KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
COLLEGE OF COMPUTER SCIENCES & ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

COE-342 – Data and Computer Communication

Programming Assignment #1

Due Date: Saturday November 9th, 2002

Q.1. Given a square wave signal of period T = 1 ms and peak amplitude A = 1V. One period of the signal is shown below:

$$s(t) = \begin{cases} 1 & -T/4 \le t \le T/4 \\ 0 & T/4 < t < T/2 \\ 0 & -T/2 < t < -T/4 \end{cases}$$

(i) You are required to calculate the Fourier Series of such a signal and determine its frequency components. Limit yourself to the first 6 frequency components.

(ii) Using Matlab, plot the signal s(t) and plot its frequency spectrum.

(iii) The above signal is sent through different media with different bandwidths. If the media have the following bandwidths, what effects does this have on the signal at the receiver?

1. BW = 20 KHz,
2. BW = 10 KHz.
3. BW = 4.5 KHz.

For each case, reconstruct the signal and plot it using Matlab. What do you conclude?

<u>Solution</u>

i) The Fourier Series expansion for s(t) is given by

$$s(t) = \frac{A_0}{2} + \sum_{n=1,2}^{\infty} A_n \cos(2\pi nft) + B_n \sin(2\pi nft)$$

where the coefficients are computed as

$$A_0 = \frac{2}{T}\int_{-T/2}^{T/2} s(t)dt$$

$$A_n = \frac{2}{T}\int_{-T/2}^{T/2} s(t)\cos(2\pi nft)dt \qquad n = 1,2,...$$

$$B_n = \frac{2}{T}\int_{-T/2}^{T/2} s(t)\sin(2\pi nft)dt \qquad n = 1,2,...$$

* The coefficient A_0 is equal to:

$$A_0 = \frac{2}{T}\int_{-T/2}^{T/2} s(t)dt = \frac{2A}{T}\times\int_{-T/4}^{T/4} dt = \frac{2A}{T}t\Big|_{t=-T/4}^{t=T/4} = \frac{2A}{T}\frac{T}{2} = A$$

* The coefficients A_n (n=1,2, 3, …) are computed as

$$A_n = \frac{2}{T}\int_{-T/2}^{T/2} s(t)\cos(2\pi nft)dt = \frac{2A}{T}\times\int_{-T/4}^{T/4}\cos(2\pi nft)dt = \frac{2A}{T}\times\frac{1}{2\pi nf}\sin(2\pi nft)\Big|_{t=-T/4}^{t=T/4}$$

$$= \frac{A}{n\pi}\sin(\pi n/2) = \begin{cases} 0 & n = 2,4,6,... \\ \frac{2A}{n\pi}(-1)^{(n-1)/2} & n = 1,3,5,... \end{cases}$$

* The coefficients B_n (n=1,2, 3, …) are computed as

$$B_n = \frac{2A}{T}\int_{-T/2}^{T/2}\sin(2\pi nft)dt = \frac{2A}{T}\times\int_{-T/4}^{T/4}\sin(2\pi nft)dt = \frac{2A}{T}\times\frac{-1}{2\pi nf}\cos(2\pi nft)\Big|_{t=-T/4}^{t=T/4} =$$

$$B_n = \frac{-A}{\pi n}\left[\cos(\pi n) - \cos(\pi n)\right] = 0 \text{ for all n}$$

The final Fourier Series expansion for s(t) is given by

$$s(t) = \frac{A}{2} + \frac{2A}{\pi}\sum_{n=1,3,5}^{\infty}(-1)^{(n-1)/2}\frac{\cos(2\pi nft)}{n} \text{, or}$$

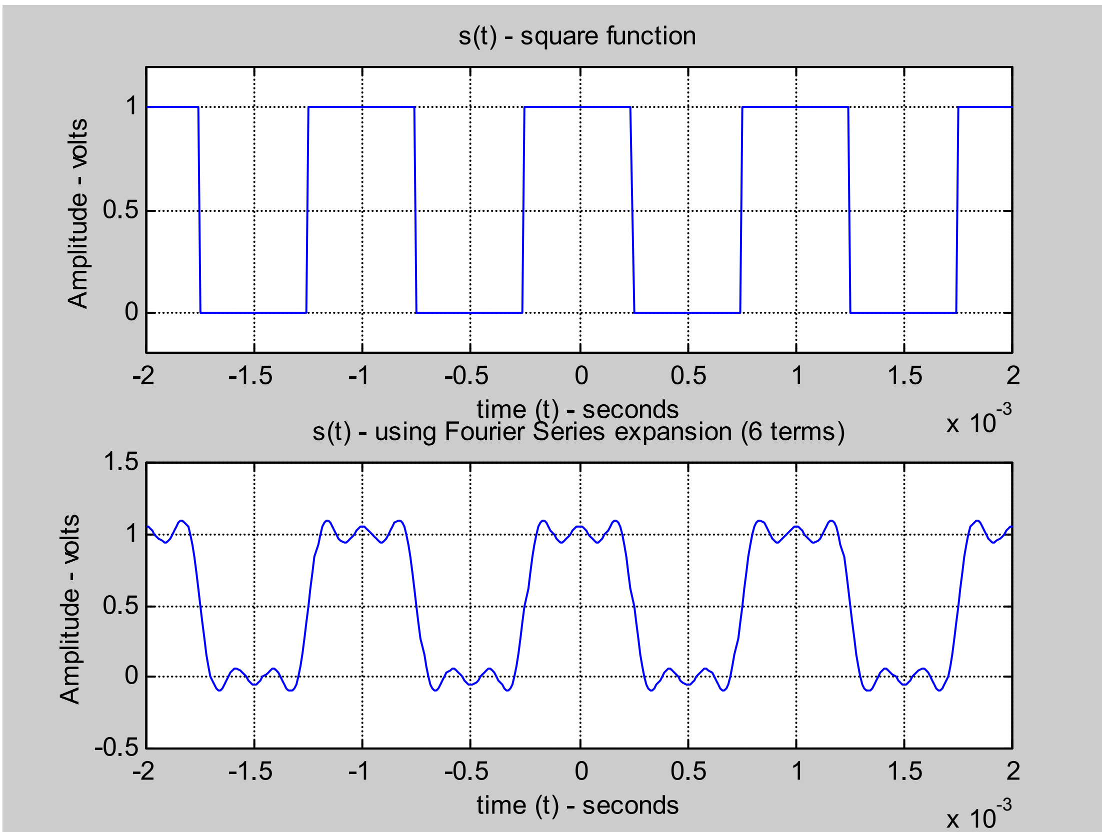
For A = 1 volts, T = 0.001 second (or f = 1000 Hz), s(t) is written as

$$s(t) = \frac{1}{2} + \frac{2}{\pi}\sum_{n=1,3,5}^{\infty}(-1)^{(n-1)/2}\frac{\cos(2000\pi nt)}{n}$$

