****

**CHITTAGONG UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

Department of Electrical and Electronic Engineering

Name of the report: Report on Digital Signal Processing

Course no: EEE-496

Course title: Digital Signal Processing Sessional

Date of submission: 01/08/2022

Name: MD. Sayedul

ID: 1702079

Section: B

Group:B1

**Problem 01:** Illustrate the spectral leakage phenomenon with suitable example using matlab

**Matlab Code:**

**clc;**

**clear;**

**close all;**

**Fs = 2000;**

**N = 4080;**

**t = (0:(N-1))/Fs;**

**y = sin(2\*pi\*100\*t);**

**plt=Plot(t, y)**

**plt.ShowBox="off";**

**plt.XLim=[2, t(end)]**

**plt.XLabel='t(sec)'**

**plt.YLabel="x(t)";**

**plt.Title="The last .04 sec of x(t) when N=4086";**

**plt.XGrid="on";**

**plt.YGrid="on";**

**Y = fft(y);**

**f = (Fs/N) \* (0:N-1);**

**plt=Plot(f,abs(Y)/sqrt(N))**

**plt.ShowBox="off";**

**plt.XLim=[90, 110];**

**plt.XLabel="frequency (Hz)"**

**N2 = 4096;**

**t2 = (0:(N2-1))/Fs;**

**x2 = sin(2\*pi\*100\*t2);**

**plt=Plot(t2, x2);**

**plt.ShowBox="off";**

**plt.XLabel='t(sec)'**

**plt.YLabel="x2(t)";**

**plt.Title="The last .047 sec of x(t) when N=4096";**

**plt.XLim=[2, t2(end)];**

**Y2 = fft(x2);**

**f2 = (Fs/N2) \* (0:N2-1);**

**plt=Plot(f,abs(Y)/sqrt(N), f2,abs(Y2)/sqrt(N2))**

**plt.Legend={"N=4080", "N=4096"}**

**plt.XLim=[90, 110];**

**plt.ShowBox="off";**

**plt.XLabel="frequency (Hz)"**

**Output:**

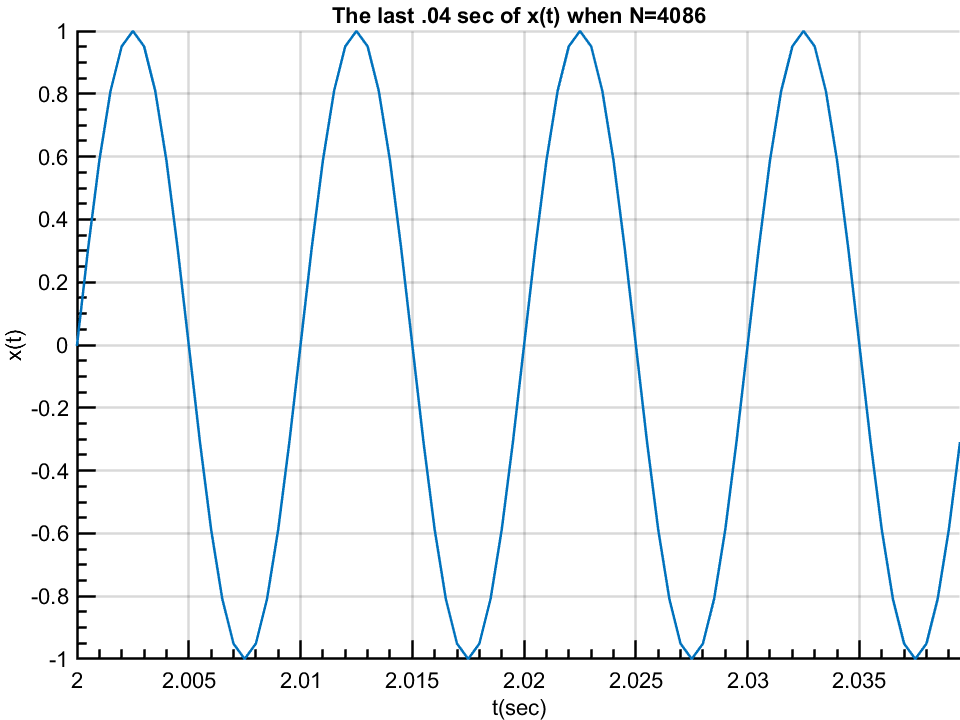


Figure.01:There is no discontinuity at the beginning and end of the segment.

