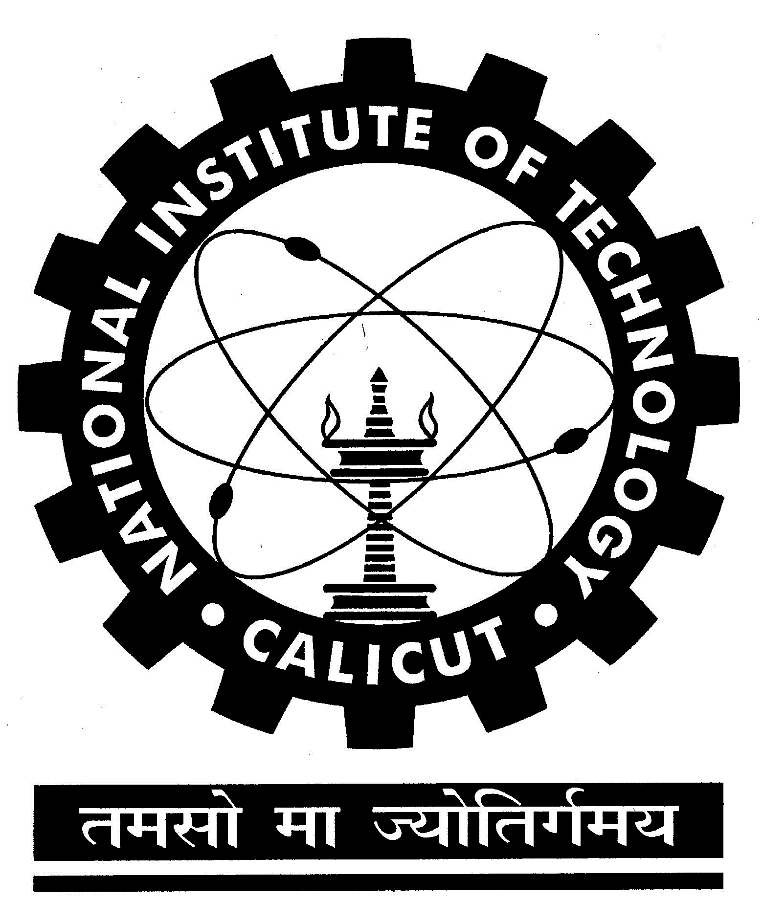
**NATIONAL INSTITUTE OF TECHNOLOGY CALICUT**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

