**PART 1.1**

%%%%%%%%% DTMF Transmitter %%%%%%%%%%

number = [0 5 0 7 4 7 1 6 7 9 7]; %% This is my phone number

Fs=16384;

xt = DTMFTRA(number);

soundsc(xt,Fs); %% Sounds very similar compared to landline, i tried with a real phone

function[x]=DTMFTRA(Number)

length = size(Number);%% Second element is the length

x = zeros(1,length(2)\*16384\*0.25);

column = [1336 1209 1336 1447 1209 1336 1447 1209 1336 1447];

row = [941 697 697 697 770 770 770 852 852 852];

%%%% 0 corresponds column 1336, row 941 ---- 1 corresponds column 1209,

%%%% row 607 etc. I have mapped this values here accordingly thanks to

%%%% that i dont need to use any if-else statements.

for i = 1:1:length(2)

for t = (i-1)\*0.25:(1/16384):i\*0.25-(1/16384)

%%% Small correction here to match the size of matrixes

%%% without this correction every loop

%%% adds 1 extra index which results in overflow

x(16384\*t+1) = cos(2\*pi\*t\*column(Number(i)+1))+cos(2\*pi\*t\*row(Number(i)+1));

end

end

end %% END of the DTMFTRA function

**PART 1.2**

%%%%%%%%% DTMF Receiver %%%%%%%%%%

number = [2 1 5 0 1 4 6 2]; %%%% My student number

xt = DTMFTRA(number);

Fs=16384;

%soundsc(xt,Fs);

XT = FT(xt);

omega=linspace(-16384\*pi,16384\*pi,16384\*2+1);

omega=omega(1:end-1);

figure(1)

plot(omega,abs(XT));

title("Magnitude of the Fourier Transform of x(t) ");

xlabel("w");

ylabel("X(jw)");

%%%%%% First Digit %%%%%%%

rectangle1 = [ones(1,(0.25)\*16384) zeros(1,(1.75)\*16384)];

xt1 = xt.\*rectangle1;

XT1 = FT(xt1);

figure(2)

plot(omega,abs(XT1));

title("Magnitude of the Fourier Transform of x1(t) ");

xlabel("w");

ylabel("X1(jw)");

%%%%%% Second Digit %%%%%%%

rectangle2 = [zeros(1,(0.25)\*16384) ones(1,(0.25)\*16384) zeros(1,(1.5)\*16384)];

xt2 = xt.\*rectangle2;

XT2 = FT(xt2);

figure(3)

plot(omega,abs(XT2));

title("Magnitude of the Fourier Transform of x2(t) ");

xlabel("w");

ylabel("X2(jw)");

%%%%%% Third Digit %%%%%%%

rectangle3 = [zeros(1,(0.5)\*16384) ones(1,(0.25)\*16384) zeros(1,(1.25)\*16384)];

xt3 = xt.\*rectangle3;

XT3 = FT(xt3);

figure(4)

plot(omega,abs(XT3));

title("Magnitude of the Fourier Transform of x3(t) ");

xlabel("w");

ylabel("X3(jw)");

%%%%%% Fourth Digit %%%%%%%

rectangle4 = [zeros(1,(0.75)\*16384) ones(1,(0.25)\*16384) zeros(1,(1)\*16384)];

xt4 = xt.\*rectangle4;

XT4 = FT(xt4);

figure(5)

plot(omega,abs(XT4));

title("Magnitude of the Fourier Transform of x4(t) ");

xlabel("w");

ylabel("X4(jw)");

%%%%%% Fifth Digit %%%%%%%

rectangle5 = [zeros(1,(1)\*16384) ones(1,(0.25)\*16384) zeros(1,(0.75)\*16384)];

xt5 = xt.\*rectangle5;

XT5 = FT(xt5);

figure(6)

plot(omega,abs(XT5));

title("Magnitude of the Fourier Transform of x5(t) ");

xlabel("w");

ylabel("X5(jw)");

%%%%%% Sixth Digit %%%%%%%

rectangle6 = [zeros(1,(1.25)\*16384) ones(1,(0.25)\*16384) zeros(1,(0.5)\*16384)];

xt6 = xt.\*rectangle6;

XT6 = FT(xt6);

figure(7)

plot(omega,abs(XT6));

title("Magnitude of the Fourier Transform of x6(t) ");

xlabel("w");

ylabel("X6(jw)");

