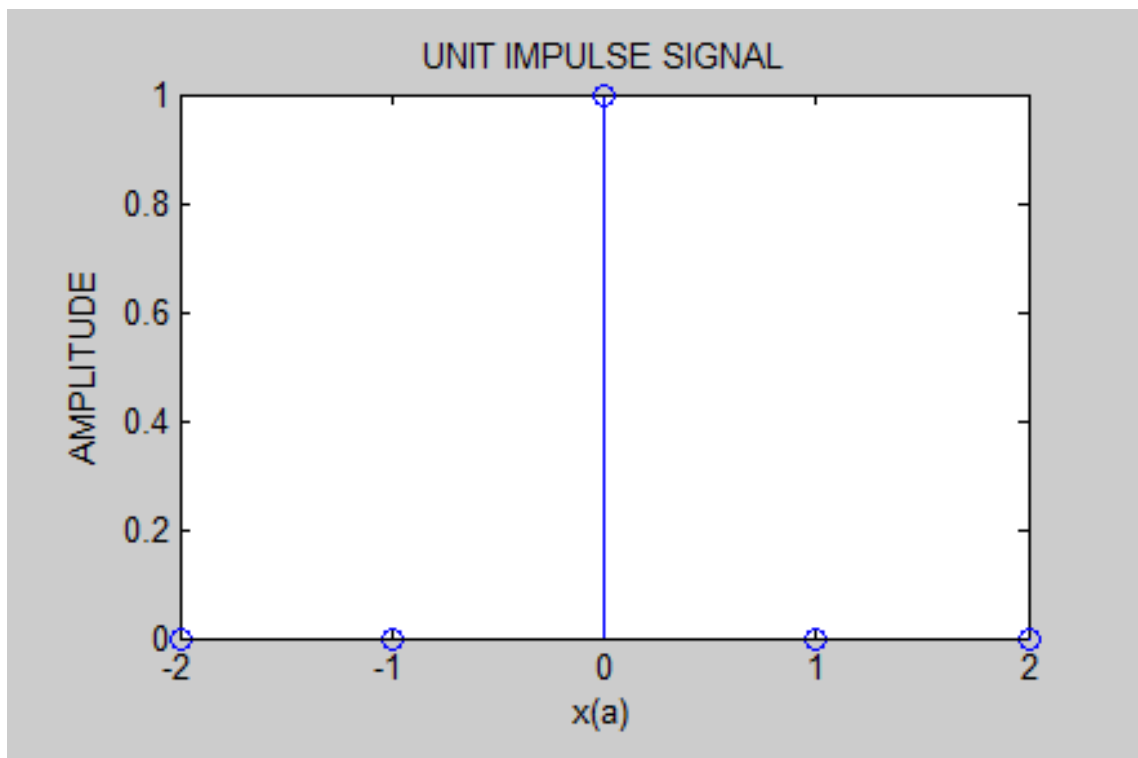
```
x = linspace( 0, 2*pi, 100);  
y1 = cos(x);  
y2 = cos(2*x);  
y3 = cos(x/2);  
plot(x, y1, '--',x,y2,':',x, y3,'-')
```



9- Unit Impulse Signal

Write a program to generate a unit impulse signal and compare the generated output