****

**EC4091: DIGITAL SIGNAL PROCESSING LABORATORY**

Additional Experiment Cycle 1

Submitted by,

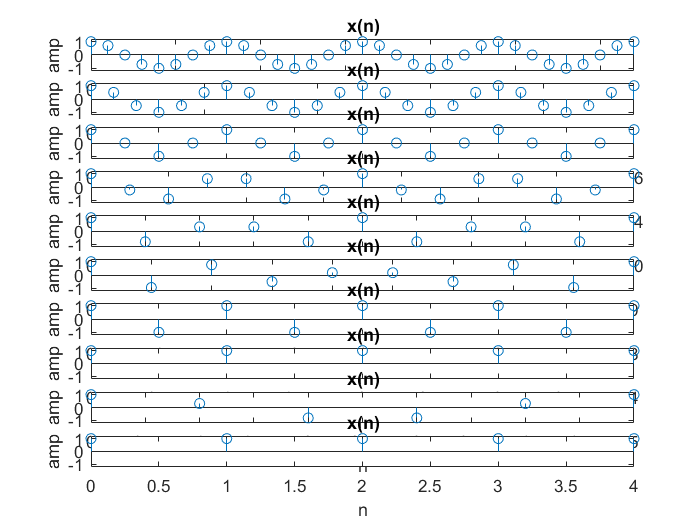
Anjali Venugopal B15007EC

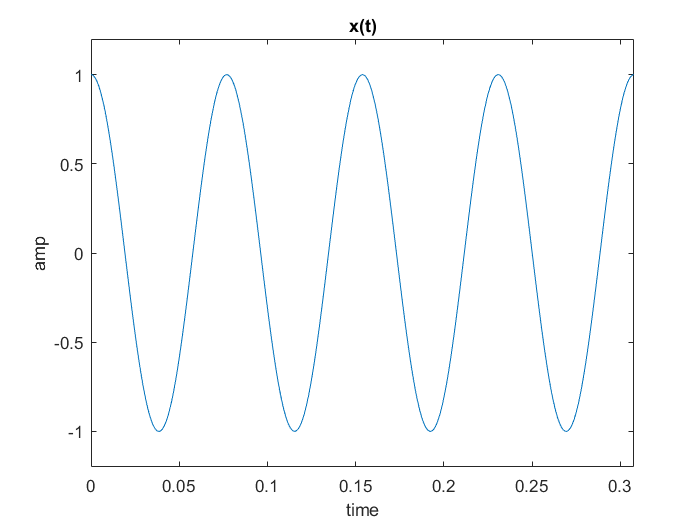
Anju Anand B150303EC

Ankit Kumar B150912EC

Aravind Krishnan B150829EC

(1)Ans:





**AIM**

Write MATLAB codes for each question.

**QUESTIONS**

1. Sampling: Program below illustrates the sampling of a continuous time sinusoid of frequency 13 Hz. Since Matlab cannot strictly generate a continuous time signal, we simulate a continuous time signal by sampling it at a very high rate Th=0.0001 sec. A plot of the samples using plot command will then look like a continuous time signal. Since the frequency of the continuous time signal is 13Hz, the Nyquist rate is T=1/26. Run the program for several values of T, below and above the Nyquist rate such as 1/(8\*13), 1/(6\*13), 1/(4\*13), 2/(7\*13), 1/(3\*13), 2/(5\*13), 4/(9\*13), 1/(2\*13), 2/(3\*13), 4/(5\*13), 1/13, etc. . (all rational multiples of 1/13 so that the resulting discrete time signal is periodic) Which of the discrete time signals will give correct reconstruction of the continuous time signal? Mathematically determine the angular frequency in rad/sample and the fundamental period N of each of the discrete time signals and verify from the plots.

Ans:

**MATLAB Code:**

f=13;

tmax=4/13;

t=0:0.0001:tmax;

xa=cos(2\*pi\*f\*t);

plot(t,xa);

xlabel('time');

ylabel('amp');

title('x(t)');

axis([0 tmax -1.2 1.2]);

T=[1/(8\*13) 1/(6\*13) 1/(4\*13) 2/(7\*13) 2/(5\*13) 4/(9\*13) 1/(2\*13) 2/(2\*13) 4/(5\*13) 1/13];

for i=1:10

nmax= tmax/T(i);

n=0:nmax;

xs= cos(2\*pi\*f\*n\*T(i));

subplot(10,i,10);

stem(n,xs);

xlabel('n');

ylabel('amp');

title('x(n)');

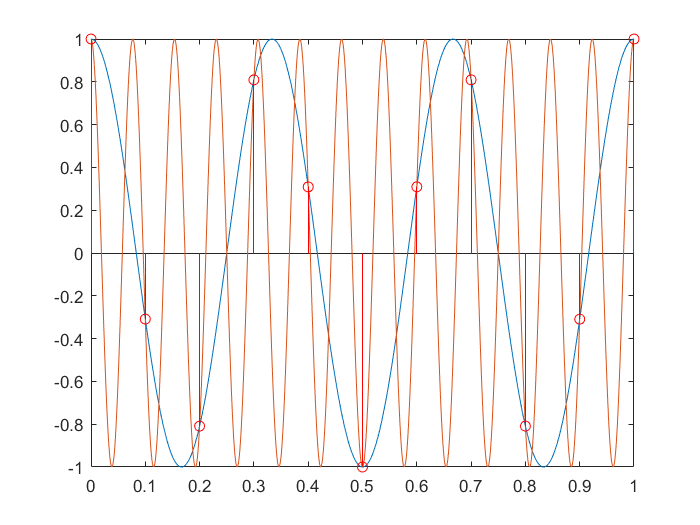
axis([0 nmax -1.2 1.2]);

end

The signal reconstructed using T= 1/(8\*13) will give the perfect output as the sampling period is very less.

|  |  |  |
| --- | --- | --- |
| **Nyquist rate = T** | **Angular Frequency(w)= (2π\*13\*T) rad/sample** | **Fundamental period (N)=**  **k/(13\*T)** where k is an integer |
| 1/(8\*13) | π/4 | 8 for k=1 |

