Principles of Communication

Systems Lab

Lab 2, 20th August 2019

Pratyush Nandi (IMT2017518)

Answer to Q1

Note: For question 1 the signal u(t) = 2I[1,3](t) – 3I[2,4](t) and umf = u(-t),

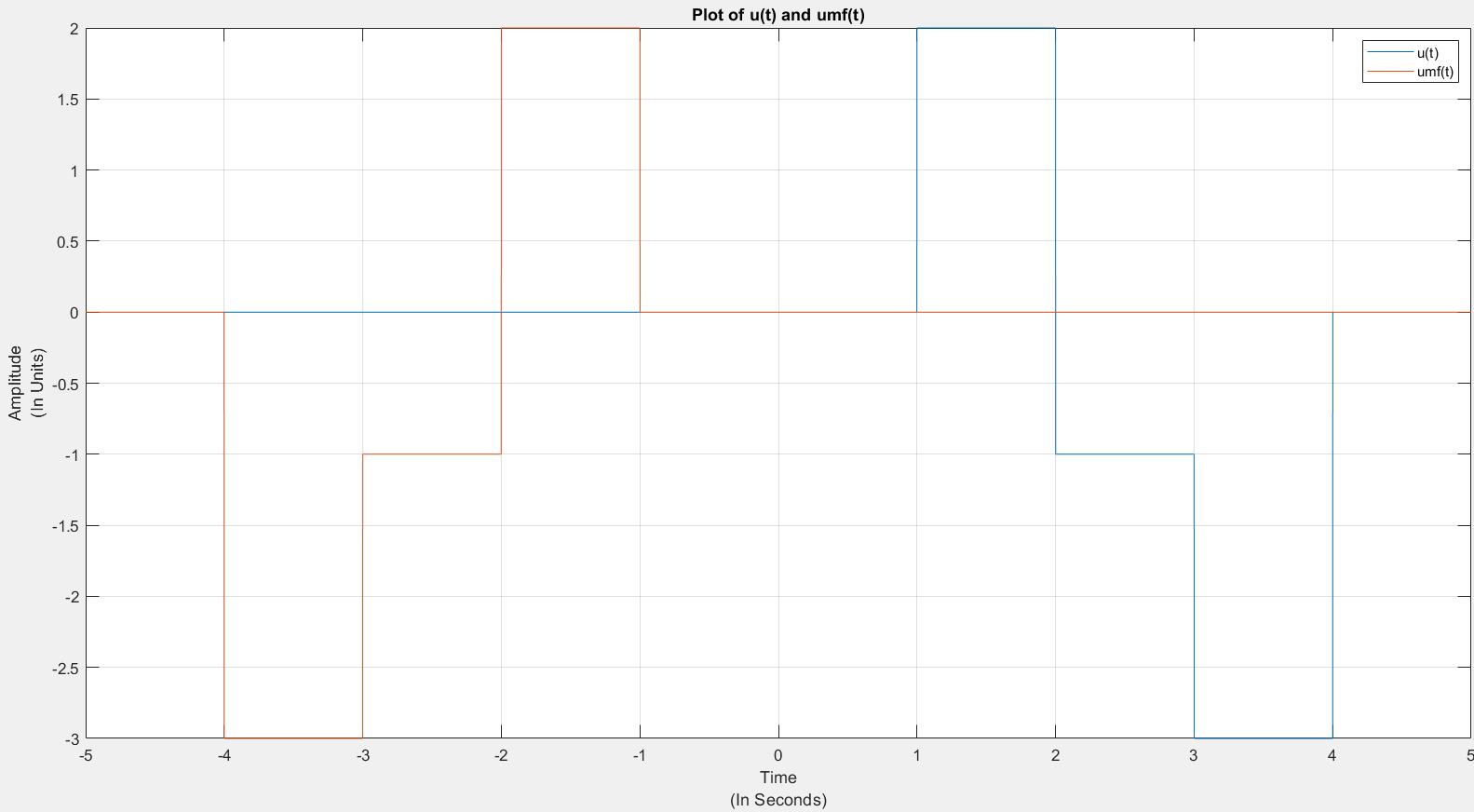
s(t) = u(t) + j v(t) where v(t) = ​I[-1,2](t) + 2I[0,1](t) and smf = s\*(-t),

s1(t) = s(t − t​o)x(e​jθ)​ here to = 2 and θ = (pi/4),

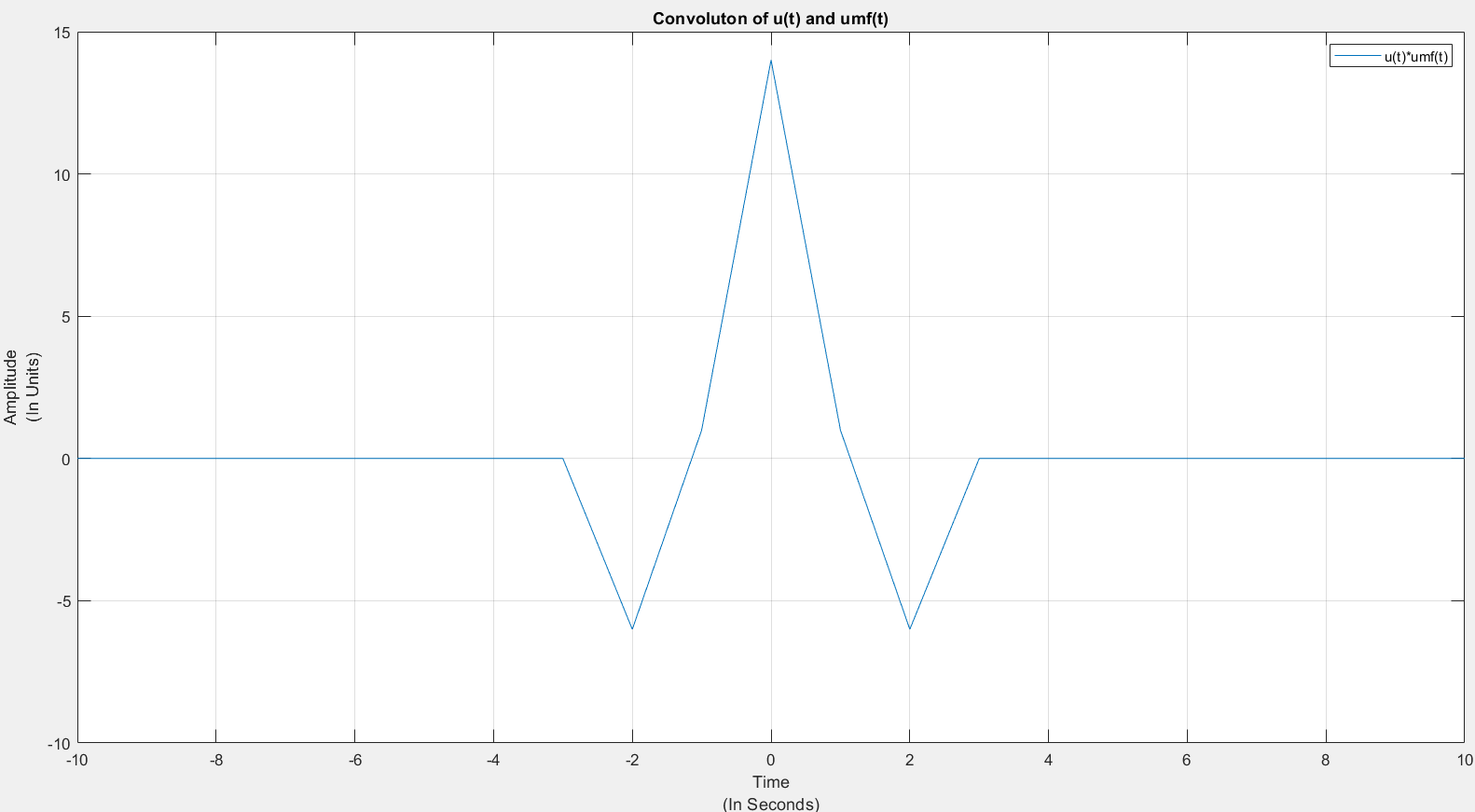
‘\*’ here means conjugate.

Unit time is in seconds and sampled at 1KHz frequency.