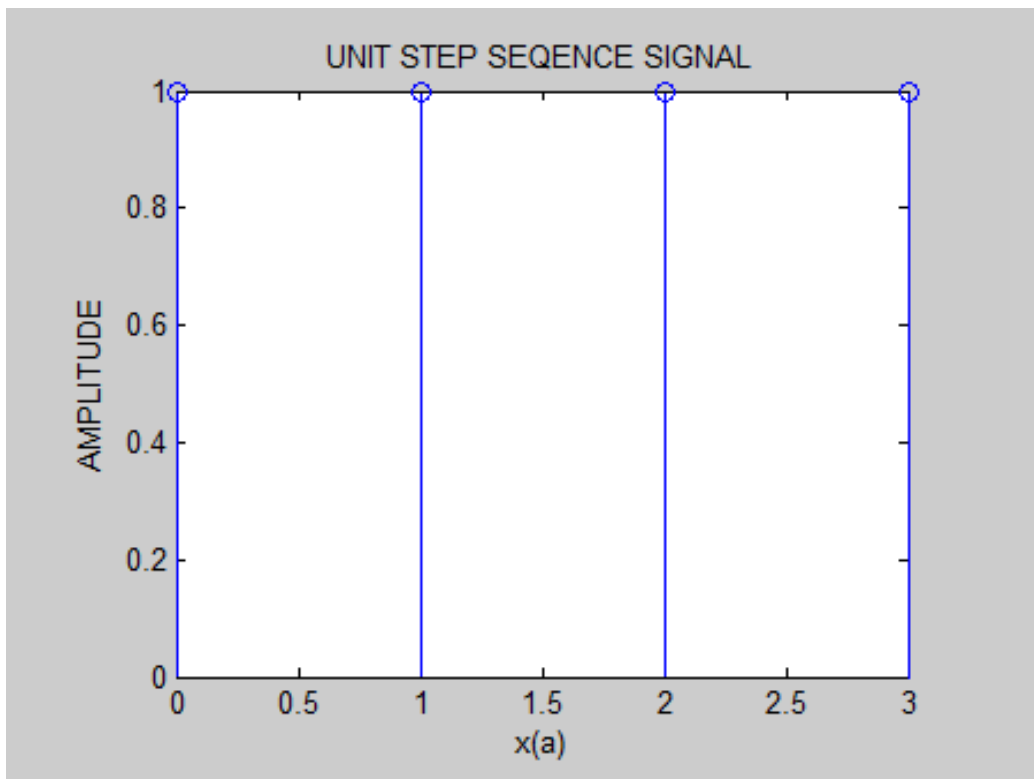
```
t=-2:2;  
y=[zeros(1,2),ones(1,1),zeros(1,2)];  
subplot(2,2,1);  
stem(t,y);  
title('UNIT IMPULSE SIGNAL');  
ylabel('AMPLITUDE');  
xlabel('x(a)');  
print
```



10- UNIT STEP SEQUENCE SIGNAL

Write a program to generate a discrete unit step signal and compare the generated output

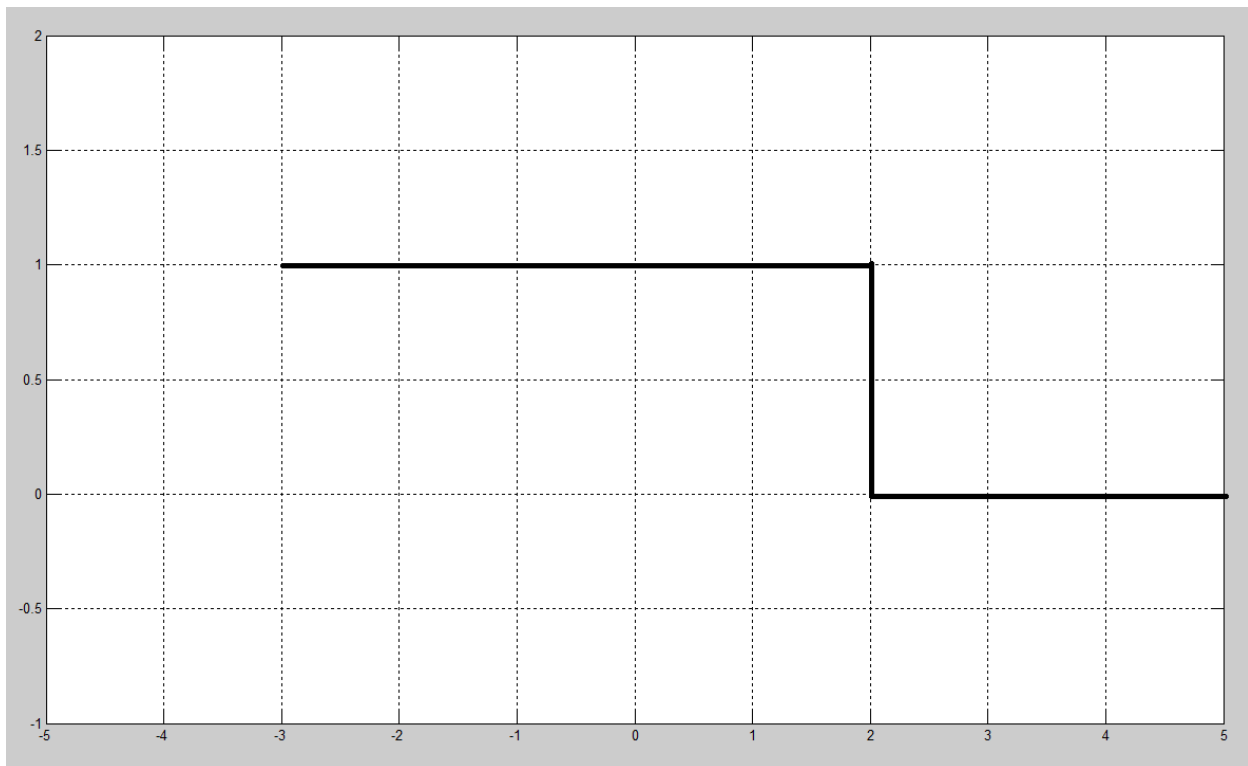
```
n=input('Enter the N value ');
t=0:1:n-1;
y=ones(1,n);
subplot(2,2,2);
stem(t,y);
title('UNIT STEP SEQUENCE SIGNAL');
ylabel('AMPLITUDE');
xlabel('x(a)');
print
```