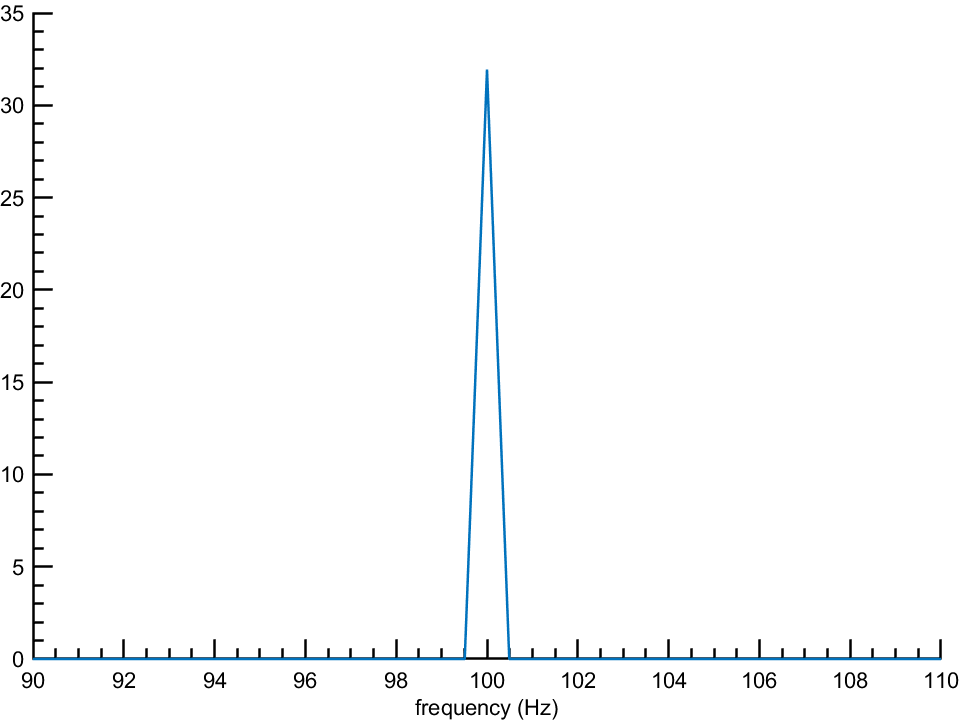


Figure.02: Fourier transform of a signal whenthere is no discontinuity at the beginning and end of the segment.

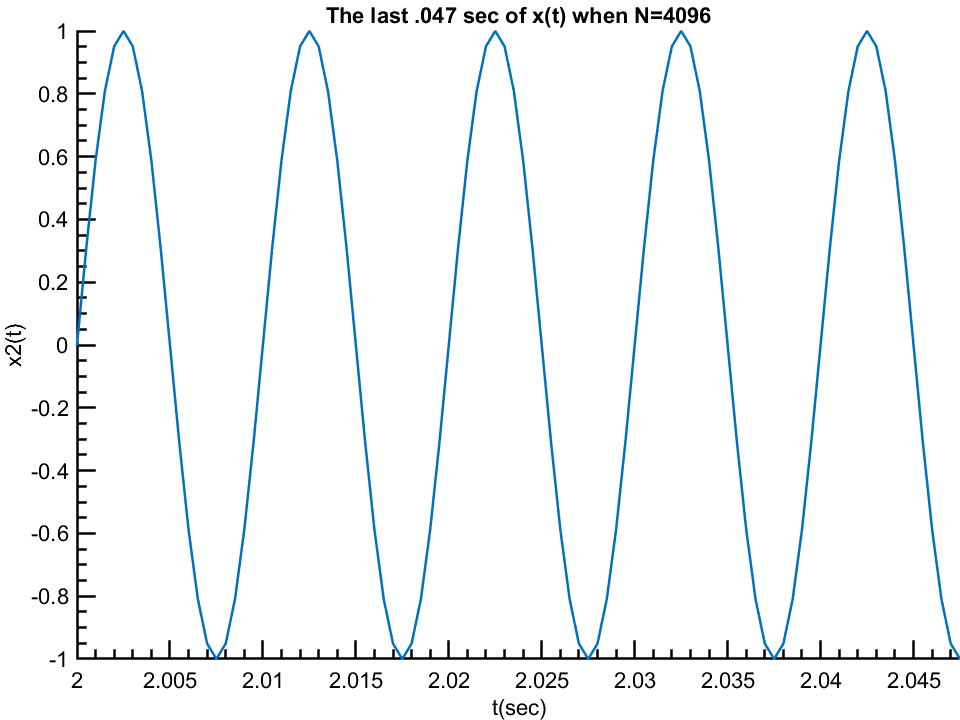


Figure.03:There is discontinuity at the beginning and end of the segment.

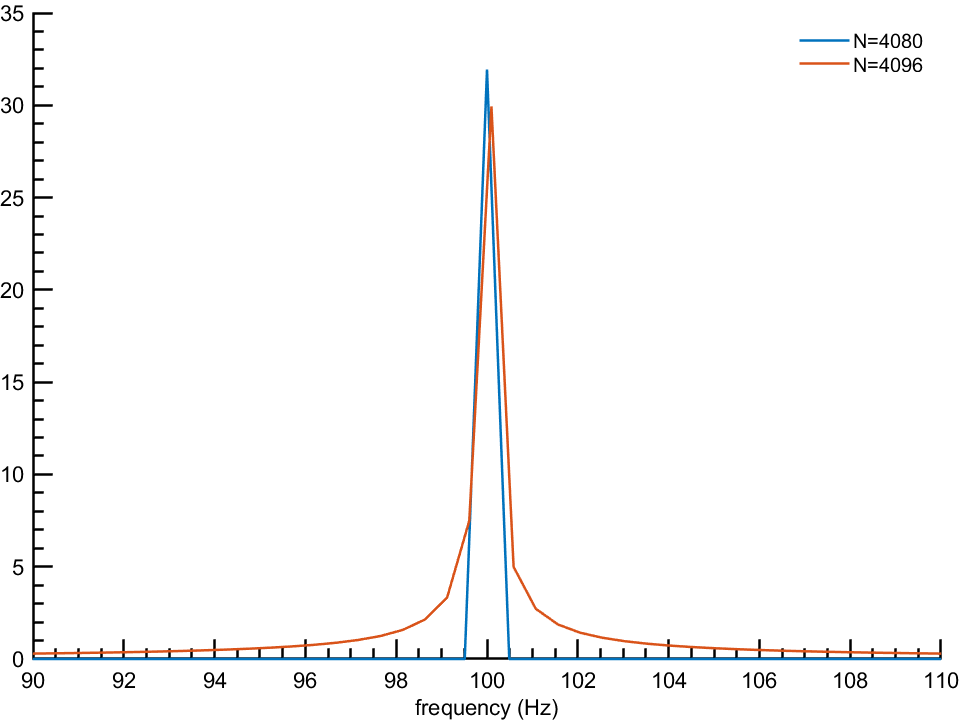


Figure.04: Fourier transform of a signal whenthere is no discontinuity at the beginning and end of the segment.