(2) Ans:



|  |  |  |
| --- | --- | --- |
| 1/(6\*13) | π/3 | 6 for k=1 |
| 1/(4\*13) | π/2 | 4 for k=1 |
| 2/(7\*13) | 4π/7 | 7 for k=2 |
| 2/(5\*13) | 4 π/5 | 5 for k=2 |
| 4/(9\*13) | 8 π/9 | 9 for k=4 |
| 1/(2\*13) | π | 2 for k=1 |
| 2/(2\*13) | 2π | 2 for k=2 |
| 4/(5\*13) | 8π/5 | 5 for k=4 |
| 1/13 | 2π | 1 for k=1 |

1. The family of continuous time sinusoids cos(Ω0+kΩs)t , k=0,±1, ±2..... where Ωs=2π/T leads to identical sampled sequences.( Prove) This phenomenon of a continuous time sinusoid of a higher frequency acquiring the identity of a sinusoidal sequence of lower frequency after sampling is called aliasing. Consider the continuous time sinusoid g1(t)=cos(6πt). When sampled with T=0.1 sec, it will lead to cos[.6πn]. When sampled with T=0.1 sec, g2(t)=cos(26πt) will also lead to cos[.6πn]. (Verify mathematically). Run the program below to see aliasing in this case.

Ans:

Let x(t) = cos(Ω0t) , x1(t) = cos(Ω0+kΩs)t

now we sample the signal x(t) by letting t= nT, where T is the sampling rate

x(n) = cos(Ω0nT)

Because sinusoids are periodic with period 2

So, x(n) = cos(Ω0nT + 2) = cos(Ω0 +kΩs)nT where Ωs = 2, k is an integer

Hence, x1(t) will be sampled in same sequence as x(t).

In the question given,

x1(t)= cos(26) = cos(6πt + 2\*10\*πt)

here, Ωs = 2/0.1 = 20

**MATLAB Code:**

Th=.001;

tmax=1;

t=0:Th:tmax;

g1=cos(6\*pi\*t);

g2=cos(26\*pi\*t);

plot(t,[g1;g2]);hold on;

T=.1; %sampling period

nmax=tmax/T; n=0:nmax;

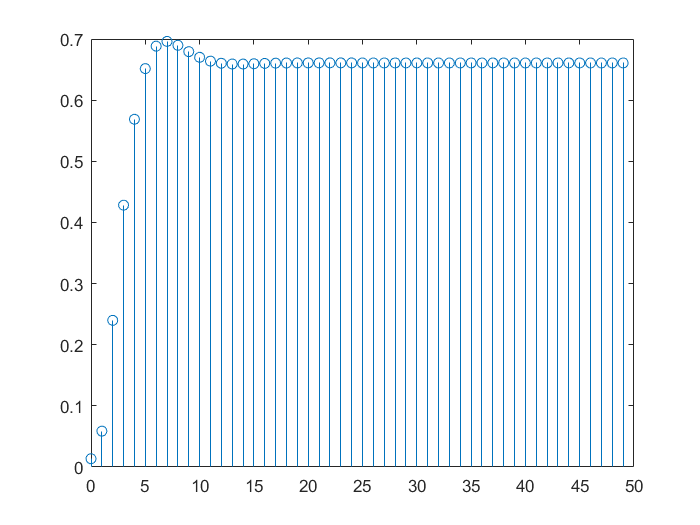
gn=cos(.6\*pi\*n);

stem(n\*T,gn,'r')%time axis denormalised by using n\*T so that the

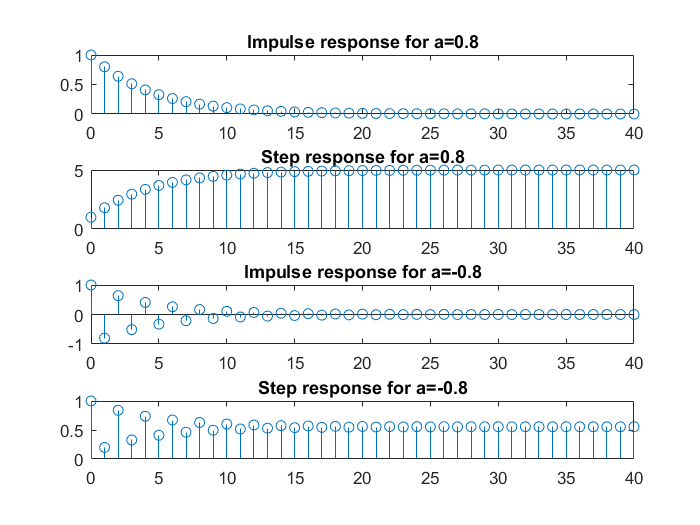
%samples can be superimposed

1. The filter() function : The filter() command recursively computes the the output y(n) of an LTI system described by a difference eqn from the input x(n) and initial conditions. b=[b0,b1,.....bM ]; a=[ a0,a1,......aN]; y=filter(b,a,x) . The number of output values in y correspond to the number of input values in x. See help on filter for more details.