11- Unit Step Function-I

Write a program to generate a continuous unit impulse signal and compare the generated output

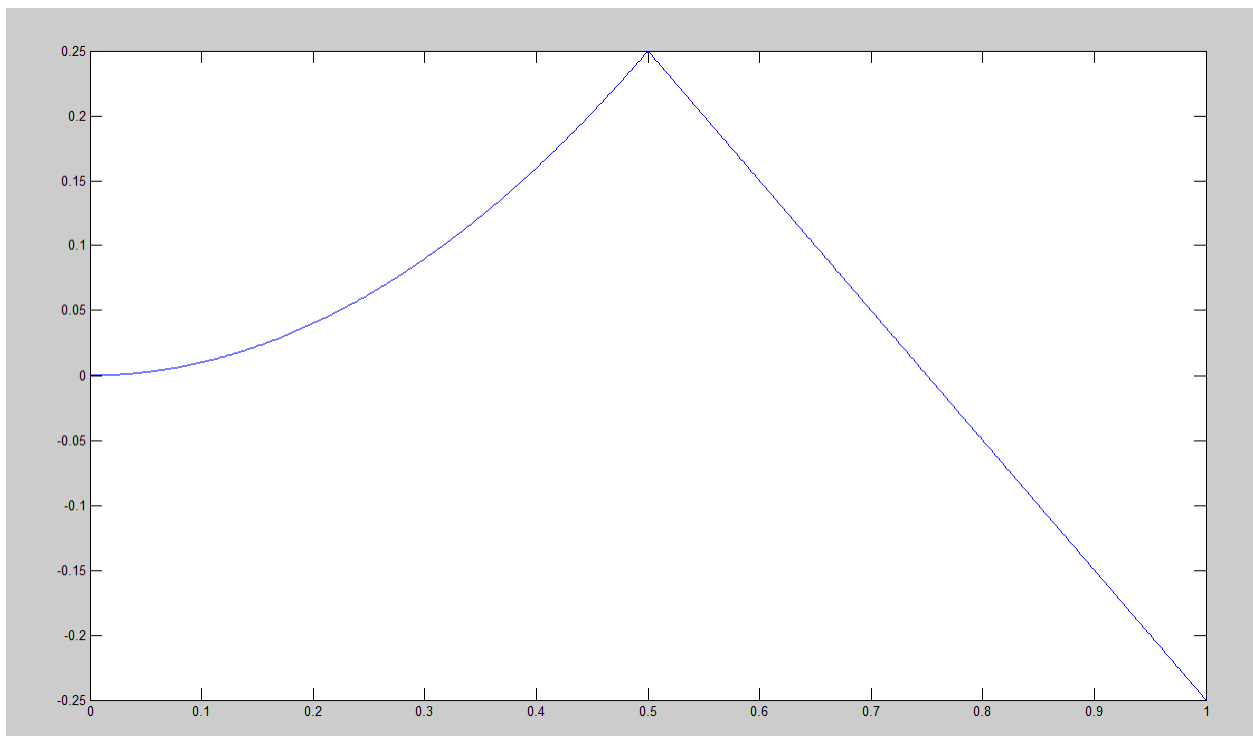
```
t=input('please enter the shift you want in unit step function\n');  
x=[t-5:t:t+5];  
y=[zeros(1,6) ones(1,6)];  
plot(x,y)  
axis([-5 5 -1 2])  
grid on
```