%%%%%% Seventh Digit %%%%%%%

rectangle7 = [zeros(1,(1.5)\*16384) ones(1,(0.25)\*16384) zeros(1,(0.25)\*16384)];

xt7 = xt.\*rectangle7;

XT7 = FT(xt7);

figure(8)

plot(omega,abs(XT7));

title("Magnitude of the Fourier Transform of x7(t) ");

xlabel("w");

ylabel("X7(jw)");

%%%%%% Eight Digit %%%%%%%

rectangle8 = [zeros(1,(1.75)\*16384) ones(1,(0.25)\*16384)];

xt8 = xt.\*rectangle8;

XT8 = FT(xt8);

figure(9)

plot(omega,abs(XT8));

title("Magnitude of the Fourier Transform of x8(t) ");

xlabel("w");

ylabel("X8(jw)");

%%%%%%%%%%%% Functions %%%%%%%%%%%%%

function[x]=DTMFTRA(Number)

length = size(Number);%% Second element is the length

x = zeros(1,length(2)\*16384\*0.25);

column = [1336 1209 1336 1447 1209 1336 1447 1209 1336 1447];

row = [941 697 697 697 770 770 770 852 852 852];

%%%% 0 corresponds column 1336, row 941 ---- 1 corresponds column 1209,

%%%% row 607 etc. I have mapped this values here accordingly thanks to

%%%% that i dont need to use any if-else statements.

for i = 1:1:length(2)

for t = (i-1)\*0.25:(1/16384):i\*0.25-(1/16384)

%%% Small correction here to match the size of matrixes

%%% without this correction every loop

%%% adds 1 extra index which results in overflow

x(16384\*t+1) = cos(2\*pi\*t\*column(Number(i)+1))+cos(2\*pi\*t\*row(Number(i)+1));

end

end

end %% END of the DTMFTRA function

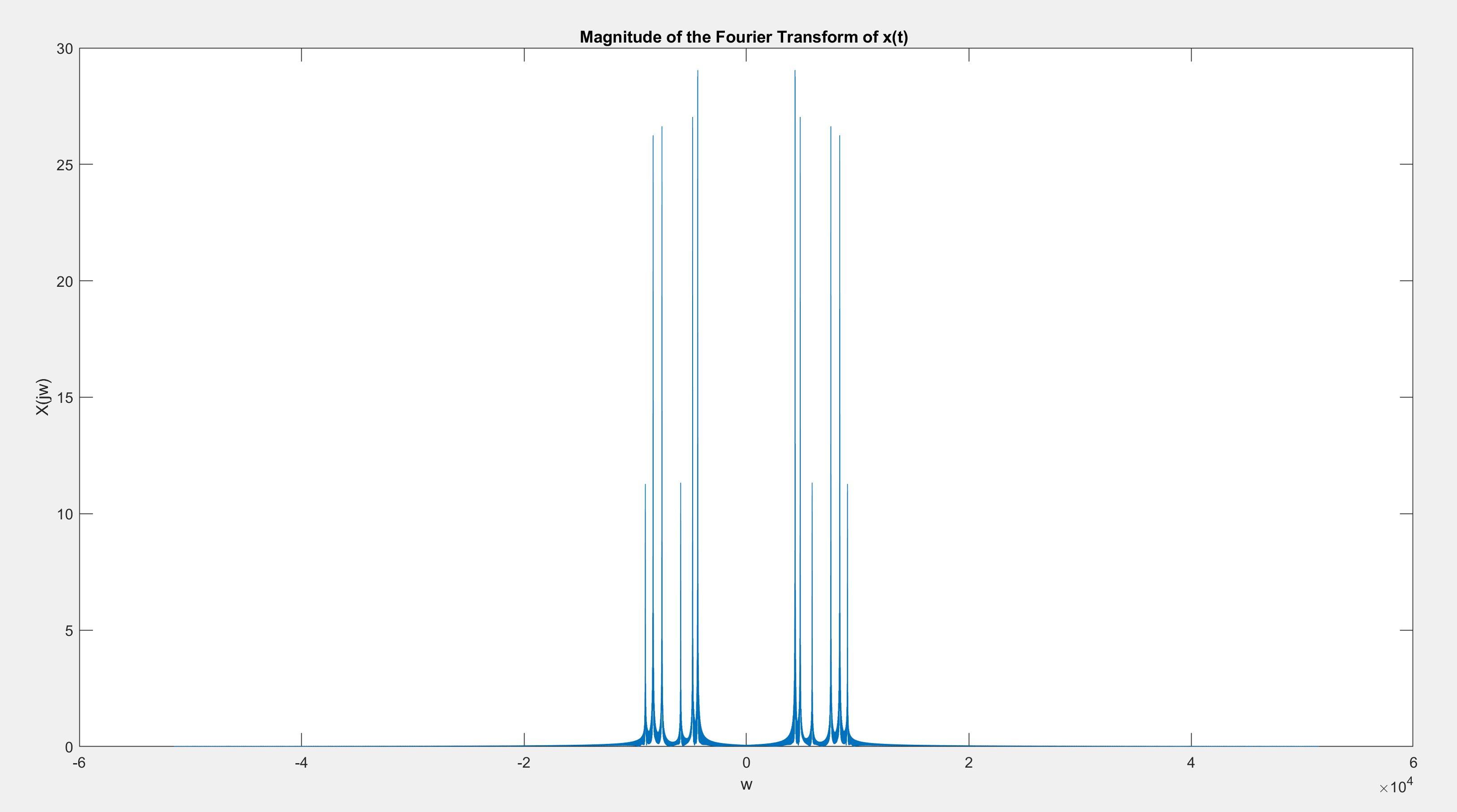
function output=FT(input)