1(a):

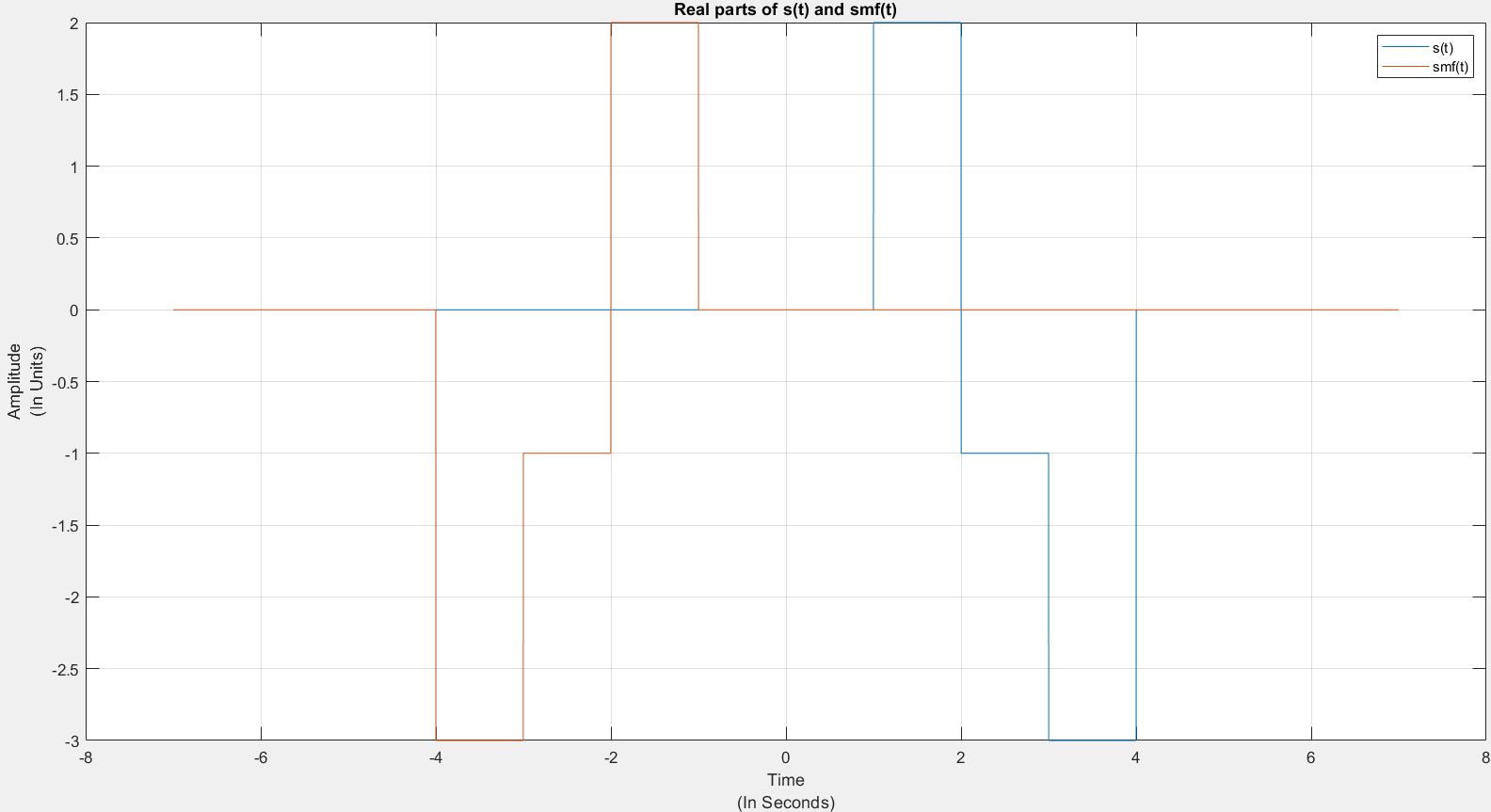


1(b):

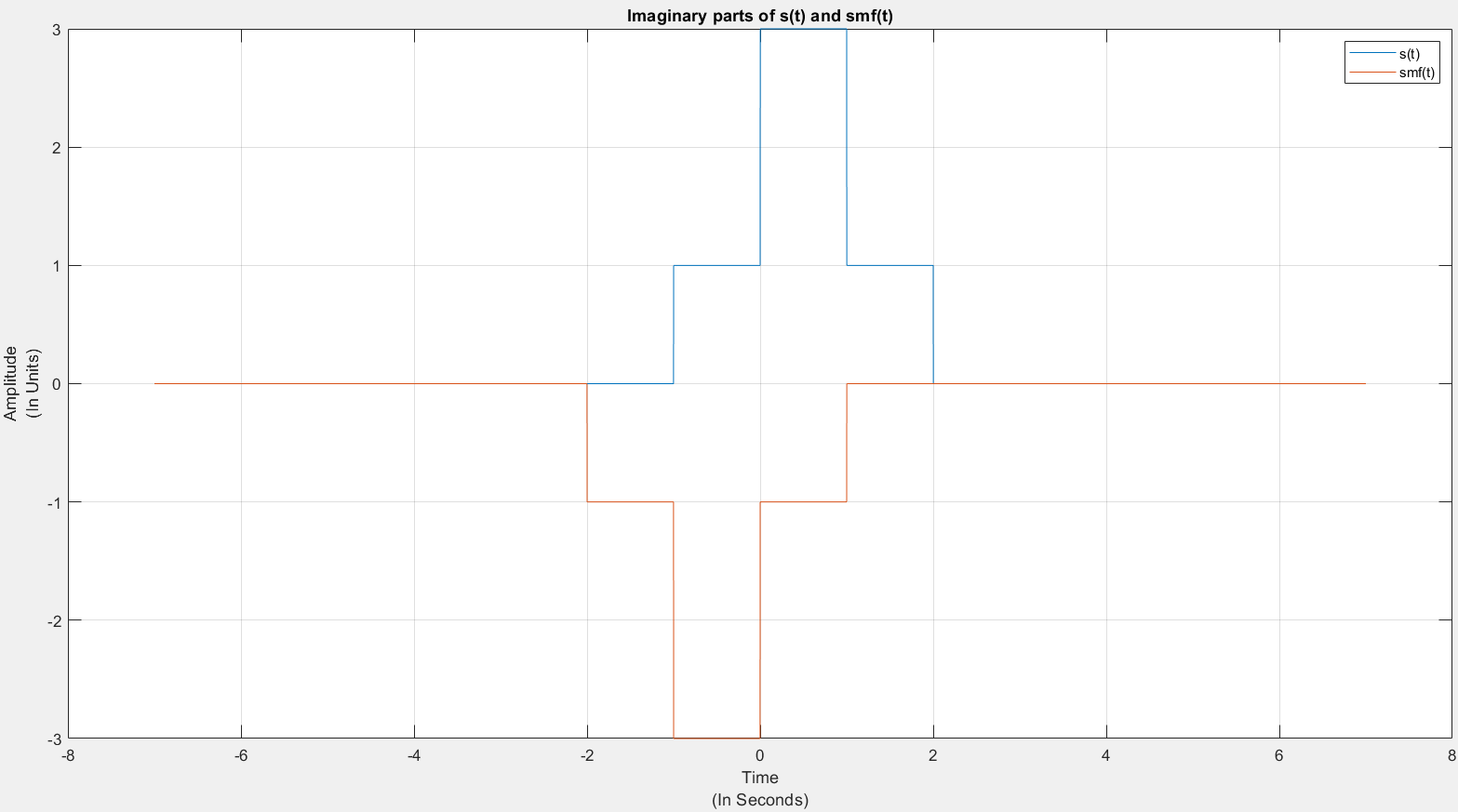


Ans: Peak can be seen at t = 0s in the plot.

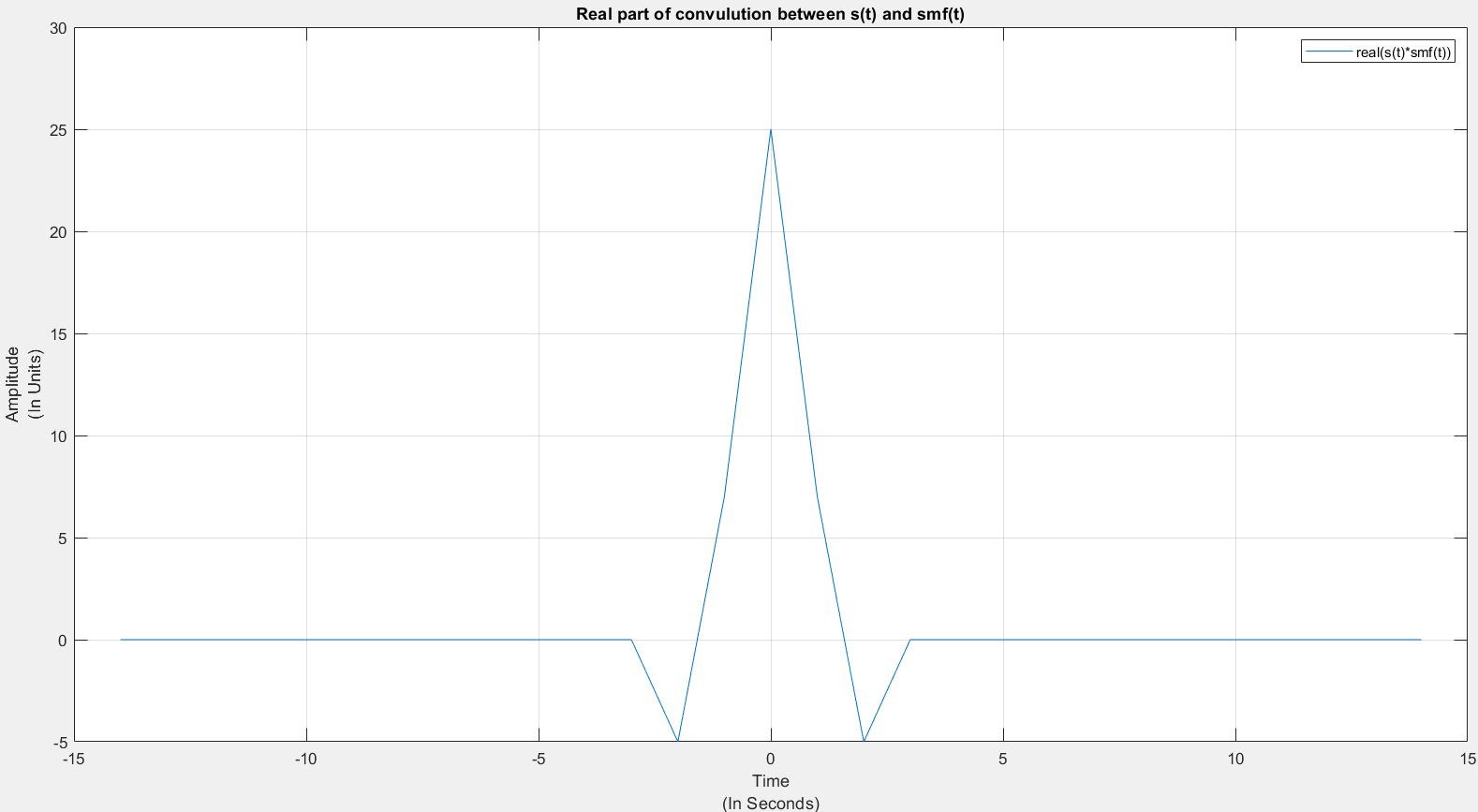
1(c): (i)