12- UNIT STEP FUNCTION -II

Write a program for linspace function.

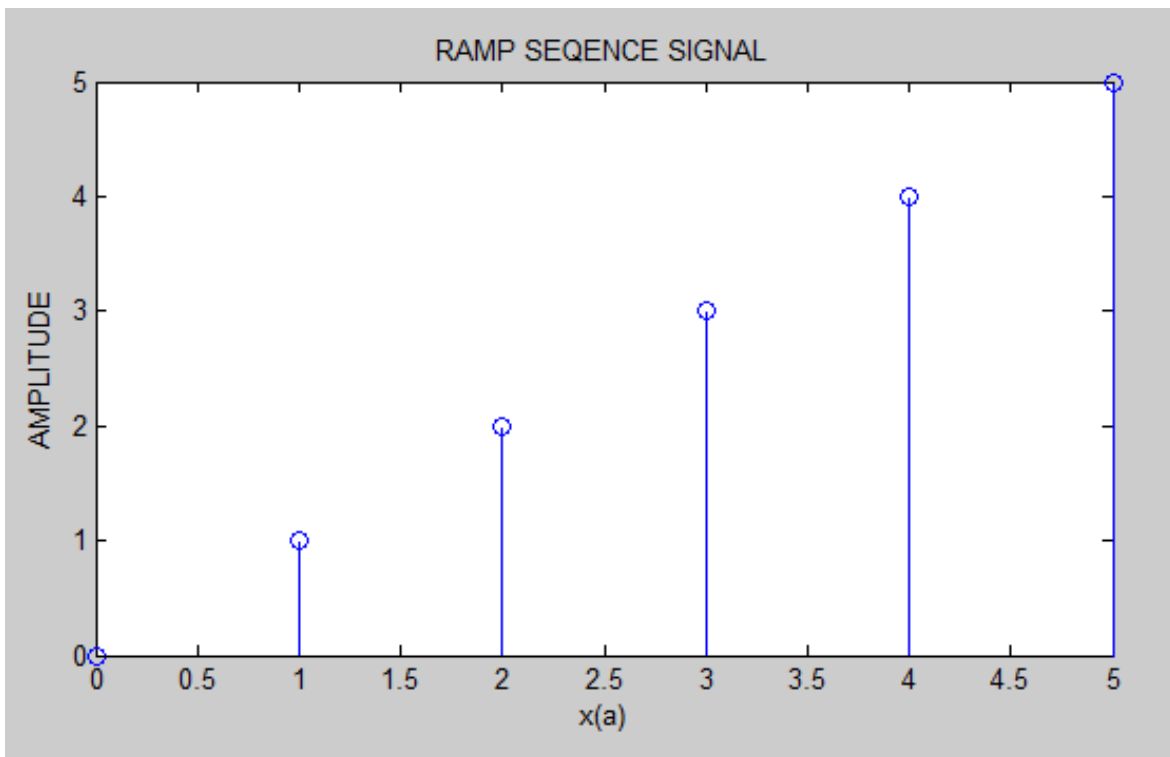
```
t = linspace(0, 1, 500);  
y = (t.^2).*(t < 0.5) + (0.75 - t).*(t >= 0.5);  
plot(t, y)
```



13- Ramp Sequence Signal

Write a program to generate a ramp sequence signal and compare the generated output