**Problem 02:** Illustrate the aliasing phenomenon with suitable example.

**MatLab Code:**

**f0 = 50;**

**fs = 80;**

**tmin = 0;**

**tmax = 5/f0;**

**t = tmin:0.001:tmax;**

**x1 = cos(2\*pi\*f0\*t)+sin(2\*pi\*(f0-20)\*t);**

**Ts = 1/fs;**

**ts = tmin:Ts:tmax;**

**x1resampled = cos(2\*pi\*f0\*ts)+sin(2\*pi\*(f0-20)\*ts);**

**x1reconstructed = zeros(1,length(t));**

**samples = length(ts);**

**for i = 1:1:length(t)**

**for n = 1:1:samples**

**x1reconstructed(i) = x1reconstructed(i) + x1resampled(n)\*sinc((t(i)-n\*Ts)/Ts); %%% CHANGE**

**end**

**end**

**figure**

**plot(t,x1, "linewidth", 1, "color", "k")**

**hold on**

**stem(ts,x1resampled,"r","filled", "linewidth", 1.3)**

**xlabel("t(sec)")**

**ylabel("signal");**

**figure**

**plot(t,x1reconstructed, "linewidth", 1.5)**

**title("Aliasing when fs<2f0")**

**xlabel("t(sec)")**

**ylabel("signal")**

**Output:**

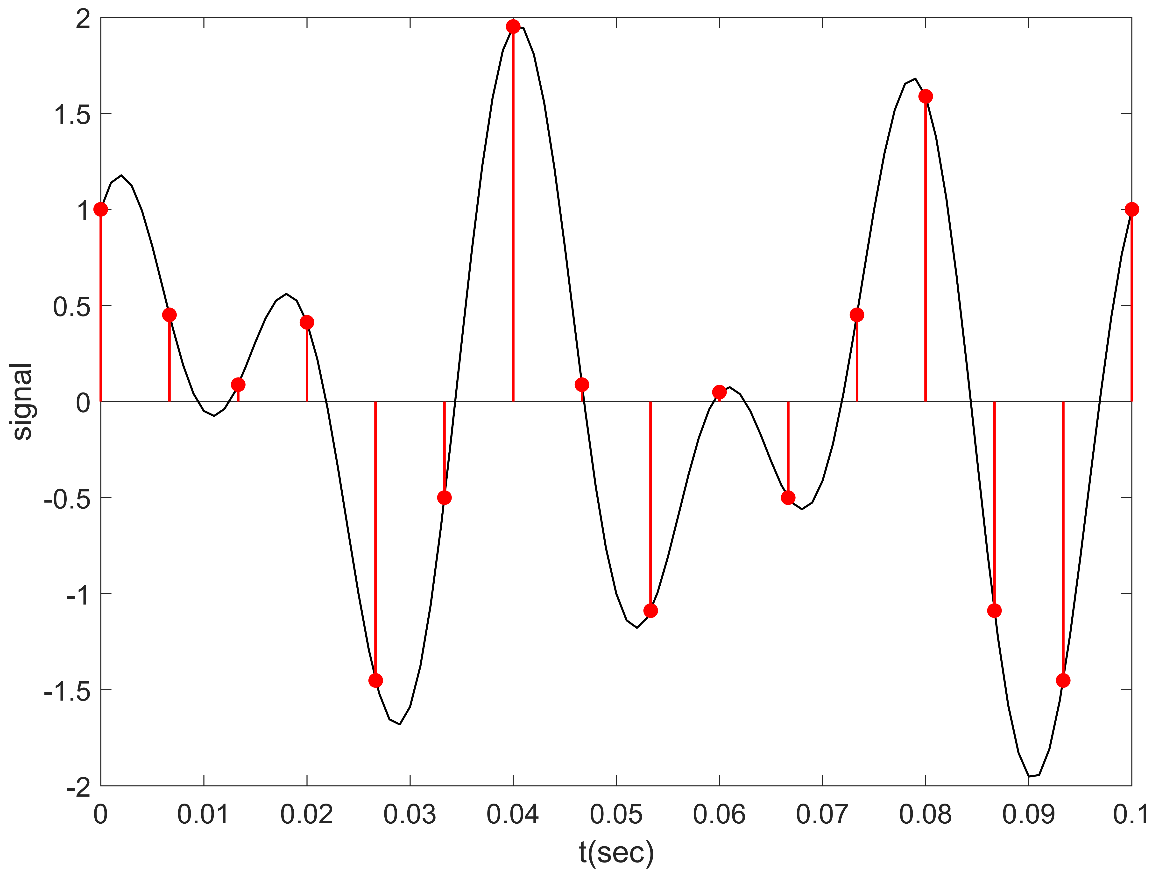


Figure.05:The sampling signal when fs>=f0

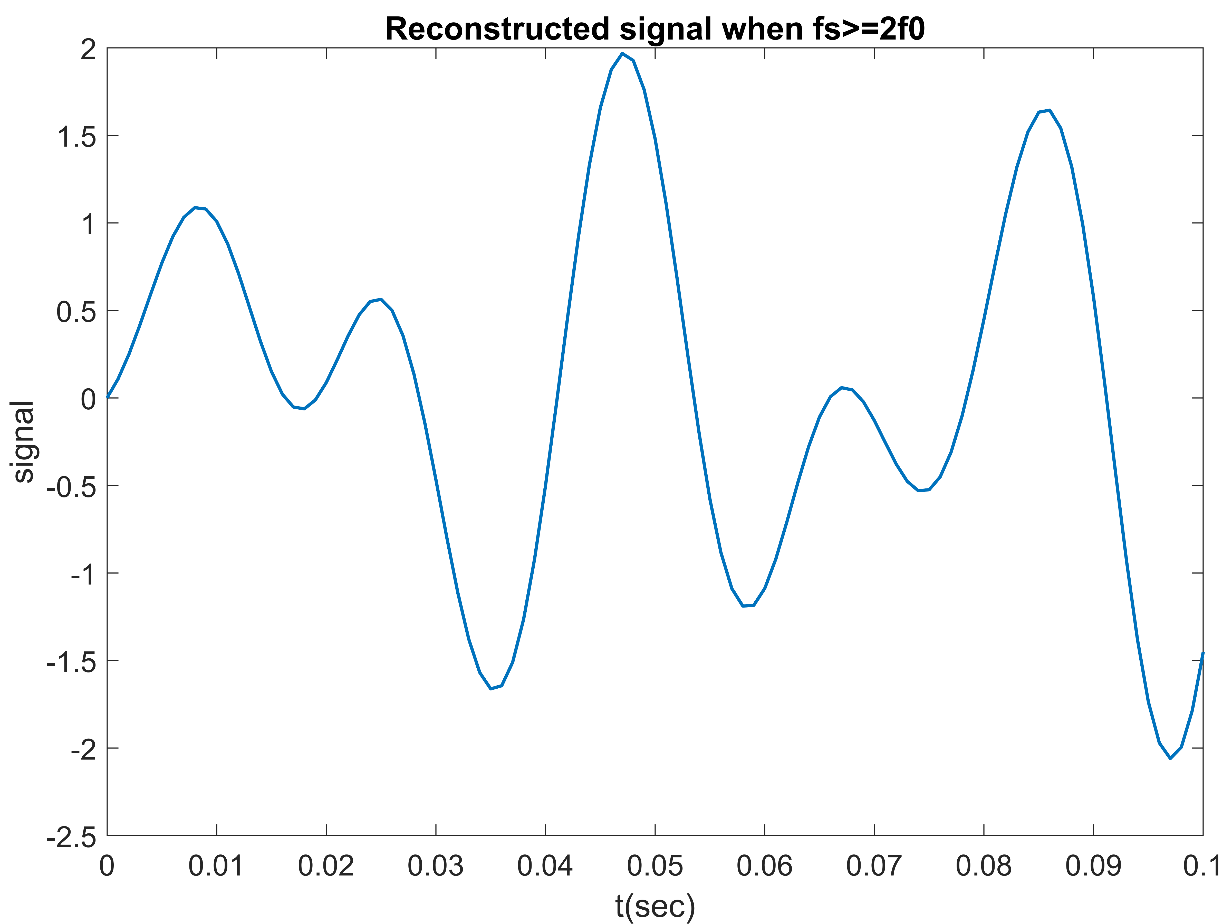