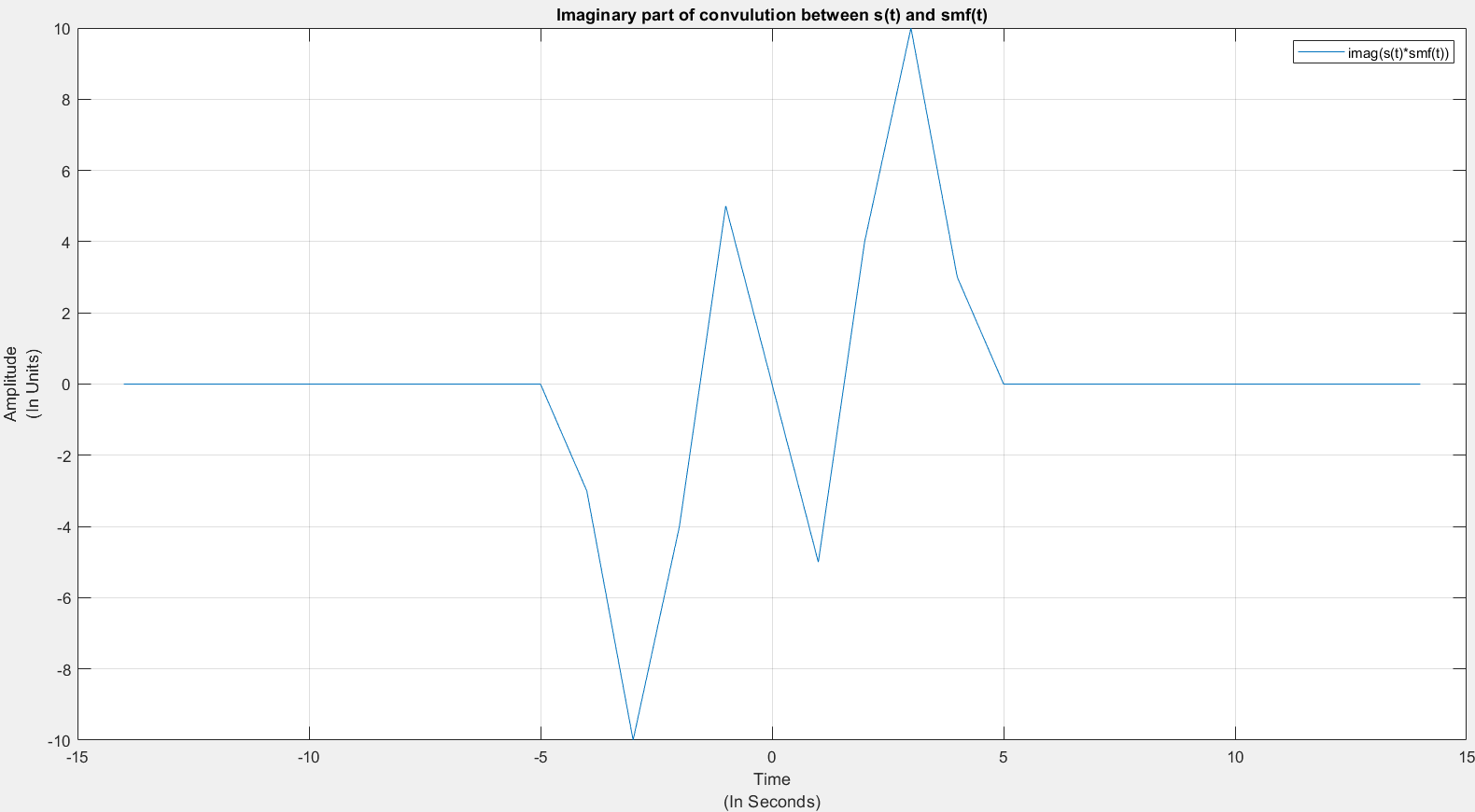
1(c): (ii)



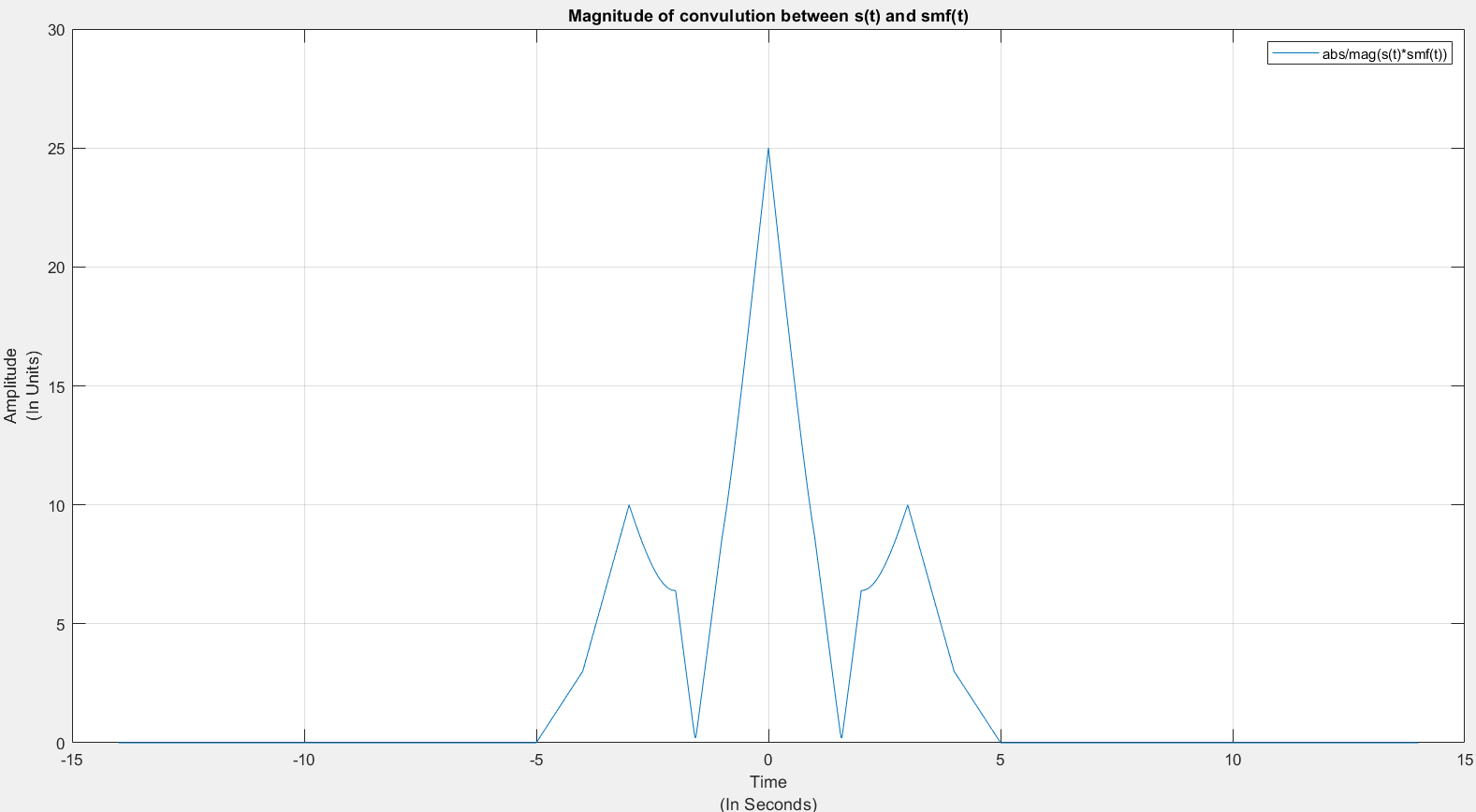
1(d): (i)