(3a) Ans:



(3b) Ans:



1. Run the code fragment below to determine the first 50 values of the output of the system described by y(n)-1.143y(n-1)+.4128y(n-2)=.0675x(n)+.1349x(n-1)+.675x(n-2) if the initial conditions are zero and x(n)=.2u(n).

a=[1 -1.143 .4128 ]; b=[.0675 .149 .675];   
y=filter(b,a,.2\*ones(1,50)); stem(0:49,y)

1. Using filter(), determine and stem the first 41 samples of the impulse and step response of the system described by y(n)-ay(n-1)=x(n) for a=.8 and -.8. Verify that the step response is the running sum of the impulse response.

Ans:

**MATLAB Code:**

a1=0.8;

c=[1 -1\*a1]; b=[1 0];

p=filter(b,c,[1 zeros(1,40)]);

subplot(411)

stem(0:40,p);

title('Impulse response for a=0.8');

q=filter(b,c,[1 ones(1,40)]);

subplot(412)

stem(0:40,q);

title('Step response for a=0.8');

a2=-0.8;

d=[1 -1\*a2]; e=[1 0];

r=filter(e,d,[1 zeros(1,40)]);

subplot(413)

stem(0:40,r);

title('Impulse response for a=-0.8');

s=filter(e,d,[1 ones(1,40)]);

subplot(414)

stem(0:40,s);

title('Step response for a=-0.8');

Yes, the step response is the running sum of the impulse response.

1. Run the following program to generate output using both conv() and filter():

h = [3 2 1 -2 1 0 -4 0 3]; % impulse response

x = [1 -2 3 -4 3 2 1]; % input sequence

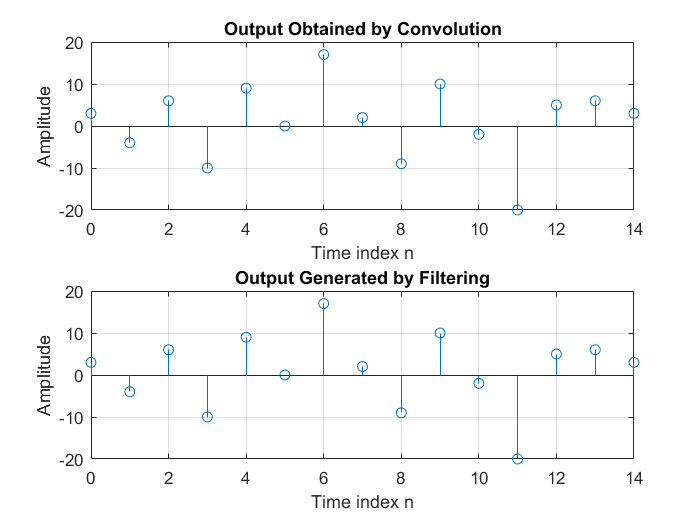
y = conv(h,x);

n = 0:14;

subplot(2,1,1);stem(n,y);

xlabel('Time index n'); ylabel('Amplitude');

(3c)Ans:



title('Output Obtained by Convolution');grid;

x1 = [x zeros(1,8)];

y1 = filter(h,1,x1);

subplot(2,1,2);stem(n,y1);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Generated by Filtering');grid;

Is there any difference between y[n] and y1[n]? What is the reason for using x1[n] obtained by zero-padding x[n] as the input for generating y1[n]?

Ans: No, there is no difference between y[n] and y1[n]. Here both convolution and filtering gives the same response. Using filter function the

number of output (y1)values produced corresponds to the number of values in input(x).

Thus if x is given as input without zero padding , filtering operation gives only first 7

(length of input)values as output.These are same as the first 7 values obtained by

convolution.

**INFERENCES**

* According to Nyquist sampling theorem,a bandlimited signal can be

reconstructed exactly if it is sampled at a rate equal to atleast twice the

maximum frequency component.

* The lower the sampling rate, it is easier to get the reconstructed signal perfectly.
* Both convolution and filtering gives the same response. But if we don”t give zero padding filtering operation gives only a length equal to length of input only.

**RESULT**

The MATLAB codes for each question was written and the plots were verified.