Figure.06: Reconstructed signal when fs>=f0

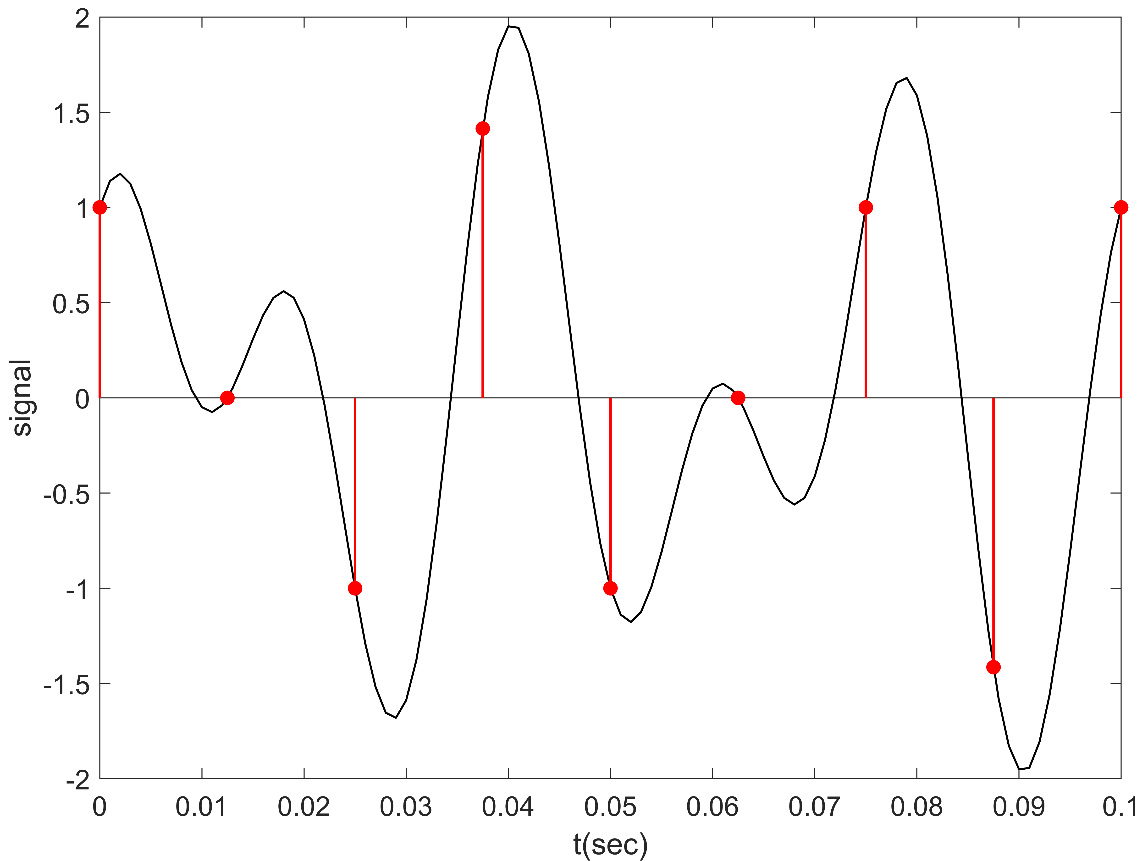


Figure.06:The sampling signal when fs<f0

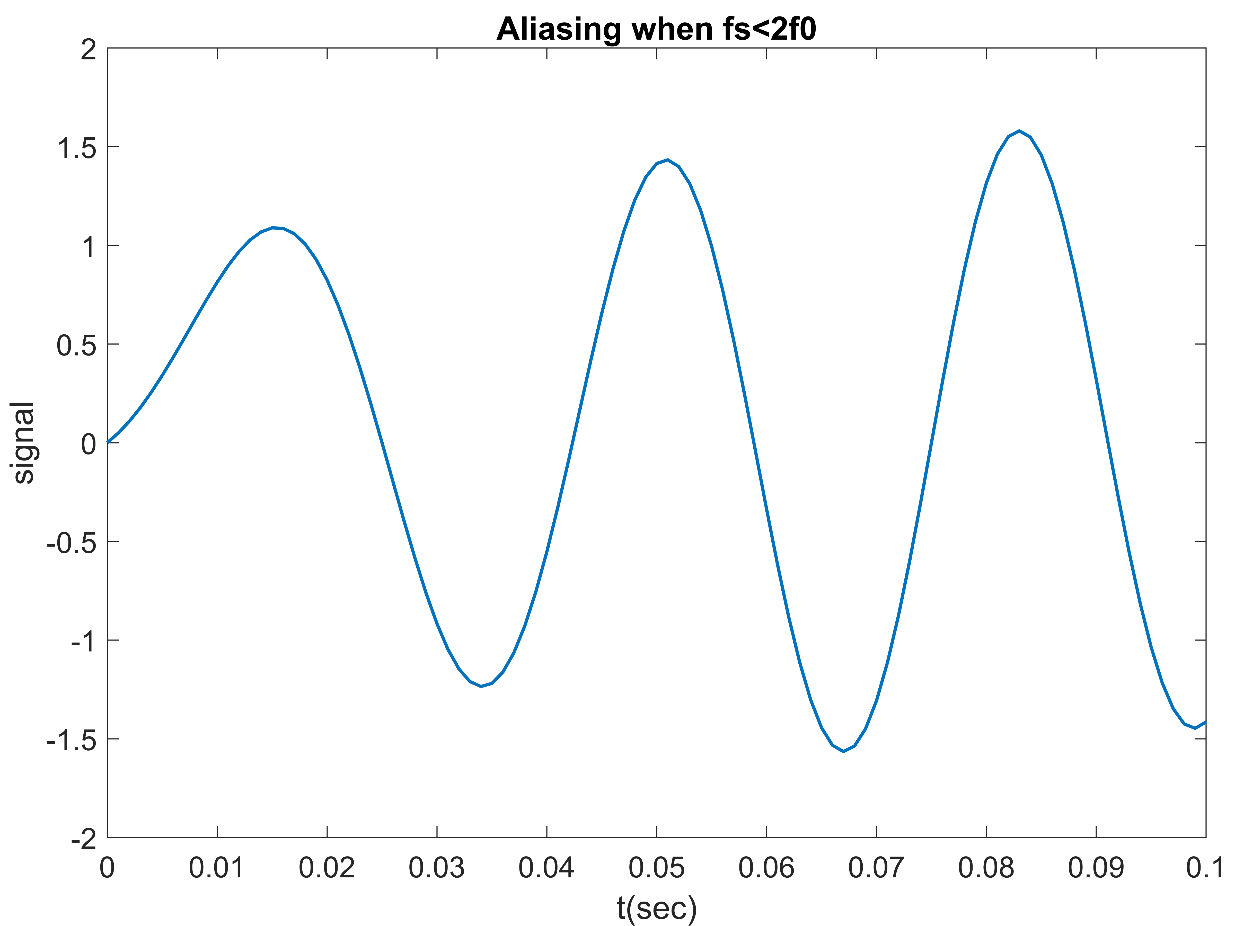


Figure.06: Reconstructed signal when fs<f0

**Problem 03**: Wavlet analysis and synthesis for sampling frequency Fs=8 sample/sec

**Matlab Code:**

**clc;**

**close all;**

**Fs=8**

**n=-5:1/8:5**

**x= heaviside(n+4)-heaviside(n+2)+triangularPulse(-2, -1, 0, n)........**

**+triangularPulse(0, 1, 2, n)+ heaviside(n-2)-heaviside(n-4)**

**[gtilde,htilde,g,h] = wfilters('db2');**

**[lowpass,highpass] = dwt(x,gtilde,htilde);**

**V0= downsample(lowpass,2)**

**V1= downsample(highpass,2)**

**W0= upsample(V0,2)**

**W1= upsample(V1,2)**

**Z0=conv(W0,g)**

**Z1=conv(W1,h)**

**xrec=Z0+Z1**

**figure**

**subplot(3,1,1),stem(lowpass, "filled")**

**subplot(3,1,2),stem(highpass, "filled")**

**subplot(3,1,3),stem(V0, "filled")**

**suptitle('Wavelet analysis and synthesis')**

**figure**

**subplot(3,1,1),stem(V1, "filled")**

**subplot(3,1,2),stem(W0, "filled")**