Limiting the expression to first 6 components

$$\hat{s}(t) = \frac{1}{2} + \frac{2}{\pi}\left[\frac{\cos(2000\pi t)}{1} + \frac{-\cos(6000\pi t)}{3} + \frac{\cos(10000\pi t)}{5} + \frac{-\cos(14000\pi t)}{7} + \frac{\cos(18000\pi t)}{9} + \frac{-\cos(22000\pi t)}{11}\right]$$

Graph for s(t) and its Fourier Series Expansion:



Matlab code for part i)

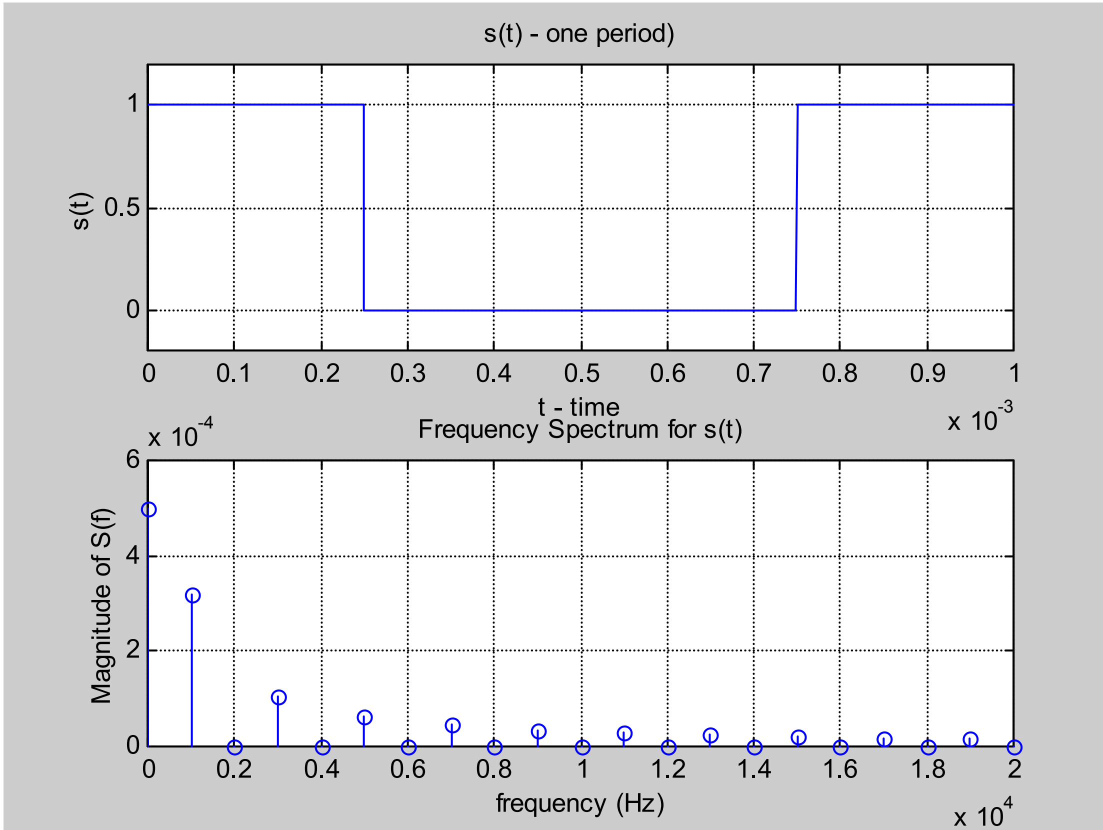
```
clear all

T = 0.001; % period of signal
A = 1; % amplitude of signal

dt = T/100;     % step in time domain
t  = -2*T:dt:2*T;% define the time axis - up to 4 time periods
%
% define the function s(t) - one period is used
s = A*(1/2 + 1/2*square(2*pi*(t+T/4)./T));


%
% Fourier Series expansion is given by:
%
%                infinity
%  s(t) = a_0/2 + sum [ a_k * cos(2*pi*k*f*t) + b_k * sin(2*pi*k*f*t)]
%                 k=1
%


aa_0            = A;
max_k           = 100;
k               = 1:max_k;
aa_k            = 2*A./(pi*k) .* (-1).^((k-1)/2);
aa_k(2:2:max_k) = 0;

bb_k            = zeros(size(k));

f  = 1/T;
ss = A/2 *ones(size(t));
for i=1:6
   ss = ss + aa_k(i)*cos(2*pi*i*f*t) + bb_k(i)*sin(2*pi*f*t);
end

figure(1);
subplot(2,1,1);
plot(t, s);
title('s(t) - square function');
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
axis([-2*T 2*T -0.2 1.2]);
grid

subplot(2,1,2);
plot(t, ss);
title('s(t) - using Fourier Series expansion (6 terms)');
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
grid
```

ii) Plotting the signal s(t) and its spectrum

To obtain the spectrum function we will use the FFT routine in Matlab.

```
% ProgAssign01_COE021342_Q1_ii
clear all
%
% N defines the numbe of samples in the time domain - and the number of
% FFT points
% Since N has be even, N is made a multiple of 2
M = 10;  % The higher the value of M (and consequently N) the better
N = 2^M;% the approximation of the FFT to the Fourier Series expansion

T = 0.001; % period of signal
A = 1; % amplitude of signal

n  = 0:1:N-1;
dt = T/N; % step in time domain
t  = dt*n; % define the time axis
%
% define the function s(t) - one period is used
s = A*(1/2 + 1/2*square(2*pi*(t+T/4)./T));

%
% fft definition includes averaging over N

S = fft(s)/N;

figure(1);
subplot(2,1,1);
plot(t, s);
title('s(t) - one period)');
xlabel('t - time');
ylabel('s(t)');
axis([0 T -0.2 1.2]);
grid
subplot(2,1,2);
f = n./T;
plot(f(1:N/2), abs(T*S(1:N/2)));
title('Frequency Spectrum for s(t)');
xlabel('frequency (Hz)');
ylabel('Magnitude of S(f)');
axis([0 20./T 0 1.2*max(abs(T*S))]);
grid
```

Graph for s(t) and its spectrum:



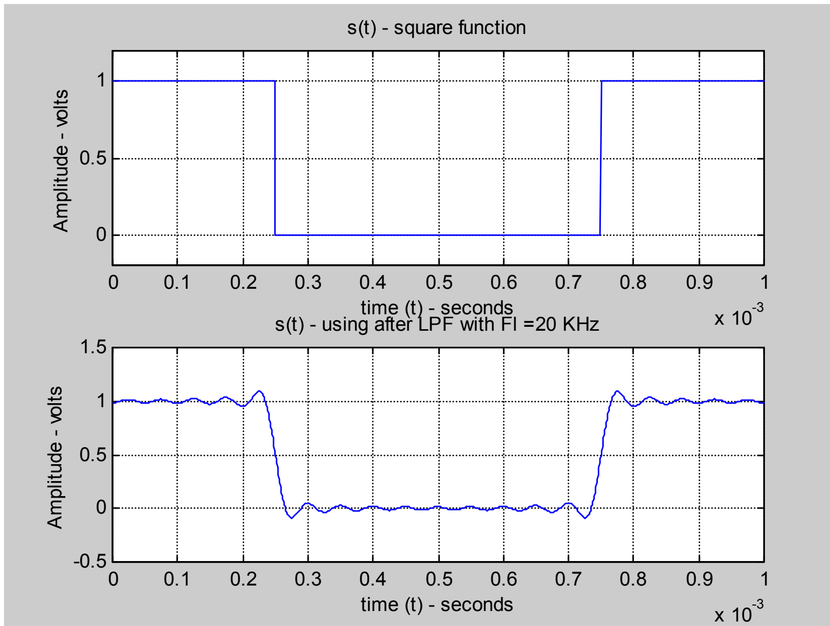
iii) Passing signal through low pass filters