1(d): (ii)

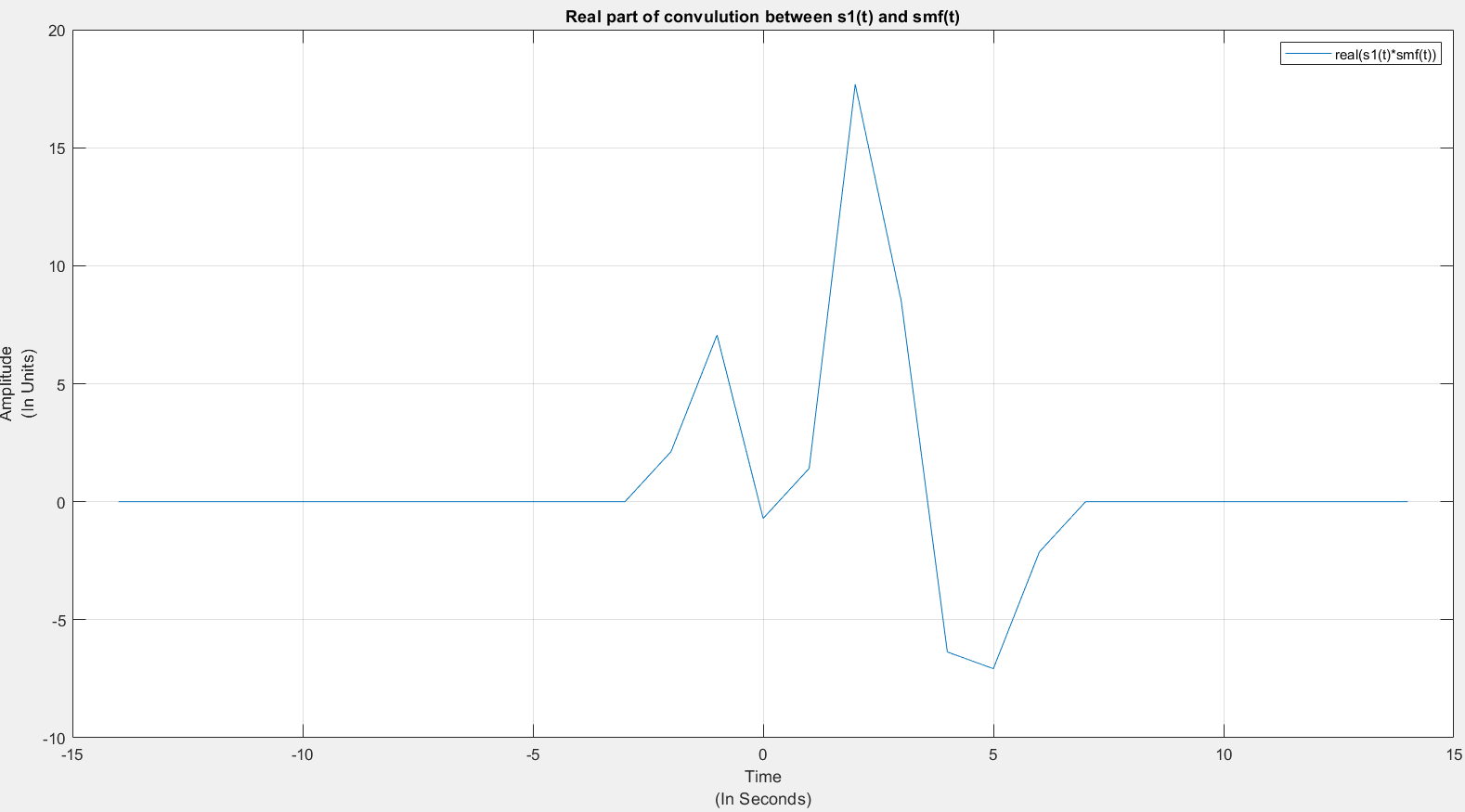


1(d): (iii)

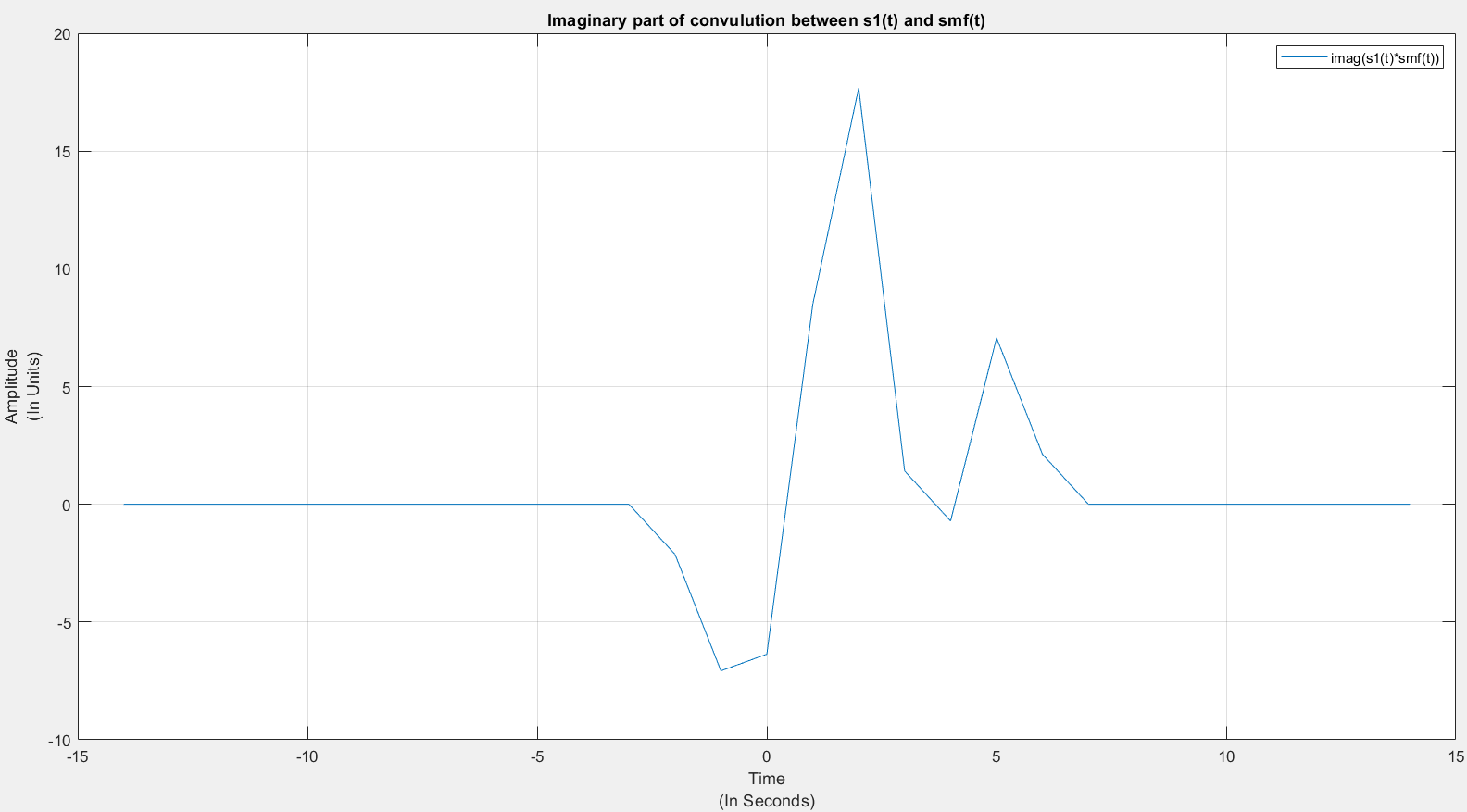


Ans: Peak can be seen in plot-7 at t = 0s.