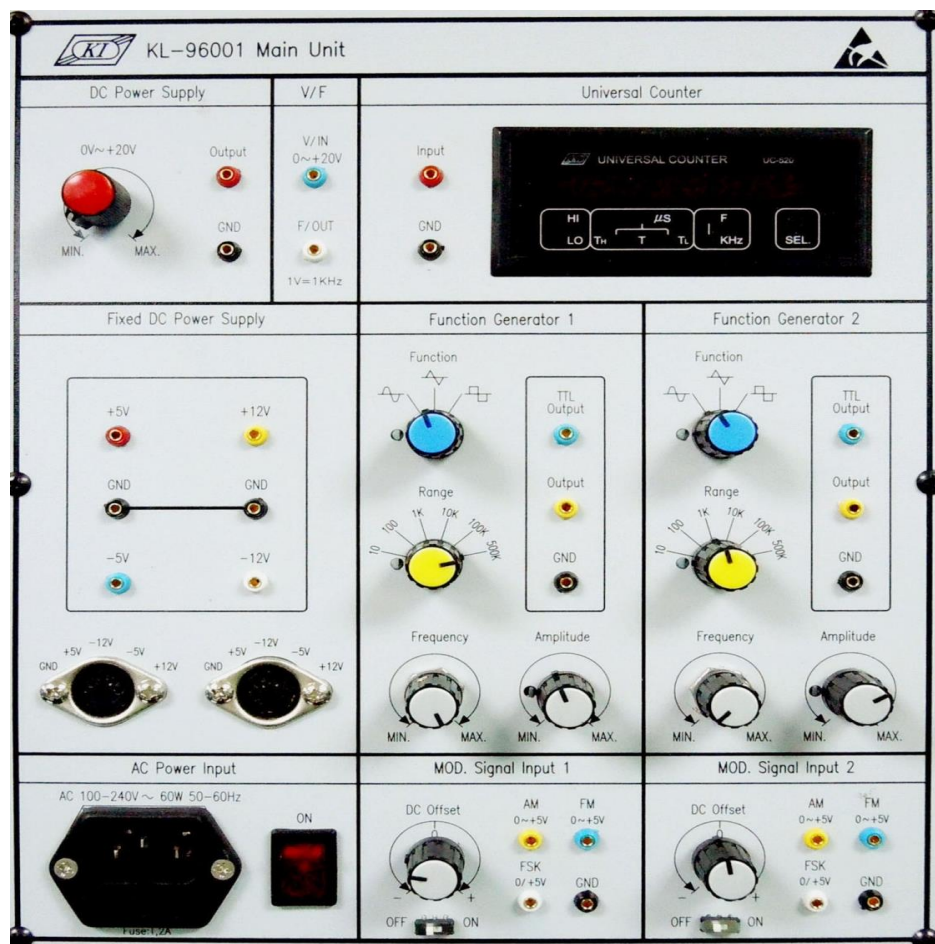
```
n=input('Enter the Length of ramp sequence Signal ');  
t=0:n;  
subplot(2,2,3);  
stem(t,t);  
title('RAMP SEQUENCE SIGNAL');  
ylabel('AMPLITUDE');  
xlabel('x(a)');  
print
```



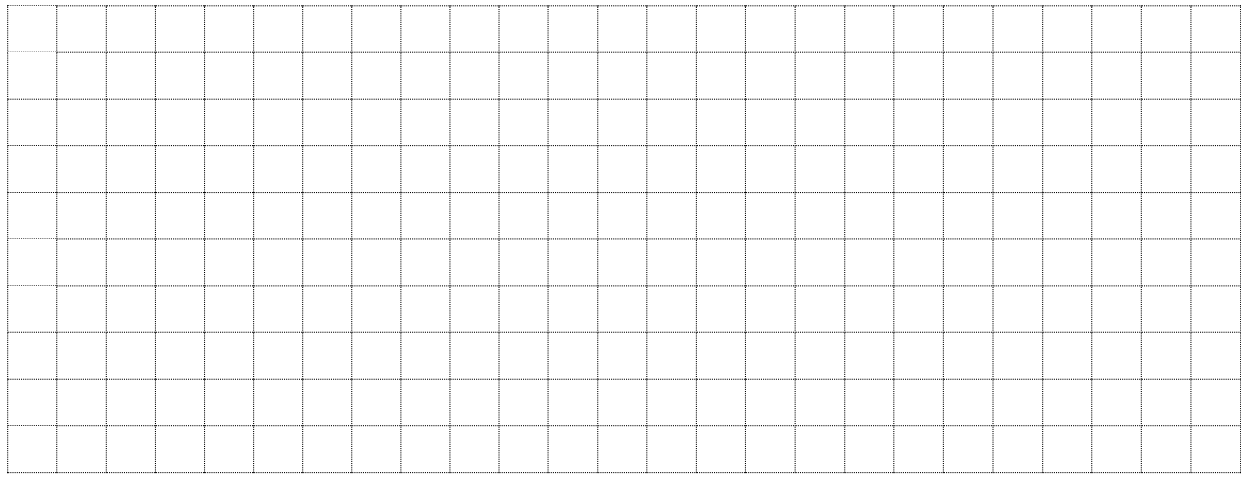
14- Introduction to the Function generator and Oscilloscope