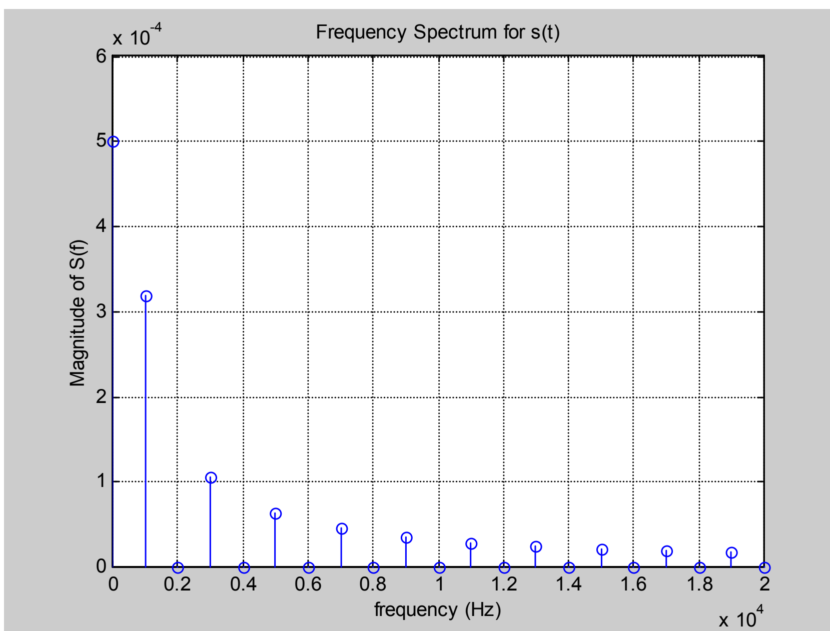
The following matlab code computes the filter output for any cut-off frequency. The variable "Low_Pass_Filter_BW" specifies the cut-off frequency for the filter. For example for part (1) where BW is equal to 20 KHz, the variable "Low_Pass_Filter_BW" should be set to "20000" Hz and so on.

```
% ProgAssign01_coe02342_Q1_1_iii.m
clear all
%
% N defines the numbe of samples in the time domain - and the number of
% FFT points
% Since N has be even, N is made a multiple of 2
M = 10;  % The higher the value of M (and consequently N) the better
N = 2^M;% the approximation of the FFT to the Fourier Series expansion

n = 0:1:N-1;
T = 0.001; % period of signal
A = 1; % amplitude of signal

dt = T/N;     % step in time domain
t  = 0:dt:T;% define the time axis - up to 4 time periods
%
% define the function s(t) - one period is used
s = A*(1/2 + 1/2*square(2*pi*(t+T/4)./T));


%
% Fourier Series expansion is given by:
%
%               infinity
%  s(t) = a_0/2 + sum [ a_k * cos(2*pi*k*f*t) + b_k * sin(2*pi*k*f*t)]
%                k=1
%


aa_0            = A;
max_k           = 100;
k               = 1:max_k;
aa_k            = 2*A./(pi*k) .* (-1).^((k-1)/2);
aa_k(2:2:max_k) = 0;

bb_k            = zeros(size(k));

f_fundamental      = 1/T;
Low_Pass_Filter_BW = 20000; % Hz
h  = floor(Low_Pass_Filter_BW/ f_fundamental); % decide number of harmonics
ss = A/2 *ones(size(t));
for i=1:h
   ss = ss + aa_k(i)*cos(2*pi*i*f_fundamental*t) + ...
             bb_k(i)*sin(2*pi*f_fundamental*t);
end
```

```
% ProgAssign01_coe02342_Q1_1_iii.m - continued

figure(1);
subplot(2,1,1);
plot(t, s);
title('s(t) - square function');
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
axis([0 T -0.2 1.2]);
grid

subplot(2,1,2);
plot(t, ss);
TitleStr = ['s(t) - using after LPF with Fl =' num2str(Low_Pass_Filter_BW/1000) '
KHz'];
title(TitleStr);
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
grid

% compute spectrum function
S = fft(ss)/N;

freq = n./T;
figure(2);
stem(freq(1:N/2), abs(T*S(1:N/2)),'');
title('Frequency Spectrum for s(t)');
xlabel('frequency (Hz)');
ylabel('Magnitude of S(f)');
axis([0 20./T 0 1.2*max(abs(T*S))]);
grid
```

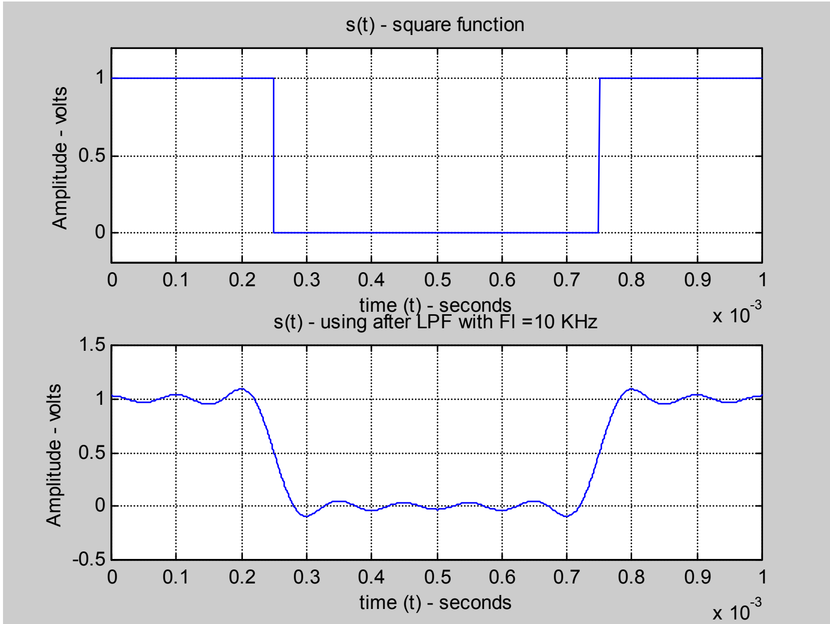
1) For BW = 20 KHz:

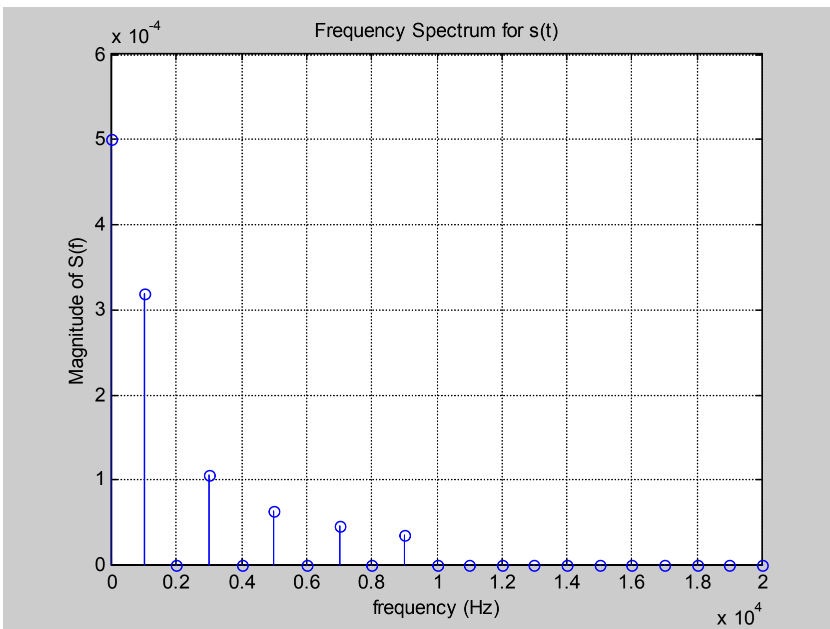




Note the existence of frequency components up to 19 KHz. The filtered signal still very similar to original s(t) since BW = 20 KHz passes most of required components to represent s(t). As BW is decreased, the filter will be rejecting more significant components of s(t) and the filter output will start to differ from the original signal.

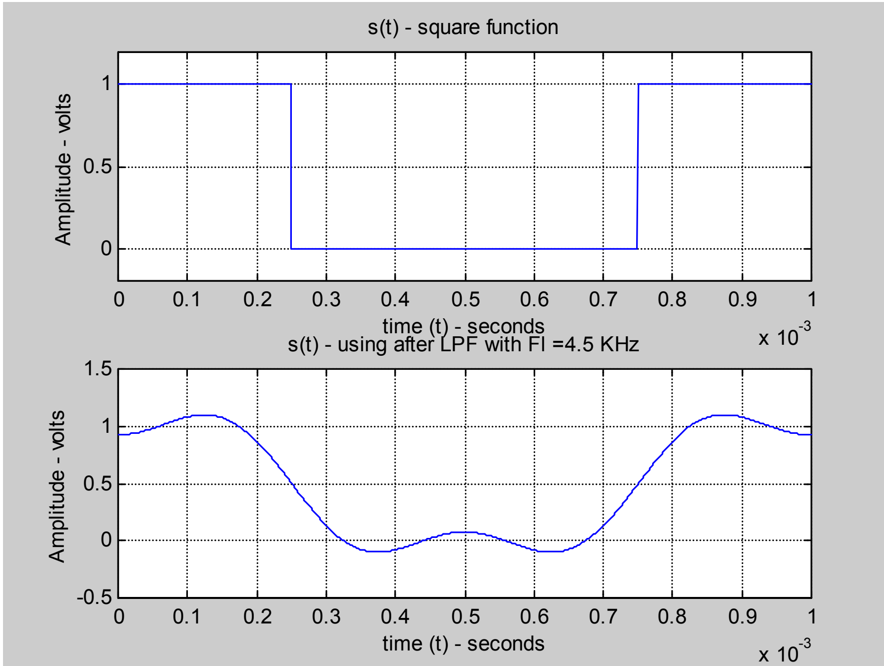
2) BW = 10 KHz

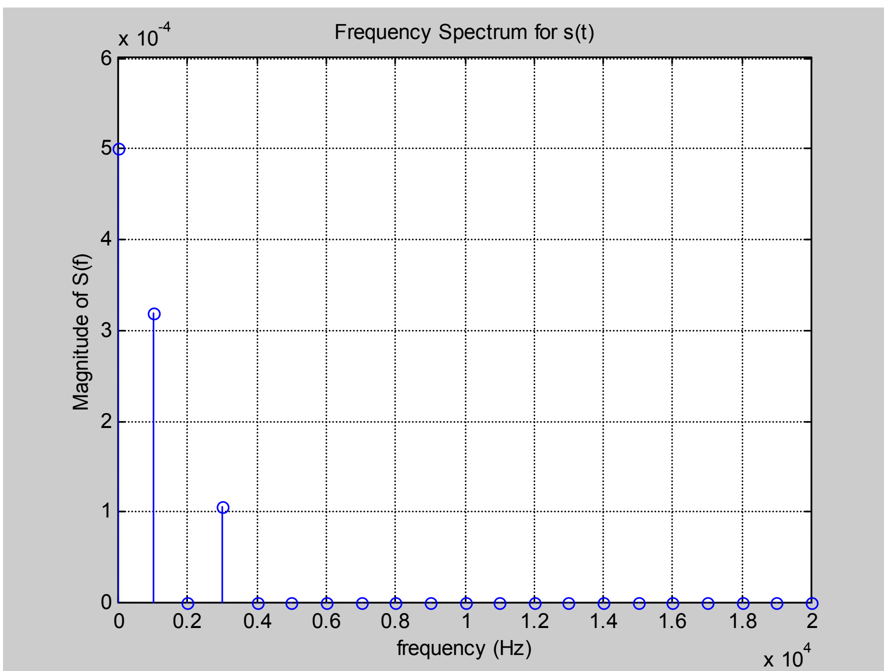




Note frequency components now exist up to 9 KHz.

3) BW = 4500 Hz





Note in addition to the DC component, only two harmonics exist in the output signal (at 1xf and 3xf) only. The filter output is shown above is not very close to original signal s(t).

Q.2. Consider the full-wave rectified cosine signal shown in Figure 3.15. Assume that the period of the signal T = 1μs and A = 1V.

> **(i)** You are required to calculate the Fourier Series of the signal and determine its frequency components. Limit yourself to the first 6 frequency components.
>
> **(ii)** If you were required to transmit this signal across a transmission media, what bandwidth would you recommend? Draw the received signal using the recommended bandwidth.