M=length(input);

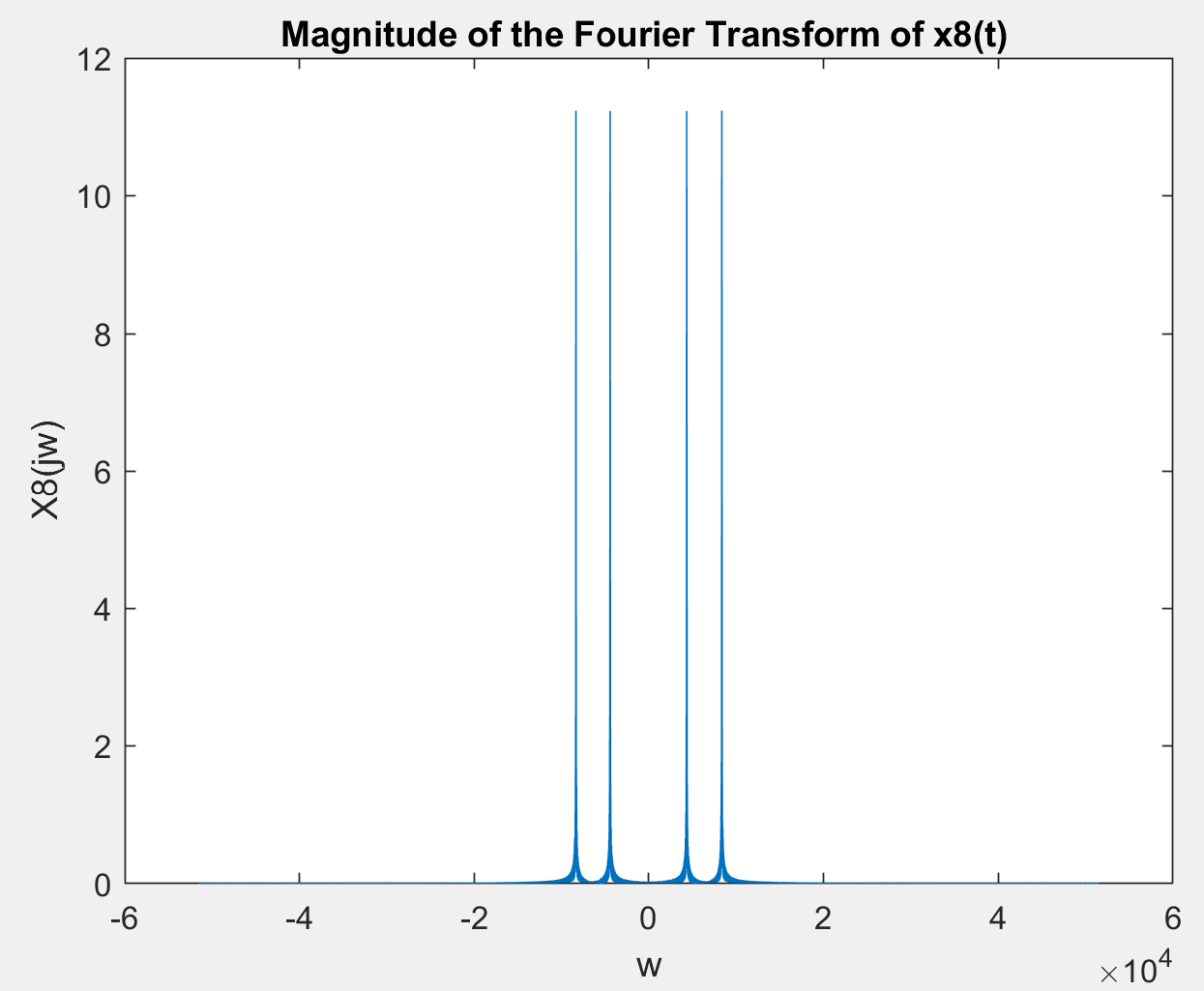