Objectives:

- Get introduced to the Power Supply of **خطأ! لم يتم العثور على مصدر المرجع.** and to the oscilloscope
- Connection and use of fixed and variable DC Power Supply.
- Connection and control of TTL and functions AC Power Supply.
 - Adjustment of the frequency on the range and fine knobs
 - Adjustment of the amplitude and the added DC via their knobs and switch.
- Use of the oscilloscope channels
 - Manual scale and position adjustments
 - Auto-range of scale and position
 - Stopping/Pausing the screen
 - Taking Measurements



Power Supply



13. Get your lab sheets and drawings signed
14. Switch all equipment off
15. Remove and order the wires

15- Waveform generation and there analysis using Fourier Transform

Introduction and Theory:

Fourier analysis of periodic waves is a method of decomposing periodic functions (i.e. $f(x+t) = f(x)$) into an infinite sum of the trigonometric functions sin and cos, known as the Fourier series, like so:

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos nx + \sum_{n=1}^{\infty} b_n \sin nx$$

Here

$$a_0 = \frac{1}{p} \int_{-p/2}^{p/2} f(x) dx$$

$$a_n = \frac{2}{p} \int_{-p/2}^{p/2} f(x) \cos\left(\frac{2\pi nx}{p}\right) dx$$

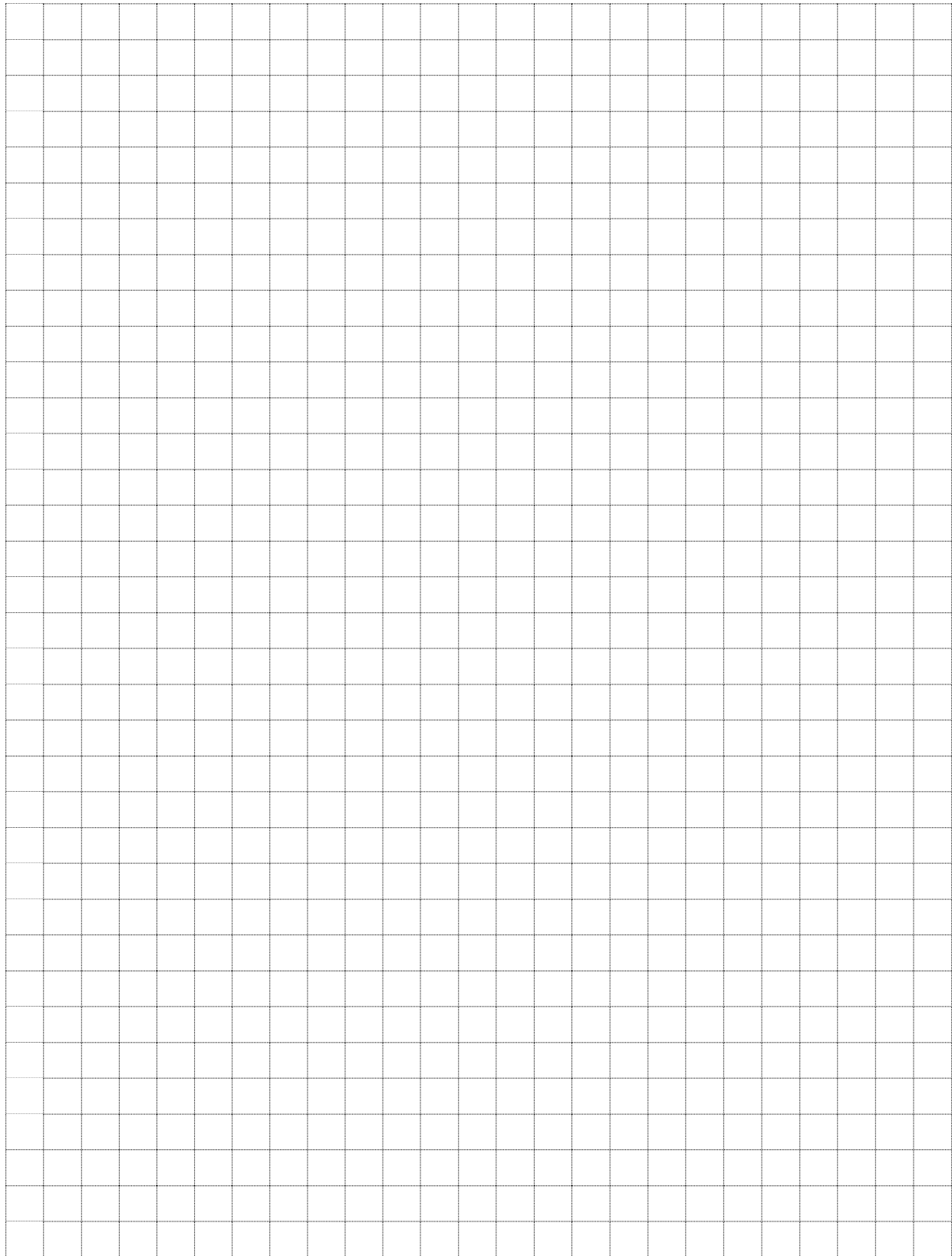
$$b_n = \frac{2}{p} \int_{-p/2}^{p/2} f(x) \sin\left(\frac{2\pi nx}{p}\right) dx$$