Solution

i) For the full-wave rectified cosine signal, the signal s(t) is given by

$$s(t) = A\left|\cos(\frac{2\pi}{T}t)\right| = \begin{cases} A\cos(\frac{2\pi}{T}t) & -T/4 < t < T/4 \\ -A\cos(\frac{2\pi}{T}t) & T/4 < t < 3T/4 \end{cases}$$

The Fourier Series expansion for s(t) is given by

$$s(t) = \frac{A_0}{2} + \sum_{n=1,2}^{\infty} A_n \cos(2\pi nft) + B_n \sin(2\pi nft)$$

where the coefficients are computed as

$$A_0 = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)dt$$

$$A_n = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)\cos(2\pi nft)dt \qquad n = 1,2,...$$

$$B_n = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)\sin(2\pi nft)dt \qquad n = 1,2,...$$

* The coefficient A_0 is equal to:

$$A_0 = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)dt = \frac{4A}{T}\times\int_{-T/4}^{T/4} \cos(\frac{2\pi}{T}t)dt = \frac{8A}{T}\times\int_{0}^{T/4} \cos(\frac{2\pi}{T}t)dt = \frac{8A}{T}\times\frac{T}{2\pi}\sin(\frac{2\pi}{T}t)\Bigg|_{t=0}^{t=T/4}$$

$$A_0 = \frac{4A}{\pi}\times\left[\sin(\frac{\pi}{2}) - 0\right] = \frac{4A}{\pi}$$

* The coefficients A_n (n=1,2, 3, …) are computed as

$$A_n = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)\cos(2\pi n f t)dt = \frac{2A}{T}\times\left[\int_{-T/4}^{T/4}\cos(2\pi f t)\cos(2\pi n f t)dt - \int_{T/4}^{3T/4}\cos(2\pi f t)\cos(2\pi n f t)dt\right]$$

$$= \frac{-A}{\pi(n^2-1)}\left[3\cos(\frac{\pi}{2}n) + \cos(3\frac{\pi}{2}n)\right] = \begin{cases} 0 & n = 3,5,... \\ \dfrac{4A}{\pi(n^2-1)}(-1)^{1+n/2} & n = 2,4,6,... \end{cases}$$

Note that the above expression is not valid for n equal to 1. For n equal to 1, A_1 is equal to

$$A_1 = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)\cos(2\pi\times 1\times ft)dt = \frac{2A}{T}\times\left[\int_{-T/4}^{T/4}\cos(2\pi f t)\cos(2\pi f t)dt - \int_{T/4}^{3T/4}\cos(2\pi f t)\cos(2\pi f t)dt\right]$$

$$A_1 = 0$$

since $\left|\cos(2\pi f t)\right|^2$ has the same area under (-T/4, T/4) and (T/4, 3T/4).

* The coefficients B_n (n=1,2, 3, …) are computed as

$$B_n = \frac{2}{T}\int_{-T/4}^{3T/4} s(t)\cos(2\pi n f t)dt = \frac{2A}{T}\times\left[\int_{-T/4}^{T/4}\cos(2\pi f t)\sin(2\pi n f t)dt - \int_{T/4}^{3T/4}\cos(2\pi f t)\sin(2\pi n f t)dt\right]$$

$$= \frac{-A}{\pi(n^2-1)}\left[\sin(3\frac{\pi}{2}n) + \sin(\frac{\pi}{2}n)\right] = 0 \text{ for all n}$$

The final Fourier Series expansion for s(t) is given by

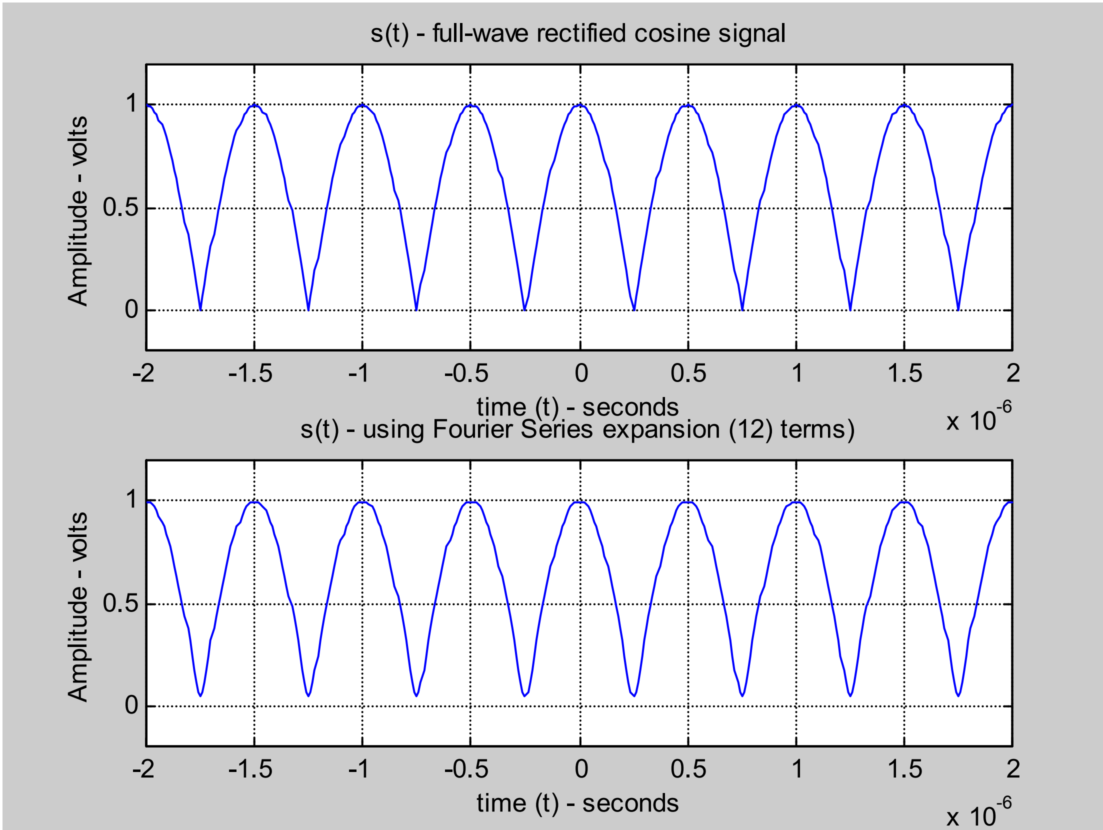
$$s(t) = \frac{4A}{\pi} + \frac{4A}{\pi}\sum_{n=2,4,6}^{\infty}(-1)^{1+n/2}\frac{\cos(2\pi n f t)}{\left(n^2-1\right)} \text{, or}$$

For A = 1 volts, T = 1μsec (or f = 10^6 Hz), s(t) is written as

$$s(t) = \frac{4}{\pi} + \frac{4}{\pi}\sum_{n=2,4,6}^{\infty}(-1)^{1+n/2}\frac{\cos(10^6\pi n t)}{\left(n^2-1\right)}$$

Limiting the expression to first 6 components

$$\hat{s}(t) = \frac{4}{\pi}\left[1 + \frac{\cos(2\times 10^6\pi t)}{3} + \frac{-\cos(4\times 10^6\pi t)}{15} + \frac{\cos(6\times 10^6\pi t)}{35} + \frac{-\cos(8\times 10^6\pi t)}{63} + \frac{\cos(10\times 10^6\pi t)}{99} + \frac{-\cos(12\times 10^6\pi t)}{143}\right]$$