**subplot(3,1,3),stem(W1, "filled")**

**suptitle('Wavelet analysis and synthesis')**

**figure**

**subplot(3,1,1)**

**stem(Z0, "filled")**

**subplot(3,1,2)**

**stem(Z1, "filled")**

**subplot(3,1,3)**

**stem(xrec, "filled")**

**suptitle('Wavelet analysis and synthesis')**

**Output**:

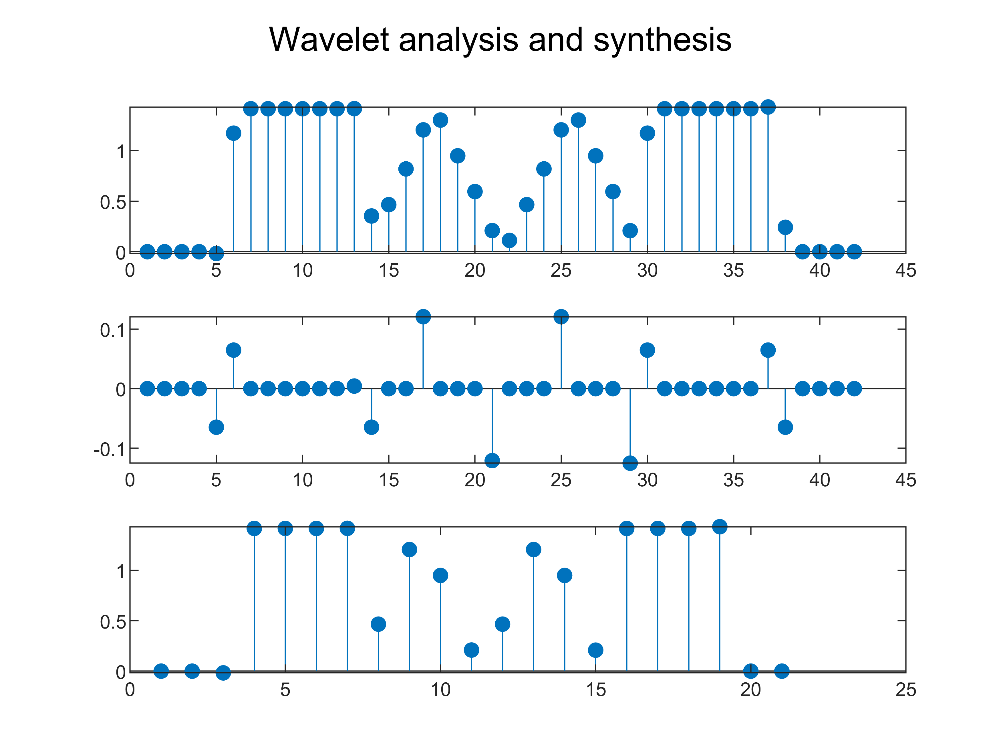


Figure.07: Wavlet analysis and synthesis

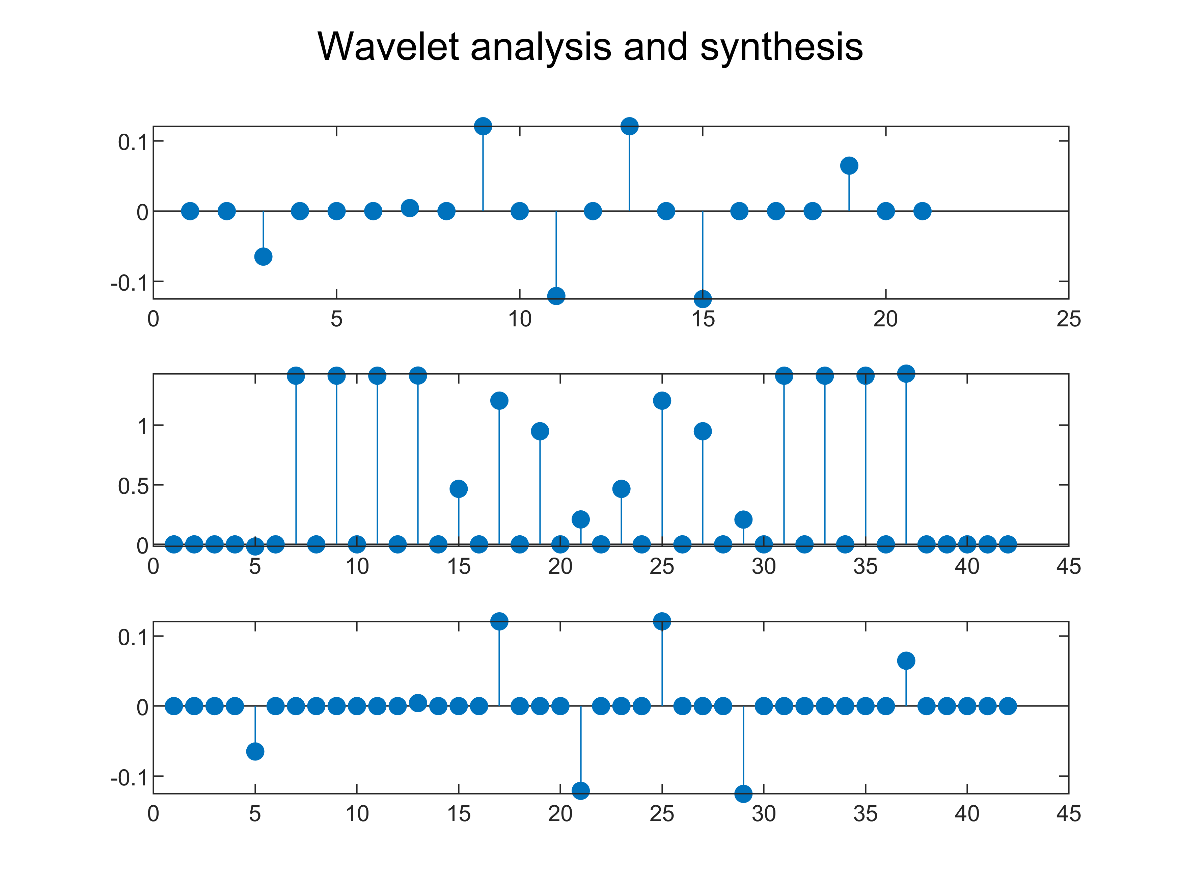


Figure.08: Wavlet analysis and synthesis

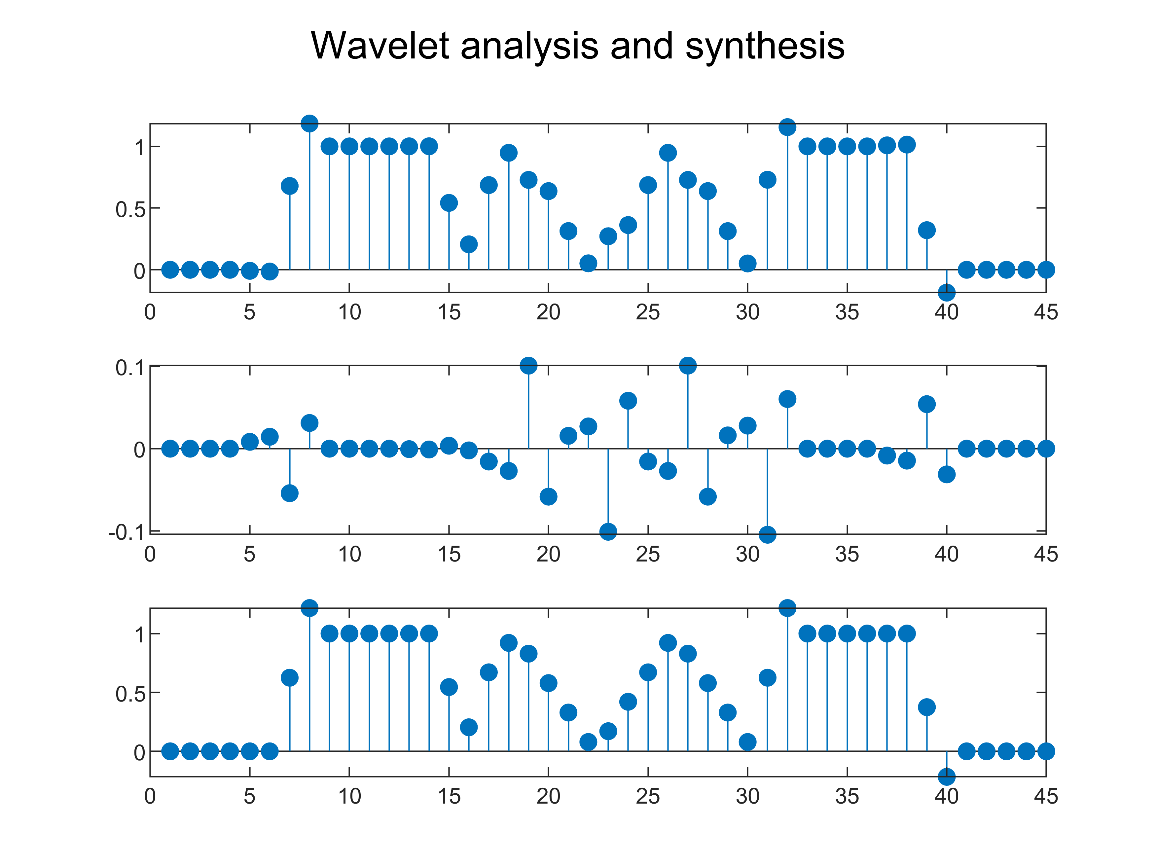


Figure.09: Wavlet analysis and synthesis

**Problem 04.**

**Matlab Code:**

**clc**

**clear**

**close all**

**Fs = 8;**

**F=-5:1/Fs:5;**

**X\_w=triangularPulse(-1, 0, 1, F);**

**C = zeros(1,3\*length(X\_w));**

**p = 0;**

**NFFT = 4096;**

**ff = (-0.5: 1/NFFT: 0.5-1/NFFT)\*Fs;**

**MAG\_X2 = ifftshift(abs(ifft(X\_w,NFFT)));**

**figure**

**subplot(311), plot(F, X\_w)**

**title('x[n]')**

**subplot(312),plot(MAG\_X2,'LineWidth',1);**

**for i = 1:1:3**

**for j = 1:length(X\_w)**

**C(j+p\*length(X\_w)) = X\_w(j);**

**end**

**p = p+1;**

**end**

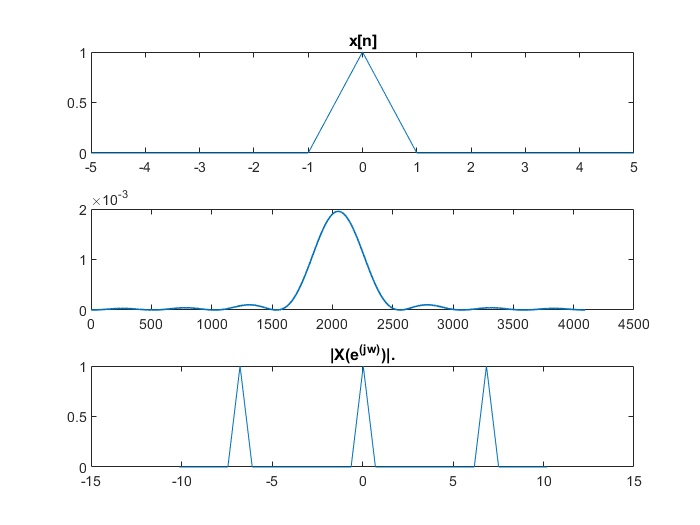