These are known as the Euler formulae for the Fourier coefficients, and the coefficients a_n and b_n give an idea of how much of any particular frequency is present in a function's Fourier series.

Experimental Method:

- The correct parameters and settings were applied.
- The graphs of the function and the Fourier transform of a sine wave were obtained and saved.
- The graphs of the function and the Fourier transform of the function were obtained and saved. And the function was identified.
- A sine wave was generated and by increasing the frequency the Fourier transform was observed moving across the screen.

Report both, the considered signal and the generated Fourier Transform signal, on the grid below.



Results and Analysis:

By Euler's Theorem, a sine wave can be represented as

$$\sin(2\pi f_0 t) = \frac{e^{i2\pi f_0 t} - e^{-i2\pi f_0 t}}{2i}$$

We then have the Fourier transform of this sine wave:

$$\begin{aligned}\hat{x}(f) &= \int_{-\infty}^{\infty} \frac{e^{i2\pi f_0 t} - e^{-i2\pi f_0 t}}{2i} \cdot e^{-2\pi i f t} dt \\ &= \frac{1}{2i} (\delta(f - f_0) - \delta(f + f_0))\end{aligned}$$

Where $\delta(x)$ is the Dirac Delta function. Since a sine wave consists of only one frequency we have $f=f_0$ and the Fourier transform has a peak at f_0 only, which we can see from the graph below.

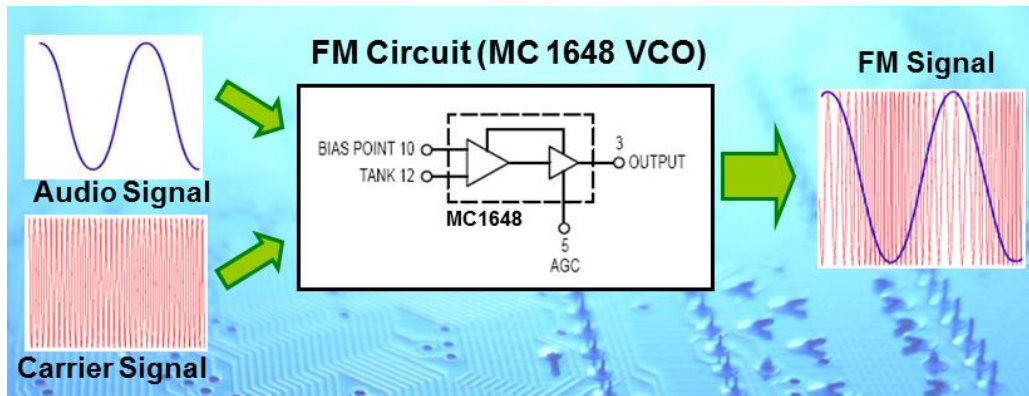
Discussions and Conclusions:

- o The Fourier transforms of many waves were investigated, recorded and explained.
- o Overall a working knowledge of Fourier transforms was gained, and the experiment was regarded as highly successful as the experimental data verified the theory throughout.

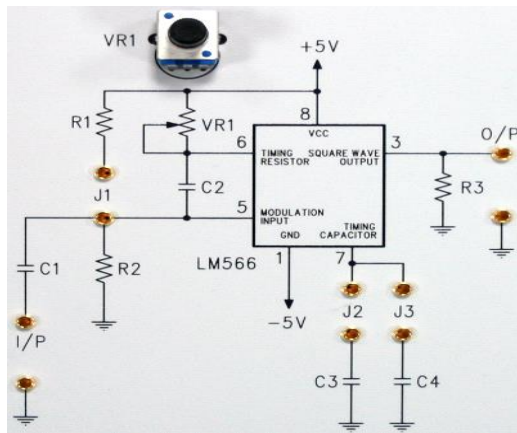
16- Frequency Manipulation

Background:

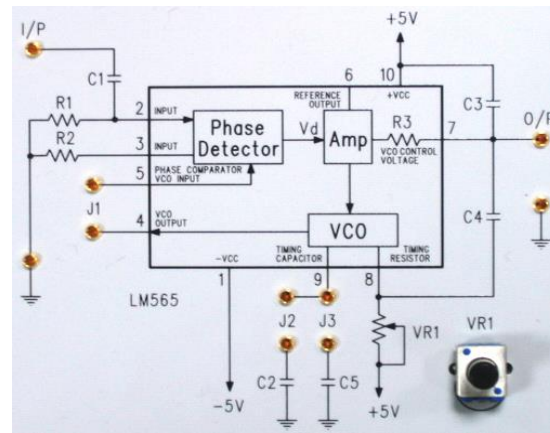
A basic feature of any signal is its frequency range. This feature has been exploited in communication engineering. With frequency modulation, the modulating signal and the carrier are combined in such a way that causes the carrier FREQUENCY (f_c) to vary above and below its normal frequency while the amplitude of the carrier remains constant.



Circuit Explanation



LM566



LM565

LM566 is a voltage-controlled oscillator of exceptional linearity with buffered squared wave and triangle wave outputs.

VR1&C3 / VR1&C4: used to determine the VCO center frequency.

I/P: for control voltage input.

O/P: for square wave output whose frequency is proportional to the voltage difference of $V_8 - V_5$

LM565 is a general purpose Phase Locked Loop widely used in frequency demodulation.

VR1&C2 / VR1&C5: used to determine the VCO center frequency.

R3 and C3: Used to form a Low Pass Filter.

I/P: FM signal input.

O/P: Demodulating signal output

Experiment Procedure:

1. Power on the Module KL-93004 and use clip jumpers to connect J1 and J3.
2. Connect a 500Hz 2V Vpp sine wave to I/P.
3. Connect CH1 and CH2 of the scope to I/P and O/P.
4. Adjust VR1 to obtain a smooth FM wave
5. Draw both waves below:



6. Use clip jumpers to connect J1 and J3 of FM Demodulator (FM565).
7. Connect O/P of the LM566 to the I/P of the LM565.
8. Connect CH1 to the LM566 I/P and CH2 to the LM565 O/P.
9. Adjust LM565 VR1 to obtain a smooth demodulating output.
10. Record the waveforms below



11. Sign your Lab sheets and drawings
12. Switch all equipment off
13. Remove and order the wires