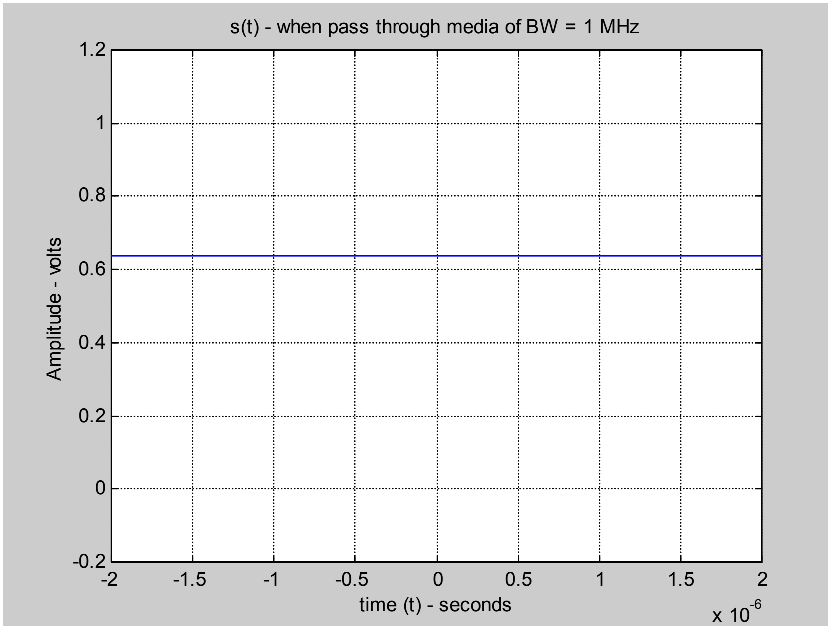

s(t) - full-wave rectified cosine signal
Amplitude - volts
1
0.5
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
time (t) - seconds
x 10^-6
s(t) - using Fourier Series expansion (12) terms)
Amplitude - volts
1
0.5
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
time (t) - seconds
x 10^-6

The matlab code used to plot the function and its Fourier representation is given below:

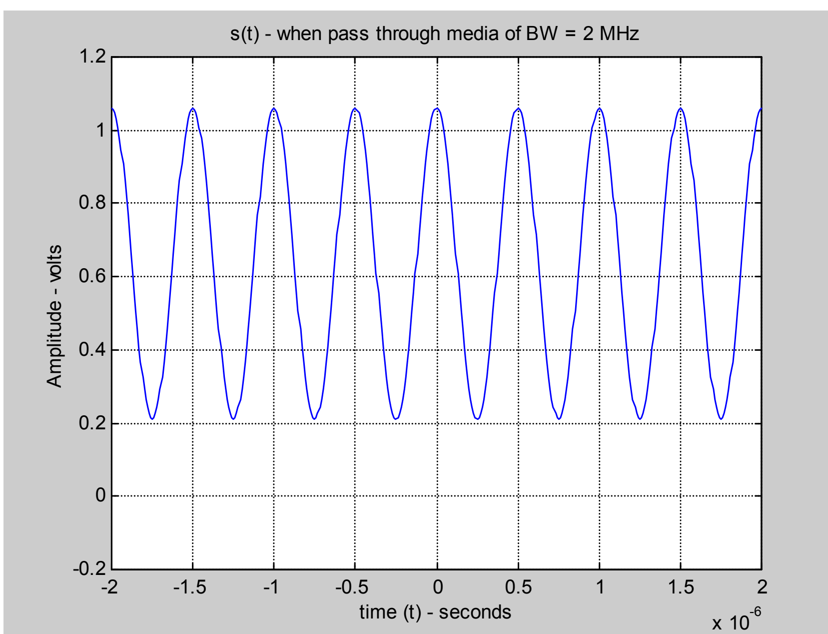
```
% ProgAssign01_coe021342_Q2_i
clear all

T = 1.0e-6; % period of signal
A = 1; % amplitude of signal

dt = T/100;      % step in time domain
t  = -2*T:dt:2*T;% define the time axis - up to 4 time periods
%
% define the function s(t) - one period is used
s = A*abs(cos(2*pi*t./T));


%
% Fourier Series expansion is given by:
%
%                infinity
%  s(t) = a_0/2 + sum [ a_k * cos(2*pi*k*f*t) + b_k * sin(2*pi*k*f*t)]
%                 k=1
%


aa_0            = 4*A/pi;
NoOfTerms       = 12;
k               = 1:NoOfTerms;
aa_k            = 4*A./(pi*(k.*k-1)) .* (-1).^(1+k./2);
aa_k(1:2:NoOfTerms) = 0;   % odd aa_k coeffients are zeros

bb_k            = zeros(size(k));

f  = 1/T;
ss = aa_0/2 *ones(size(t));

for i=1:NoOfTerms
   ss = ss + aa_k(i)*cos(2*pi*i*f*t) + bb_k(i)*sin(2*pi*f*t);
end

figure(1);
subplot(2,1,1);
plot(t, s);
title('s(t) - full-wave rectified cosine signal');
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
axis([-2*T 2*T -0.2 1.2]);
grid

subplot(2,1,2);
plot(t, ss);
TitleStr = ['s(t) - using Fourier Series expansion (' num2str(NoOfTerms) ') terms)'];
title(TitleStr);
xlabel('time (t) - seconds');
ylabel('Amplitude - volts');
axis([-2*T 2*T -0.2 1.2]);
grid
```

ii) Choosing a suitable transmission media

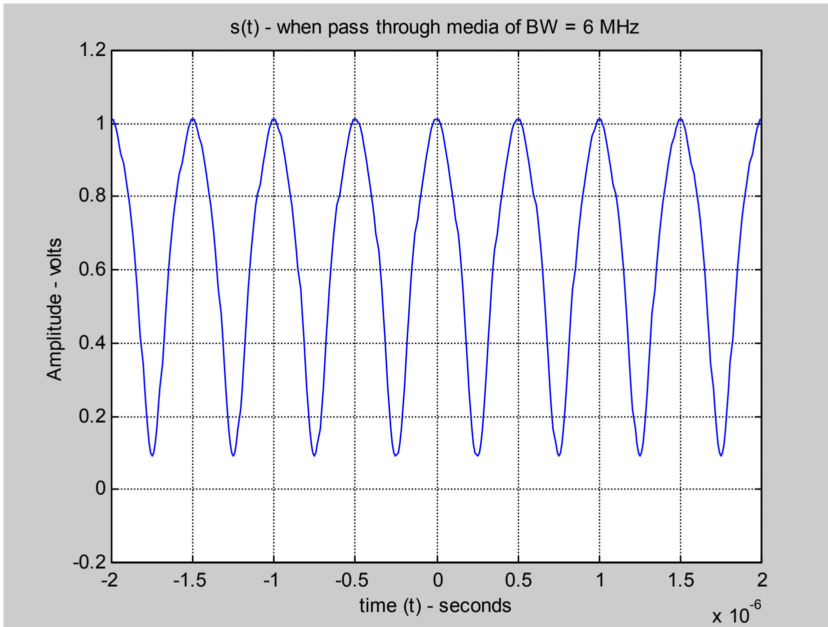
When media bandwidth is small (1 MHz) for example, only the DC component is propagated – higher harmonics are attenuated or rejected. The output signal is as shown below for BW = 1 MHz.