**A = max(F)/Fs \*2\*pi + 2\*pi**

**D = -A:2\*A/243:A-(20.42/243);**

**subplot(313),plot(D+(20.42/243),C);**

**title('|X(e^{(jw)})|.')**

**Output:**



**Problem 4.2:**

**MatLab Code:**

**clc**

**clear**

**close all**

**Fs = 48;**

**N = -10:0.001:10;**

**ustepp = zeros(size(N));**

**y = zeros(size(N));**

**for i = 1: length(N)**

**n = N(i);**

**y(i) = ustep(n-4)-ustep(n-6);**

**end**

**B = y + fliplr(y);**

**NFFT = 4096;**

**C = zeros(1,3\*length(B));**

**p = 0;**

**ff = (-0.5: 1/NFFT: 0.5-1/NFFT)\*Fs;**

**MAG\_X2 = ifftshift(abs(ifft(B,NFFT)));**

**figure**

**subplot(311), plot(N,B)**

**title('x[n]')**

**subplot(312),plot(ff,MAG\_X2,'LineWidth',1);xlim([-5 5])**

**for i = 1:1:3**

**for j = 1:length(B)**

**C(j+p\*length(B)) = B(j);**

**end**

**p = p+1;**

**end**

**A = max(N)/Fs \*2\*pi + 2\*pi**

**D = -A:2\*A/(3\*length(B)):A-2\*A/(3\*length(B));**

**subplot(313),plot(D+2\*A/(3\*length(B)),C);**

**title('|X(e^{(jw)})|.')**

**function y = ustep(t)**

**if t<0**

**y=0;**

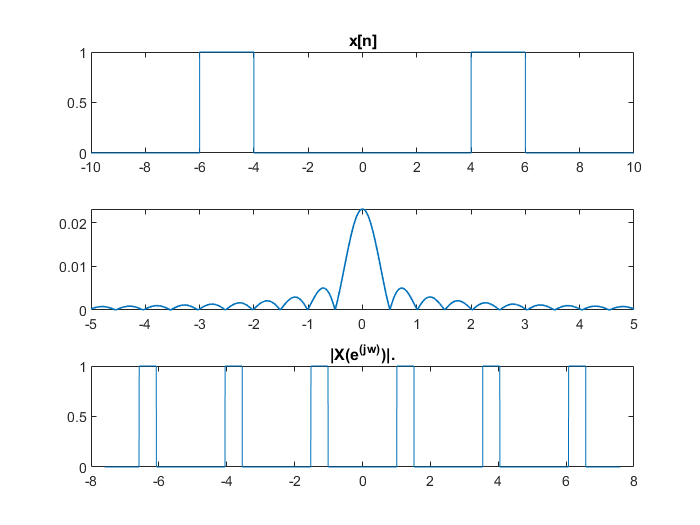