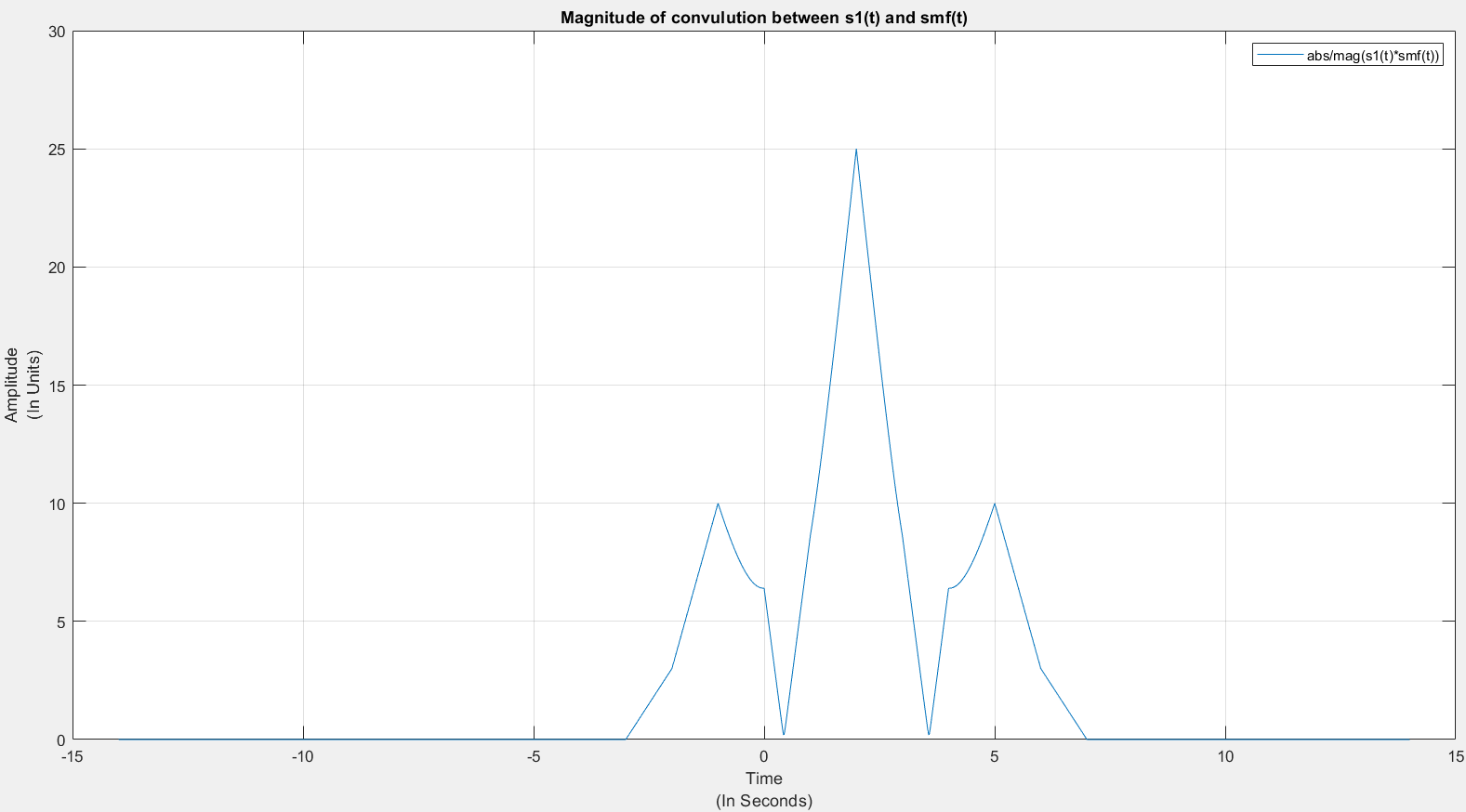
1(e): (i)



1(e): (ii)



1(e): (iii)



Ans: Peak can be seen in plot-10 at t = 2s.

1(f):

Let y(t) = s(t) \* smf(t) . (Here y(t) is output of convolution and ‘\*’ here means convolution)

s1(t) \* smf(t) = {eiθ {s(t – to)} \* smf(t) Eq(1)

We know that,

There is a shift in the convolution of s1(t) and smf(t)(Plot-10 shifted by 2 unit compared to Plot-7) that shift corresponds to the shift in s1(t-to). In our case to = 2.

Eq(1):

= [y(t – t0)]x[ei(θ)]

= [y(t – t0)]x[cos(θ) + i sin(θ)] (here ei(θ) = cos(θ) + i sin(θ))

= {[real(y(t-to])]x[cos(θ)] – [imag(y(t-to))]x[sin(θ)]} + i {[real(y(t-to))]x[sin(θ)] + [imag(y(t- to))]x[cos(θ)]} }

Now, in the imaginary graph of y(t) find t’ where the imag(y(t’)) = 0. Substitute t = t’ + t0 in above equation. And in the real graph of the equation get the value at t = t’ + t0. Divide this value with the value of the real graph of y(t) at t = t’. The ratio will be equal to cos(θ).

Similarly, we get the time t’’ in the real graph of y(t) where the value is 0. We substitute t = t’’ + t0 in the above equation and divide it with real graph of y(t) at t = t’’. This gives us the -sin(θ).

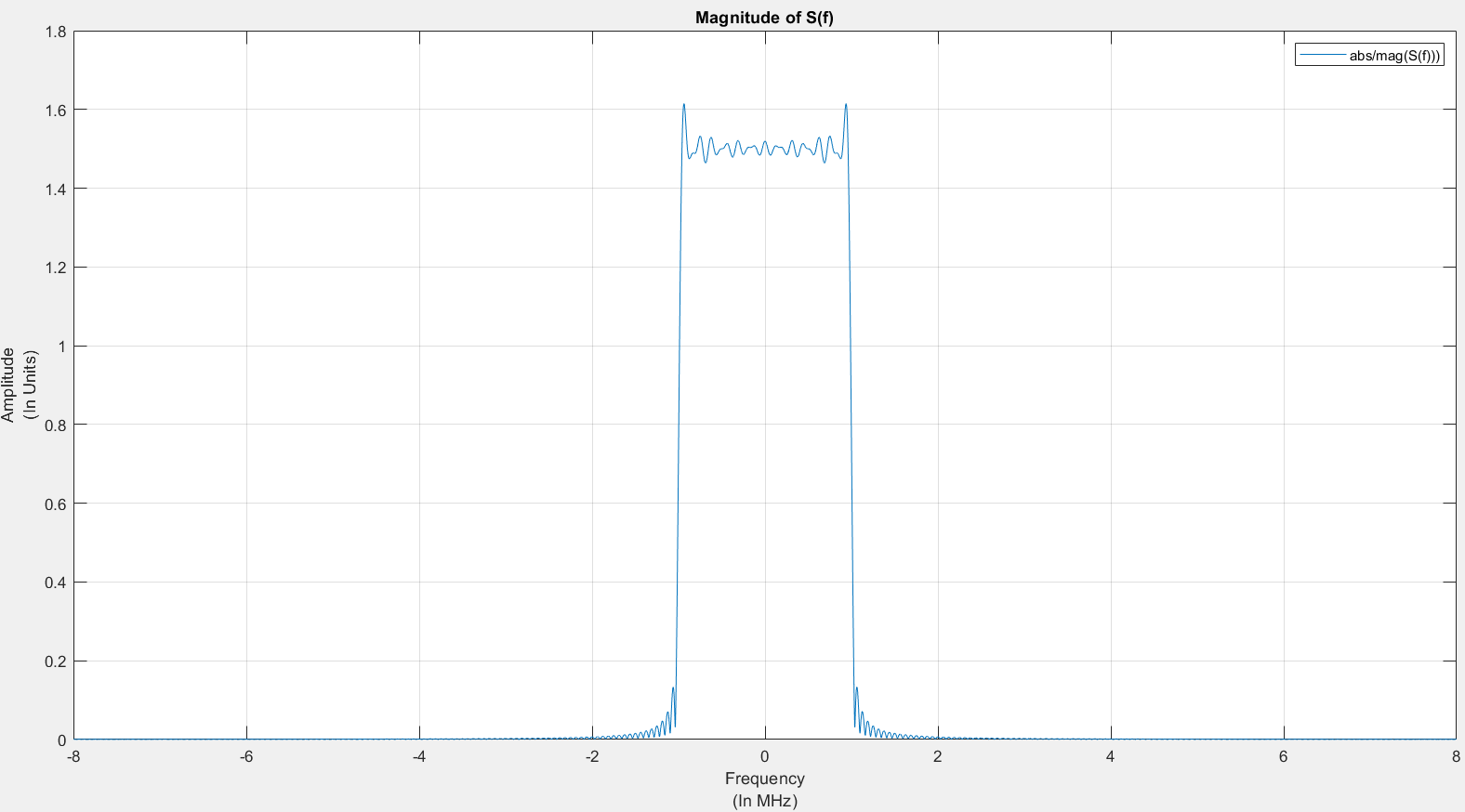
Final Conclusion:

θ = tan-1(sin (θ)/cos(θ))