When media bandwidth is increased to 2 MHz; the DC component and the first harmonic at 2Xf is propagated – the output signal now looks similar to the input signal s(t) but not yet close (note the lowest amplitude is about 0.2 volts instead of 0)



For a media bandwidth of 6 MHz, the output signal is now more similar to original signal s(t).



One can show that the propagated signal has about 99% of the original power of s(t) when a bandwidth of 4 MHz - i.e. first two harmonics at 2Xf and 4Xf are passed. This percentage increases to 99.9% when the media allows the first 6 harmonics to pass: 2Xf, 4Xf, 6Xf, 8Xf, 10Xf, and 12Xf – i.e. when the media bandwidth is equal to 12 MHz.

For part ii) of Q2 a bandwidth higher than 4 MHz seems to be suitable.

Q.3. Given the triangular pulse shown in Figure 3.16, find its Fourier transform and then plot it using Matlab.

Solution:

The triangular pulse s(t) can be written as

$$s(t) = \begin{cases} A\left(\frac{t}{\tau} + 1\right) & -\tau < t < 0 \\ A\left(\frac{-t}{\tau} + 1\right) & 0 < t < \tau \end{cases}$$

The Fourier transform for s(t), S(f), is given by

$$S(f) = \int_{-\infty}^{+\infty} s(t)\exp(-j2\pi f t)dt$$

Therefore S(f) is computed as

$$S(f) = \int_{-\infty}^{+\infty} s(t)\exp(-j2\pi f t)dt = \int_{-\tau}^{0} A\left(\frac{t}{\tau}+1\right)\exp(-j2\pi f t)dt + \int_{0}^{\tau} A\left(\frac{-t}{\tau}+1\right)\exp(-j2\pi f t)dt$$

Let $I_1 = \int_{-\tau}^{0} A\left(\frac{t}{\tau}+1\right)\exp(-j2\pi f t)dt$ while $I_2 = \int_{0}^{\tau} A\left(\frac{-t}{\tau}+1\right)\exp(-j2\pi f t)dt$

$$I_1 = \int_{-\tau}^{0} A\left(\frac{t}{\tau}+1\right)\exp(-j2\pi f t)dt = \int_{-\tau}^{0} \frac{A}{\tau} t\exp(-j2\pi f t)dt + \int_{-\tau}^{0} A\exp(-j2\pi f t)dt$$

$$= \frac{A}{\tau}\left[\frac{t}{(-j2\pi f)}\exp(-j2\pi f t) - \frac{1}{(-j2\pi f)^2}\exp(-j2\pi f t)\right]\Bigg|_{t=-\tau}^{t=0} + \left[\frac{A}{(-j2\pi f)}\exp(-j2\pi f t)\right]\Bigg|_{t=-\tau}^{t=0}$$

$$= \frac{A}{\tau}\left[\left(0 - \frac{1}{(-j2\pi f)^2}\right) - \left(\frac{-\tau}{(-j2\pi f)}\exp(j2\pi f\tau) - \frac{1}{(-j2\pi f)^2}\exp(j2\pi f\tau)\right)\right] + \left[\frac{A}{(-j2\pi f)}(1-\exp(j2\pi f\tau))\right]$$

$$= \frac{jA}{(2\pi f)} + \frac{A/\tau}{(2\pi f)^2}[1-\exp(j2\pi f\tau)]$$

Similarly, the quantity I_2 is equal to

$$I_2 = \int_{0}^{\tau} A\left(\frac{-t}{\tau}+1\right)\exp(-j2\pi f t)dt = \int_{0}^{\tau} \frac{-A}{\tau} t\exp(-j2\pi f t)dt + \int_{0}^{\tau} A\exp(-j2\pi f t)dt$$

$$= \frac{-A}{\tau}\left[\frac{t}{(-j2\pi f)}\exp(-j2\pi ft) - \frac{1}{(-j2\pi f)^2}\exp(-j2\pi ft)\right]\Bigg|_{t=0}^{t=\tau} + \left[\frac{A}{(-j2\pi f)}\exp(-j2\pi ft)\right]\Bigg|_{t=0}^{t=\tau}$$

$$= \frac{-A}{\tau}\left[\left(\frac{\tau}{(-j2\pi f)}\exp(-j2\pi f\tau) - \frac{1}{(-j2\pi f)^2}\exp(-j2\pi f\tau)\right) - \left(0 - \frac{1}{(-j2\pi f)^2}\right)\right]$$
$$+ \left[\frac{A}{(-j2\pi f)}\left(\exp(-j2\pi f\tau) - 1\right)\right]$$

$$= \frac{-jA}{(2\pi f)} + \frac{A/\tau}{(2\pi f)^2}\left[1 - \exp(-j2\pi f\tau)\right]$$

Adding the two integrals

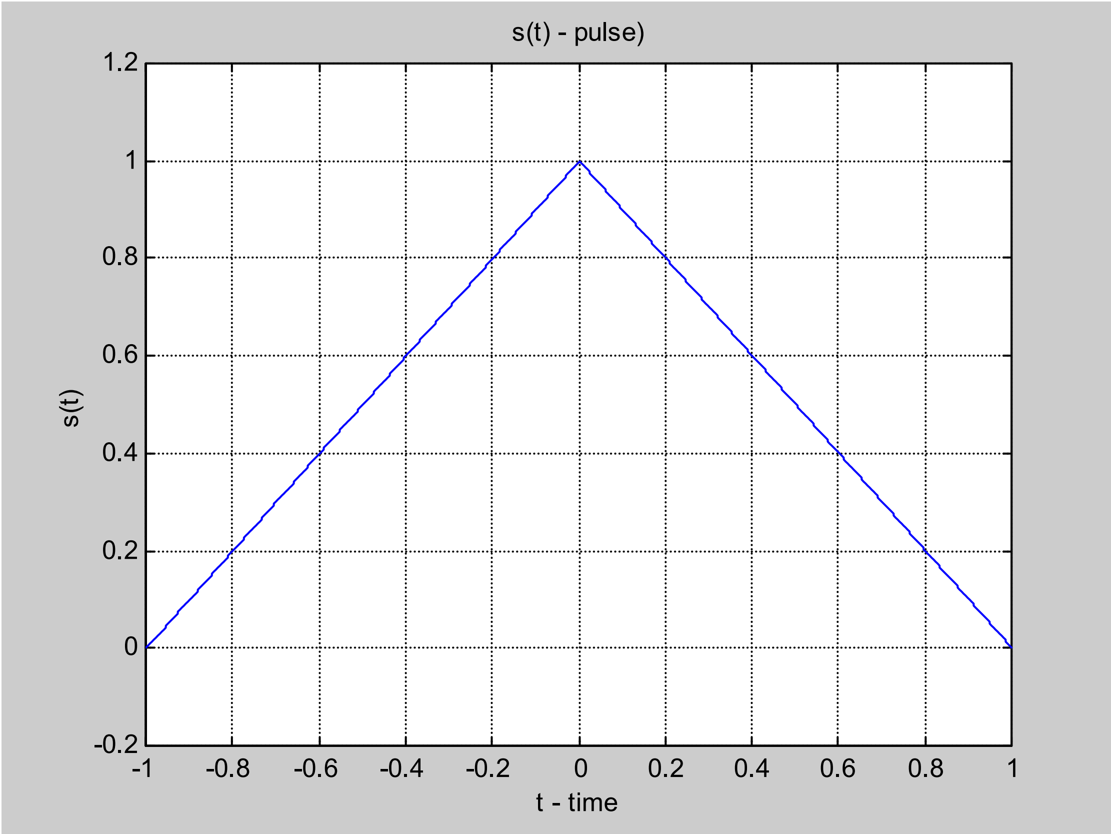
$$S(f) = I_1 + I_2 = \frac{A/\tau}{(2\pi f)^2}\left[1 - \exp(j2\pi f\tau)\right] + \frac{A/\tau}{(2\pi f)^2}\left[1 - \exp(-j2\pi f\tau)\right]$$