**else**

**y=1;**

**end**

**end**

**Output:**



**Problem 4.3:**

**Matlab Code:**

**clc**

**clear**

**close all**

**Fs = 64;**

**F=-5:1/Fs:5;**

**N = -10:0.001:10;**

**ustepp = zeros(size(N));**

**y = zeros(size(N));**

**for i = 1: length(N)**

**n = N(i);**

**y(i) = u\_step(n-4)-u\_step(n-6)+ u\_ramp(n-6)-2\*u\_ramp(n-7)+u\_ramp(n-8);**

**end**

**B = y + fliplr(y);**

**NFFT = 4096;**

**C = zeros(1,3\*length(B));**

**p = 0;**

**ff = (-0.5: 1/NFFT: 0.5-1/NFFT)\*Fs;**

**MAG\_X2 = ifftshift(abs(ifft(B,NFFT)));**

**figure**

**subplot(311), plot(N,B)**

**title('x[n]')**

**subplot(312),plot(ff,MAG\_X2,'LineWidth',1);xlim([-5 5])**

**for i = 1:1:3**

**for j = 1:length(B)**

**C(j+p\*length(B)) = B(j);**

**end**

**p = p+1;**

**end**

**A = max(N)/Fs \*2\*pi + 2\*pi**

**D = -A:2\*A/(3\*length(B)):A-2\*A/(3\*length(B));**

**subplot(313),plot(D+2\*A/(3\*length(B)),C);**

**title('|X(e^{(jw)})|.')