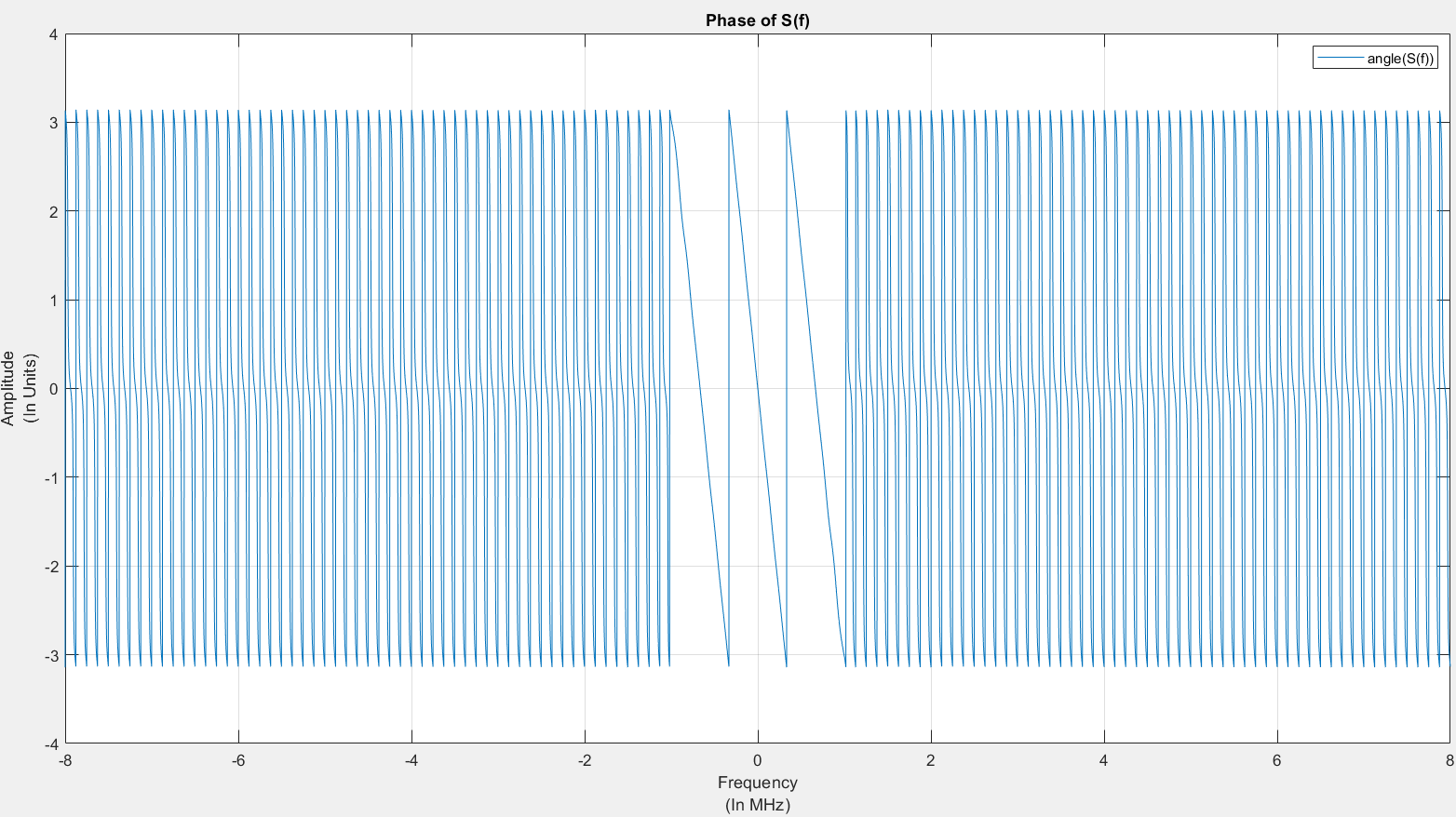
Answer to Q2

Note: For question 2 s(t) = 3sinc(2t − 3), unit time is microsecond and signal is sampled at 16MHz and is truncated to the range [-8,8] and desired frequency is 1 MHz. Here S(f) is fourier transform of s(t)

2(b):



2(c):



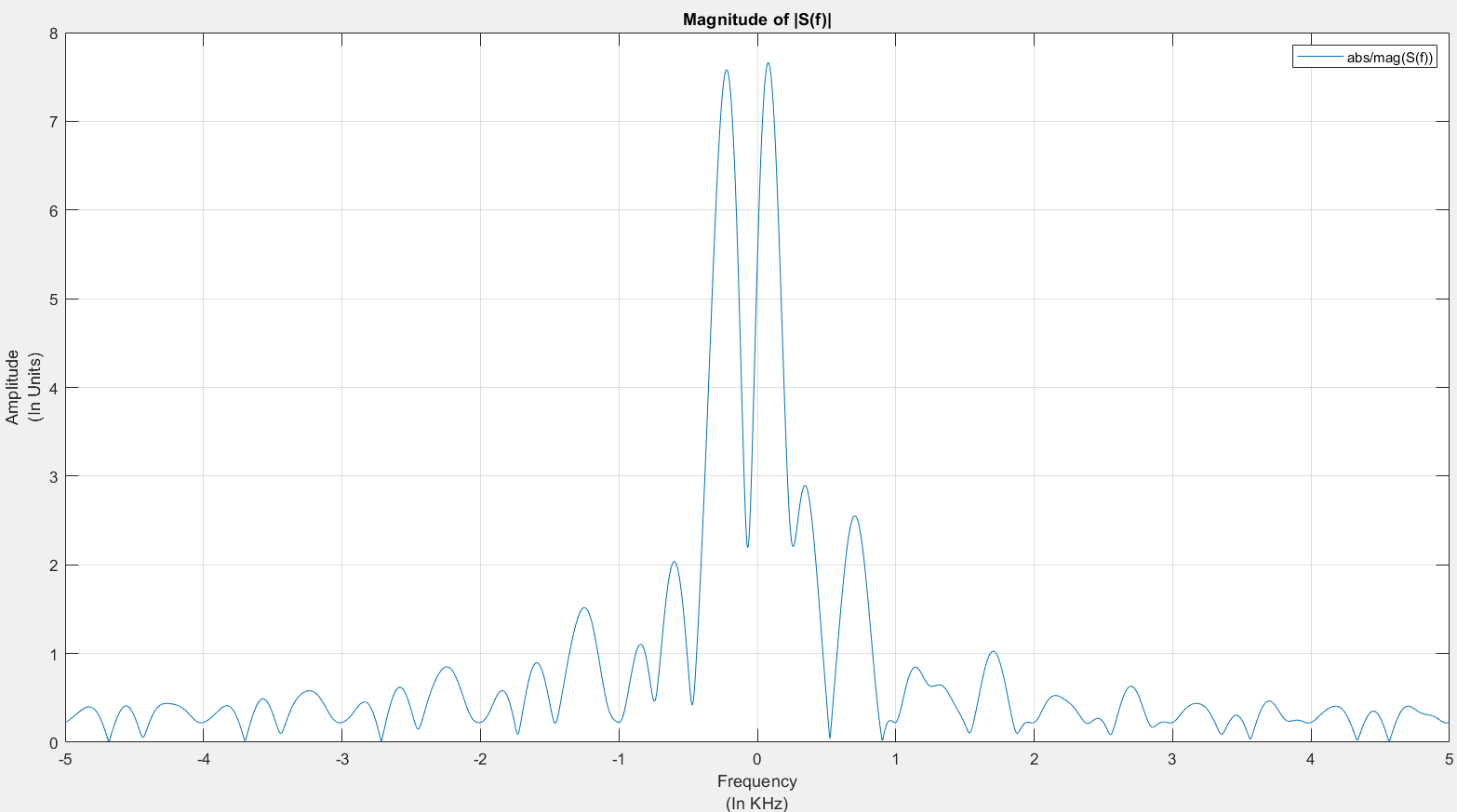
Ans: The range of frequencies over which phase plot has meaning is[-1,1] MHz.

Answer to Q3

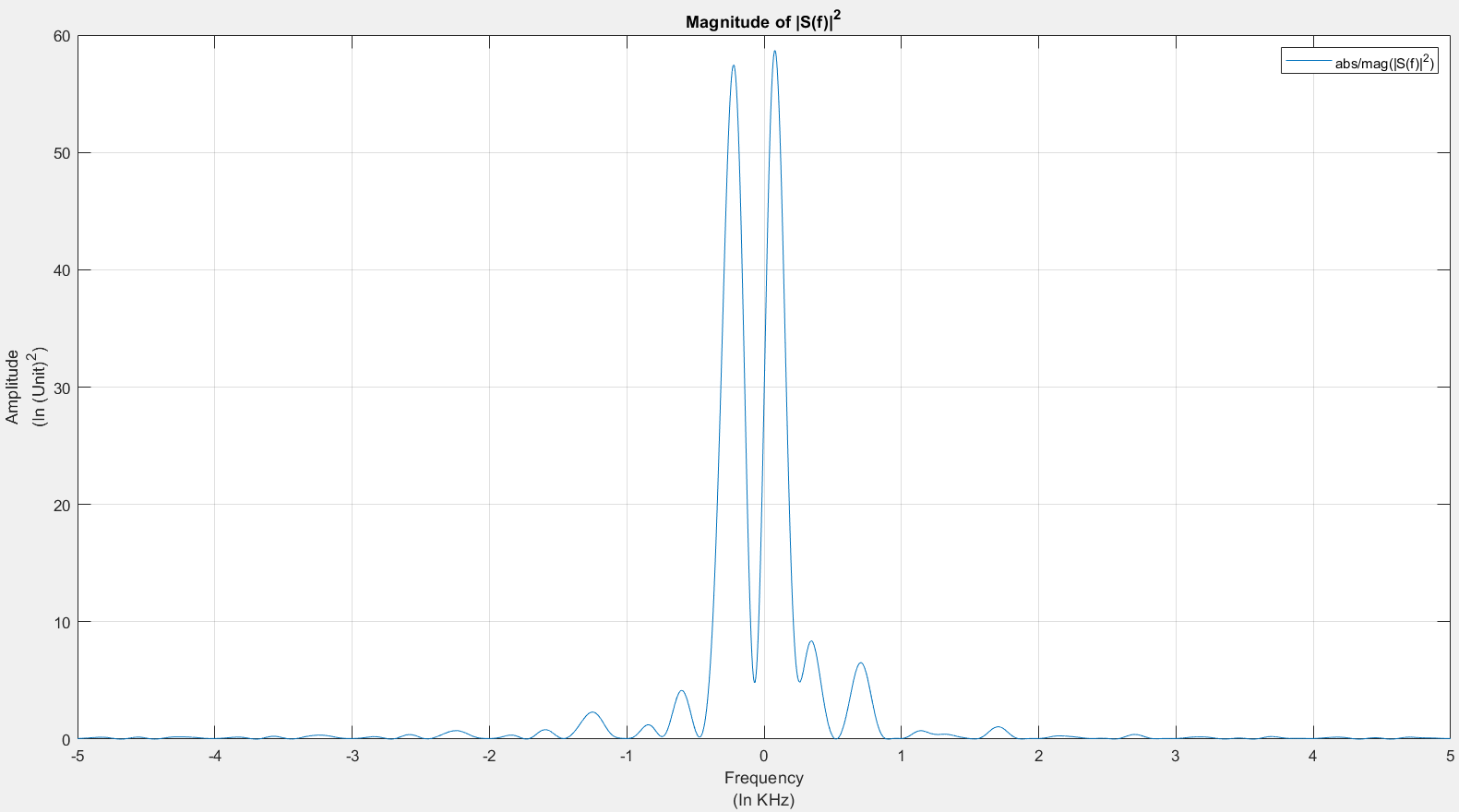
Note: For question 3 we will consider s(t) from question 1 ie s(t) = u(t) + j v(t) where u(t) = 2I[1,3](t) – 3I[2,4](t), v(t) = ​I[-1,2](t) + 2I[0,1](t) and smf = s\*(-t),