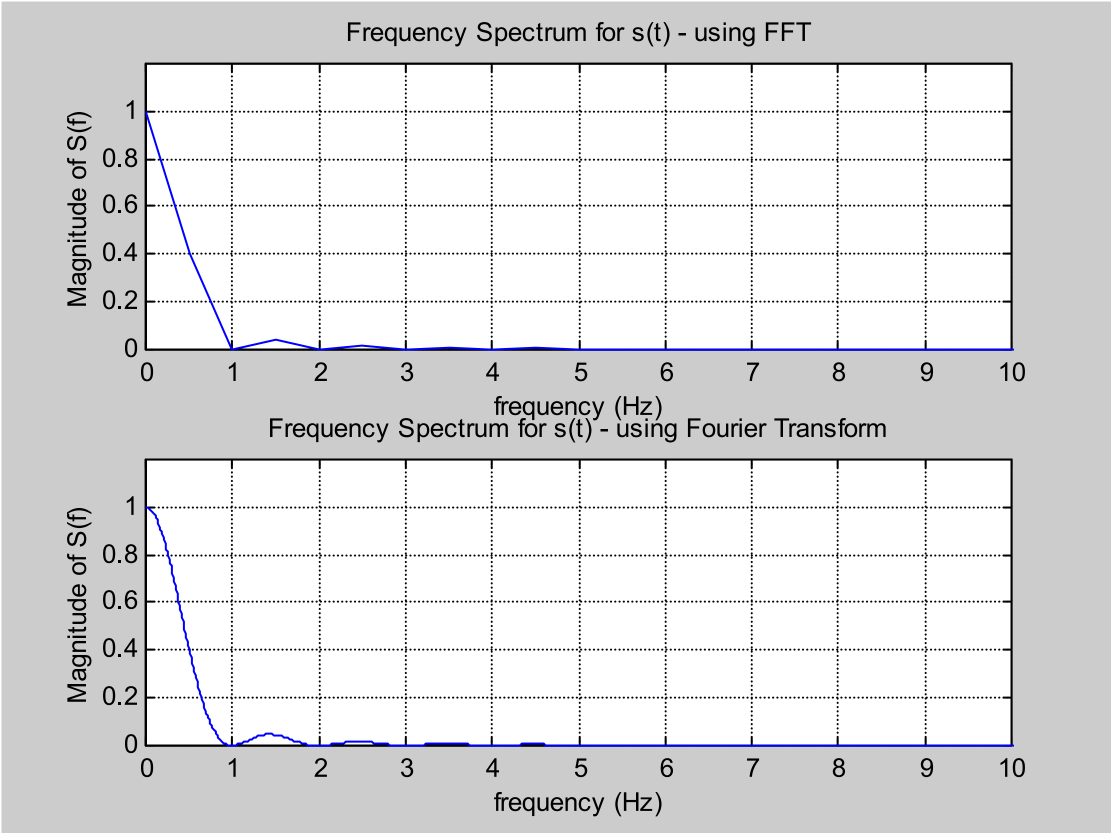
$$= \frac{2A/\tau}{(2\pi f)^2}\left[1 - \frac{\exp(j2\pi f\tau) + \exp(-j2\pi f\tau)}{2}\right] = \frac{A}{(2\pi^2 f^2 \tau)}\left[1 - \cos(2\pi f\tau)\right]$$

$$= \frac{A}{(\pi^2 f^2 \tau)}\sin^2(\pi f\tau) = A\tau\left[\frac{\sin(\pi f\tau)}{\pi f\tau}\right]^2$$

The following code plots the pulse s(t) and it frequency spectrum function S(f). The frequency spectrum function S(f) is graphed using the analytic equation obtained above and also using the approximation obtained through the use of the FFT routine. The graphs are generated for τ = 1 second, and A = 1 volt.

```
% ProgAssig01_coe021coe342_Q3_i
clear all
%
% N defines the numbe of samples in the time domain - and the number of
% FFT points
% Since N has be even, N is made a multiple of 2
M = 10;  % The higher the value of M (and consequently N) the better
N = 2^M;% the approximation of the FFT to the Fourier Series expansion

%
% Specification for the triangular pulse
Taw = 1; % Signal is defined on (-Taw, Taw)
A   = 1; % amplitude of signal
n  = 0:1:N-1;
dt = Taw/N; % step in time domain

tneg = -Taw: dt: 0;
tpos = dt:dt:Taw;
t  = [tneg tpos]; % define the time axis
%
% define the function s(t) - one period is used
s = A*(-abs(t)/Taw + 1) ;

%
% fft definition includes averaging over N

freq = 0:0.01:20;
S1   = A*Taw*(sin(pi*freq*Taw)./(pi*freq*Taw)).^2;

S2 = fft(s)/N;

figure(1);
plot(t, s);
title('s(t) - pulse)');
xlabel('t - time');
ylabel('s(t)');
axis([-Taw Taw -0.2 1.2]);
grid
figure(2);
subplot(2,1,1);
f = n./(2*Taw);  % 2Taw is the width of the pulse in time-domain
plot(f(1:N/2), abs(Taw*S2(1:N/2)));
title('Frequency Spectrum for s(t) - using FFT');
xlabel('frequency (Hz)');
ylabel('Magnitude of S(f)');
axis([0 10./Taw 0 1.2*max(abs(Taw*S2))]);
grid

subplot(2,1,2);
plot(freq, S1);
title('Frequency Spectrum for s(t) - using Fourier Transform');
xlabel('frequency (Hz)');
ylabel('Magnitude of S(f)');
axis([0 10./Taw 0 1.2*max(abs(S1))]);
grid
```


s(t) - pulse)
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
s(t)
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
t - time


Frequency Spectrum for s(t) - using FFT
Magnitude of S(f)
1
0.8
0.6
0.4
0.2
0
0
1
2
3
4
5
6
7
8
9
10
frequency (Hz)
Frequency Spectrum for s(t) - using Fourier Transform
Magnitude of S(f)
1
0.8
0.6
0.4
0.2
0
0
1
2
3
4
5
6
7
8
9
10
frequency (Hz)