**

**function y = u\_step(t)**

**if t<0**

**y=0;**

**else**

**y=1;**

**end**

**end**

**function z = u\_ramp(t)**

**if t>0**

**z=t;**

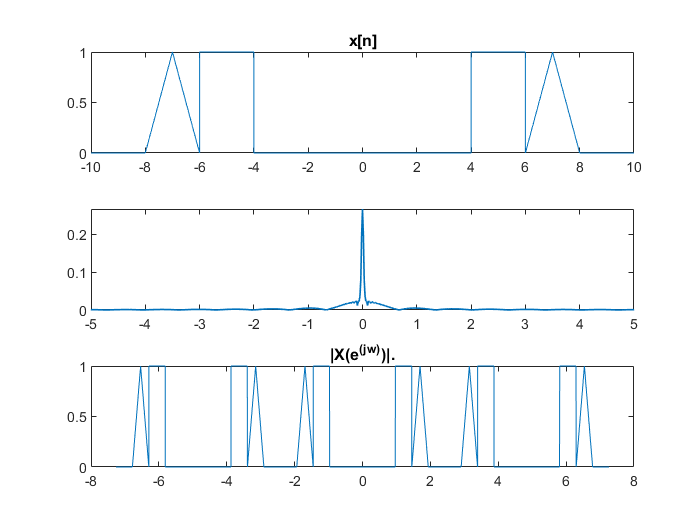
**else**

**z=0;**

**end**

**end**

**Output:**



**Problem 4.4:**

**Matlab Code:**

**clc**

**clear**

**close all**

**Fs = 32;**

**F=-5:1/Fs:5;**

**n=-5:0.01:5**

**B= heaviside(n+4)-heaviside(n+2)+triangularPulse(-2, -1, 0, n)....**

**+triangularPulse(0, 1, 2, n)+ heaviside(n-2)-heaviside(n-4);**

**NFFT = 4096;**

**C = zeros(1,3\*length(B));**

**p = 0;**

**ff = (-0.5: 1/NFFT: 0.5-1/NFFT)\*Fs;**

**MAG\_X2 = ifftshift(abs(ifft(B,NFFT)));**

**figure**

**subplot(311), plot(n,B)**

**title('x[n]')**

**subplot(312),plot(ff,MAG\_X2,'LineWidth',1);xlim([-5 5])**

**title('|X(e^{(jw)})|.')**

**for i = 1:1:3**

**for j = 1:length(B)**

**C(j+p\*length(B)) = B(j);**

**end**

**p = p+1;**

**end**

**A = max(n)/Fs \*2\*pi + 2\*pi**

**D = -A:2\*A/(3\*length(B)):A-2\*A/(3\*length(B));**

**subplot(313),plot(D+2\*A/(3\*length(B)),C);**

**Output:**