‘\*’ here means conjugate. Here unit time is millisecond and is sampled at 10MHz

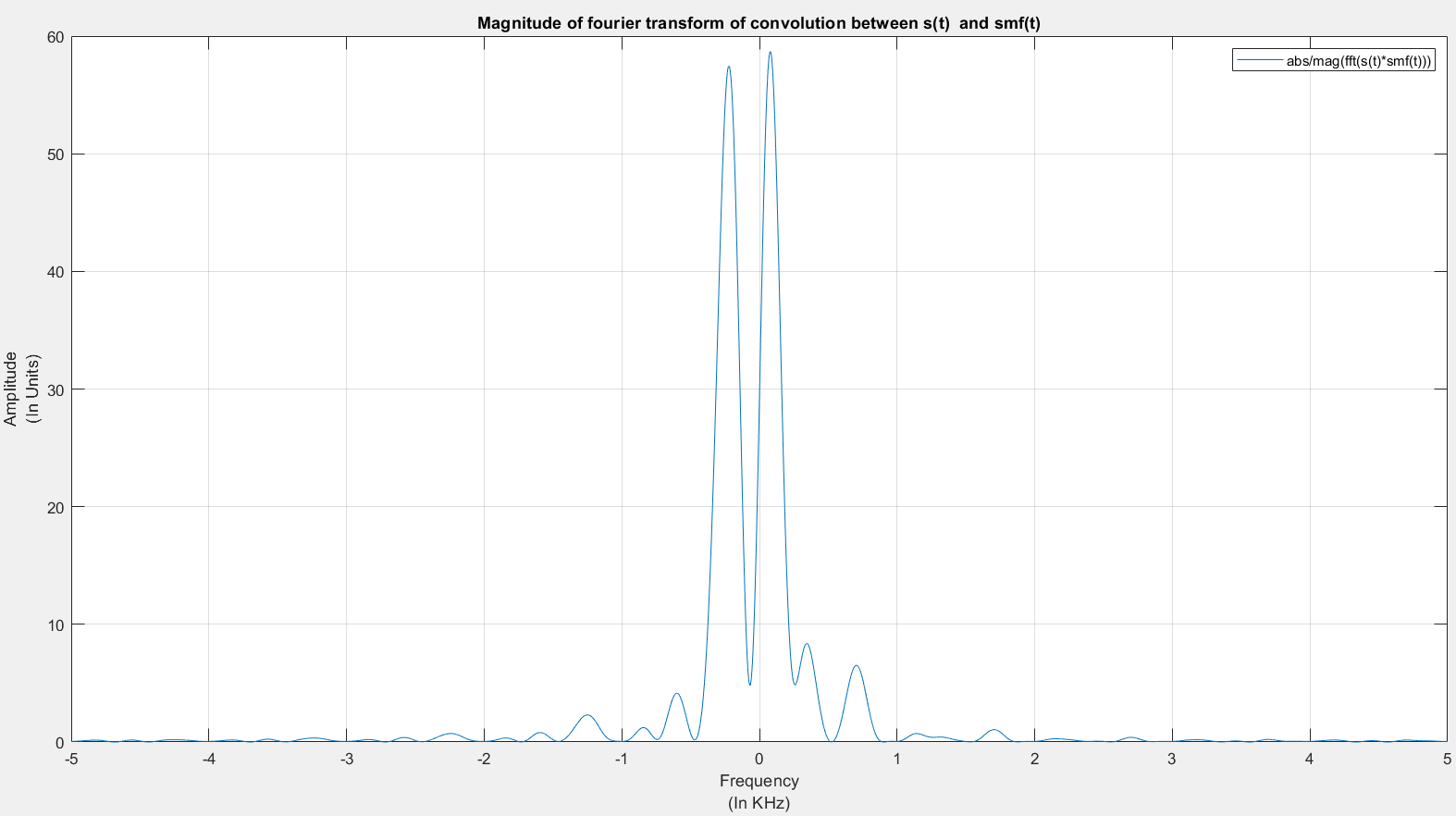
3(a):

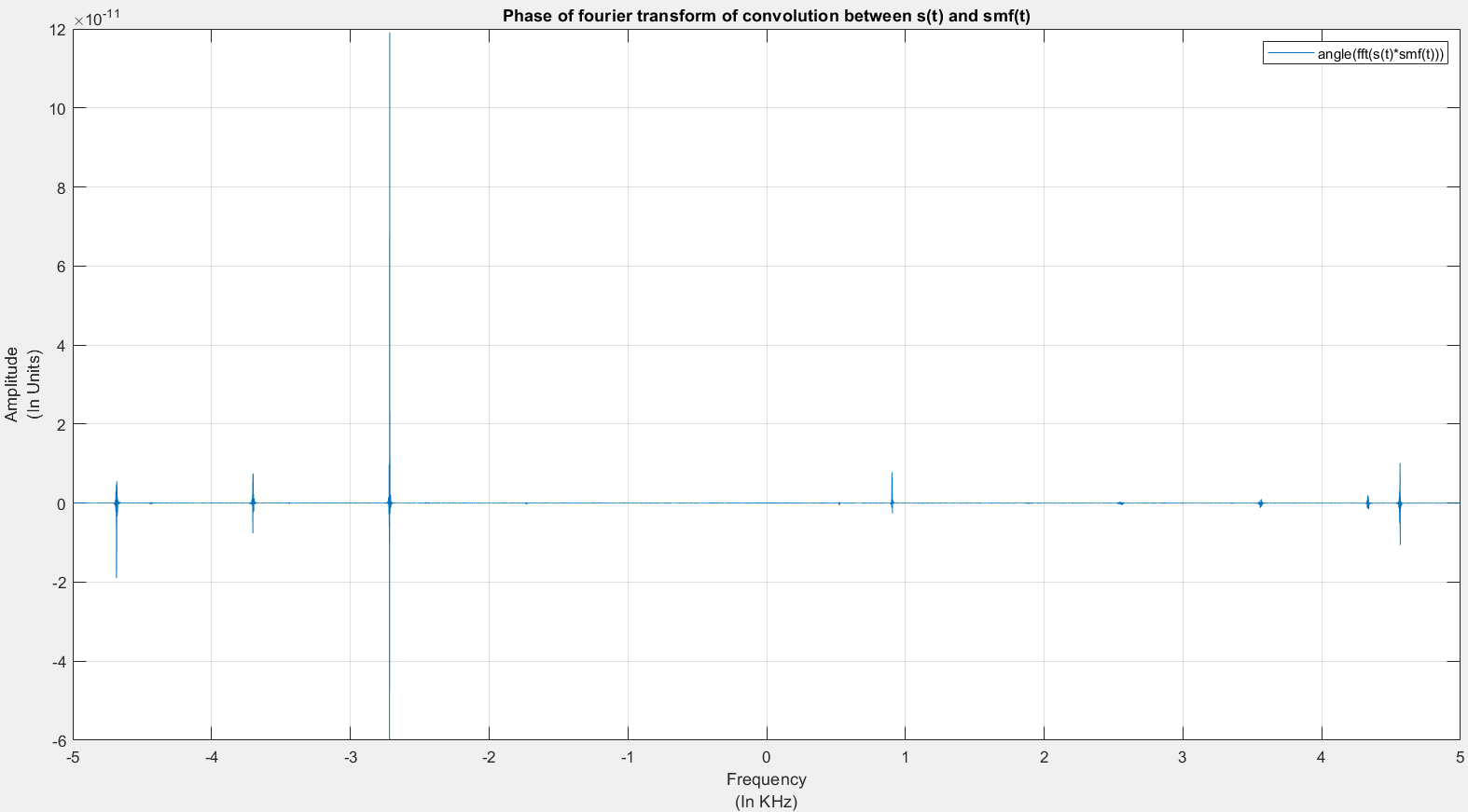


3(a2): The 3(b) Plot ie Plot-14 matches with this plot that means magnitude of convolution and |S(f)^2| is same.



3(b):



3(c): 

Ans: The convolution in time domain is nothing but multiplication in frequency domain so s(t)\*smf(t) 🡪(fft) gives us S(f)xS(-f) (because smf(t) = s\*(-t) and we know that s\*(-t) 🡪(fft) is S(-f)) so we can see that S(f) and S(-f) have opposite in sign and equal in magnitude phase so we get all the phase cancelled and phase plot is almost zero in amplitude.

Appendix:

Code for Q1 :

dt = 0.001;

t = -5:dt:5;

u = signalx(t);

umf = signalx(-t);

figure(1);

plot(t,u);

hold on;

plot(t,umf);

hold off;

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Plot of u(t) and umf(t)');

legend('u(t)','umf(t)');

grid on;

[y,t1] = contconv(double(u),double(umf),t(1),t(1),dt);

figure(2);

plot(t1,y);

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Convoluton of u(t) and umf(t)');

legend('u(t)\*umf(t)');

grid on;

function u = signalx(t)

syms x;

y = piecewise(1 <= x <= 2, 2, 2 <= x <= 3, -1, 3 <= x <= 4, -3, 0);

u = subs(y,x,t);

end

function [y,t] = contconv(x1,x2,s1,s2,dt)

y = conv(x1,x2)\*dt;

s1\_2 = s1 + (length(x1)-1)\*dt;

s2\_2 = s2 + (length(x2)-1)\*dt;

t1 = s1+ s2;

t2 = s2\_2 + s1\_2;

t = t1:dt:t2;

end

dt = 0.001;

to = 2;

theta = pi/4;

t = -7:dt:7;

u = signalx(t);

v = signalx1(t);

s = u + 1i\*v;

sc = signalx(-t) - 1i\*signalx1(-t);

figure(3);

plot(t,double(real(s)));

hold on;

plot(t,double(real(sc)));

hold off;

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Real parts of s(t) and smf(t)');

legend('s(t)','smf(t)');

grid on;

figure(4);

plot(t,imag(s));

hold on;

plot(t,imag(sc));

hold off;

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Imaginary parts of s(t) and smf(t)');

legend('s(t)','smf(t)');

grid on;

[s\_c,t1] = contconv(double(s),double(sc),t(1),t(1),dt);

figure(5);

plot(t1,real(s\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Real part of convulution between s(t) and smf(t)');

legend('real(s(t)\*smf(t))');

grid on;

figure(6);

plot(t1,imag(s\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Imaginary part of convulution between s(t) and smf(t)');

legend('imag(s(t)\*smf(t))');

grid on;

figure(7);

plot(t1,abs(s\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Magnitude of convulution between s(t) and smf(t)');

legend('abs/mag(s(t)\*smf(t))');

grid on;

s1 = (signalx(t-2) + 1i\*signalx1(t-2))\*exp(1i\*theta);

[s1\_c,t3] = contconv(double(s1),double(sc),t(1),t(1),dt);

figure(8);

plot(t3,real(s1\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Real part of convulution between s1(t) and smf(t)');

legend('real(s1(t)\*smf(t))');

grid on;

figure(9);

plot(t3,imag(s1\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Imaginary part of convulution between s1(t) and smf(t)');

legend('imag(s1(t)\*smf(t))');

grid on;

figure(10);

plot(t3,abs(s1\_c));

xlabel({'Time','(In Seconds)'});ylabel({'Amplitude','(In Units)'});

title('Magnitude of convulution between s1(t) and smf(t)');

legend('abs/mag(s1(t)\*smf(t))');

xt = get(gca, 'XTick');

set(gca, 'XTick',xt, 'XTickLabel',xt/1)

grid on;

grid minor;

function u = signalx(t)

syms x;

y = piecewise(1 <= x <= 2, 2, 2 <= x <= 3, -1, 3 <= x <= 4, -3, 0);

u = subs(y,x,t);

end

function v = signalx1(t)

syms x;

y = piecewise(-1 <= x <= 0, 1, 0 <= x <= 1, 3, 1 <= x <= 2, 1, 0);

v = subs(y,x,t);

end

function [y,t] = contconv(x1,x2,s1,s2,dt)

y = conv(x1,x2)\*dt;

s1\_2 = s1 + (length(x1)-1)\*dt;

s2\_2 = s2 + (length(x2)-1)\*dt;

t1 = s1+ s2;

t2 = s2\_2 + s1\_2;

t = t1:dt:t2;

end

Code for Q2 and Q3 :

function two

q\_2();

q\_3();

end

function q\_2

dt = (1/16);

t = -8:dt:8;

s = 3\*sinc(2\*t - 3);

[Y,f,df] = contFT(s,t(1),dt,10^(-3));

figure(11);

plot(f,abs(Y));

xlabel({'Frequency','(In MHz)'});ylabel({'Amplitude','(In Units)'});

title('Magnitude of S(f)');

legend('abs/mag(S(f)))');

grid on;

figure(12);

plot(f,angle(Y));

xlabel({'Frequency','(In MHz)'});ylabel({'Amplitude','(In Units)'});

title('Phase of S(f)');

legend('angle(S(f))');

grid on;

grid minor;

end

function q\_3

dt = 0.1;

to = 2;

theta = pi/4;

t = -7:dt:7;

u = signalx(t);

v = signalx1(t);

s = u + 1i\*v;

sc = signalx(-t) - 1i\*signalx1(-t);

[s\_c,t1] = contconv(double(s),double(sc),t(1),t(1),dt);

[S,f,df] = contFT(double(s),t(1),dt,10^(-3));

[S\_C,F,DF] = contFT(double(s\_c),t1(1),dt,10^(-3));

figure(13);

plot(f,abs(S));

xlabel({'Frequency','(In KHz)'});ylabel({'Amplitude','(In Units)'});

title('Magnitude of |S(f)|');

legend('abs/mag(S(f))');

grid on;

figure(14);

plot(F,abs(S\_C));

xlabel({'Frequency','(In KHz)'});ylabel({'Amplitude','(In Units)'});

title('Magnitude of fourier transform of convolution between s(t) and smf(t)');

legend('abs/mag(fft(s(t)\*smf(t)))');

grid on;

figure(15);

plot(F,angle(S\_C));

xlabel({'Frequency','(In KHz)'});ylabel({'Amplitude','(In Units)'});

title('Phase of fourier transform of convolution between s(t) and smf(t)');

legend('angle(fft(s(t)\*smf(t)))');

grid on;

end

function [y,t] = contconv(x1,x2,s1,s2,dt)

y = conv(x1,x2)\*dt;

s1\_2 = s1 + (length(x1)-1)\*dt;

s2\_2 = s2 + (length(x2)-1)\*dt;

t1 = s1+ s2;

t2 = s2\_2 + s1\_2;

t = t1:dt:t2;

end

function u = signalx(t)

syms x;

y = piecewise(1 <= x <= 2, 2, 2 <= x <= 3, -1, 3 <= x <= 4, -3, 0);

u = subs(y,x,t);

end

function v = signalx1(t)

syms x;

y = piecewise(-1 <= x <= 0, 1, 0 <= x <= 1, 3, 1 <= x <= 2, 1, 0);

v = subs(y,x,t);

end

function [X,f,df] = contFT(x,tstart,dt,df\_desired)

%Use Matlab DFT for approximate computation of continuous time Fourier transform

%INPUTS

%x = vector of time domain samples, assumed uniformly spaced %tstart= time at which first sample is taken

%dt = spacing between samples

%df\_desired = desired frequency resolution

%OUTPUTS

% X=vector of samples of Fourier transform

%f=corresponding vector of frequencies at which samples are obtained

%df=freq resolution attained (redundant--already available from %difference of consecutive entries of f

%%%%%%%%%

%minimum FFT size determined by desired freq res or length of x

Nmin=max(ceil(1/(df\_desired\*dt)),length(x));

%choose FFT size to be the next power of 2

Nfft = 2^(nextpow2(Nmin));

%compute Fourier transform, centering around DC

X=dt\*fftshift(fft(x,Nfft));

%achieved frequency resolution

df=1/(Nfft\*dt);

%range of frequencies covered

f = ((0:Nfft-1)-Nfft/2)\*df;

%same as f=-1/(2\*dt):df:1/(2\*dt) - df %phase shift associated with start time

X=X.\*exp(-1i\*2\*pi\*f\*tstart);

end