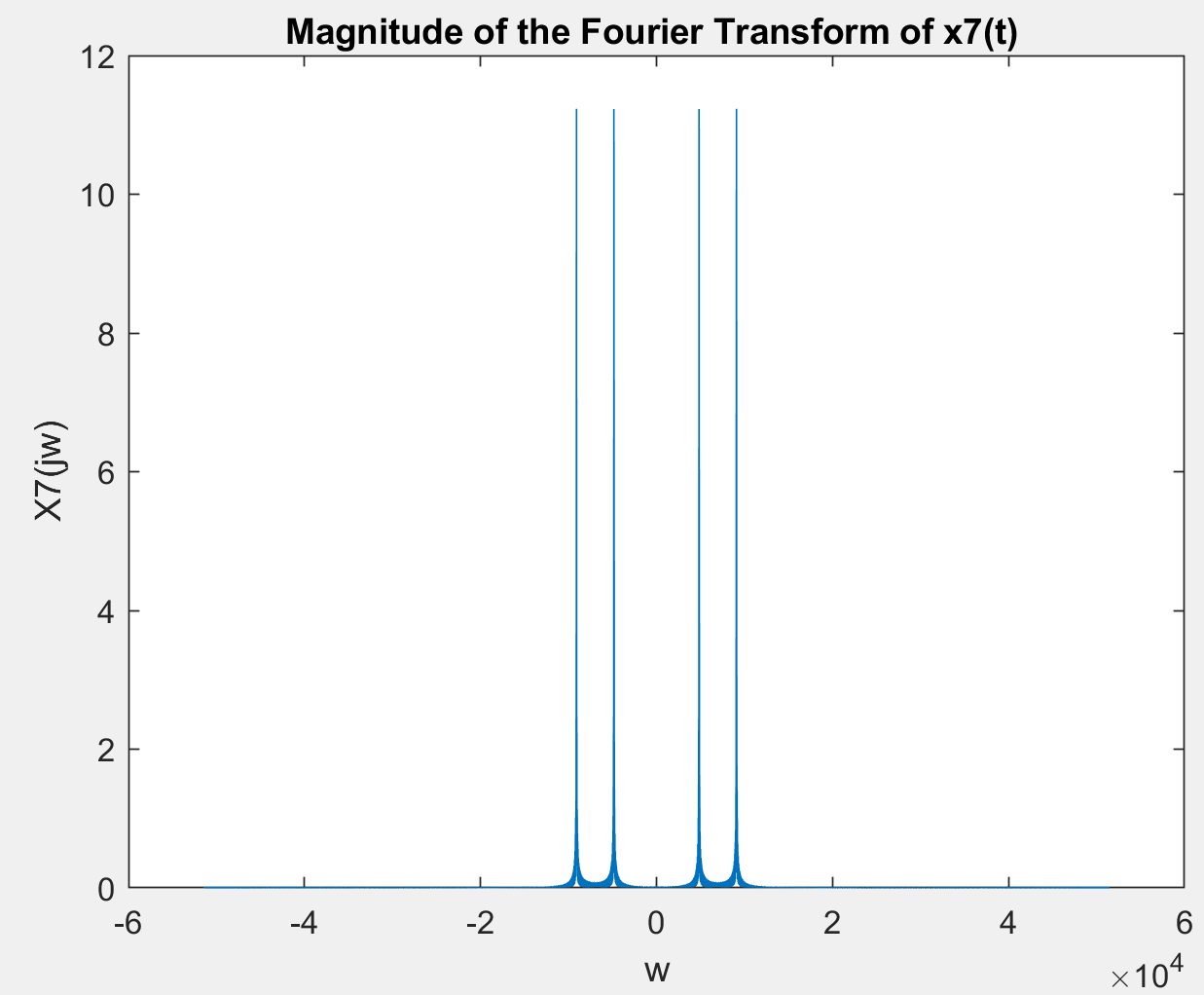
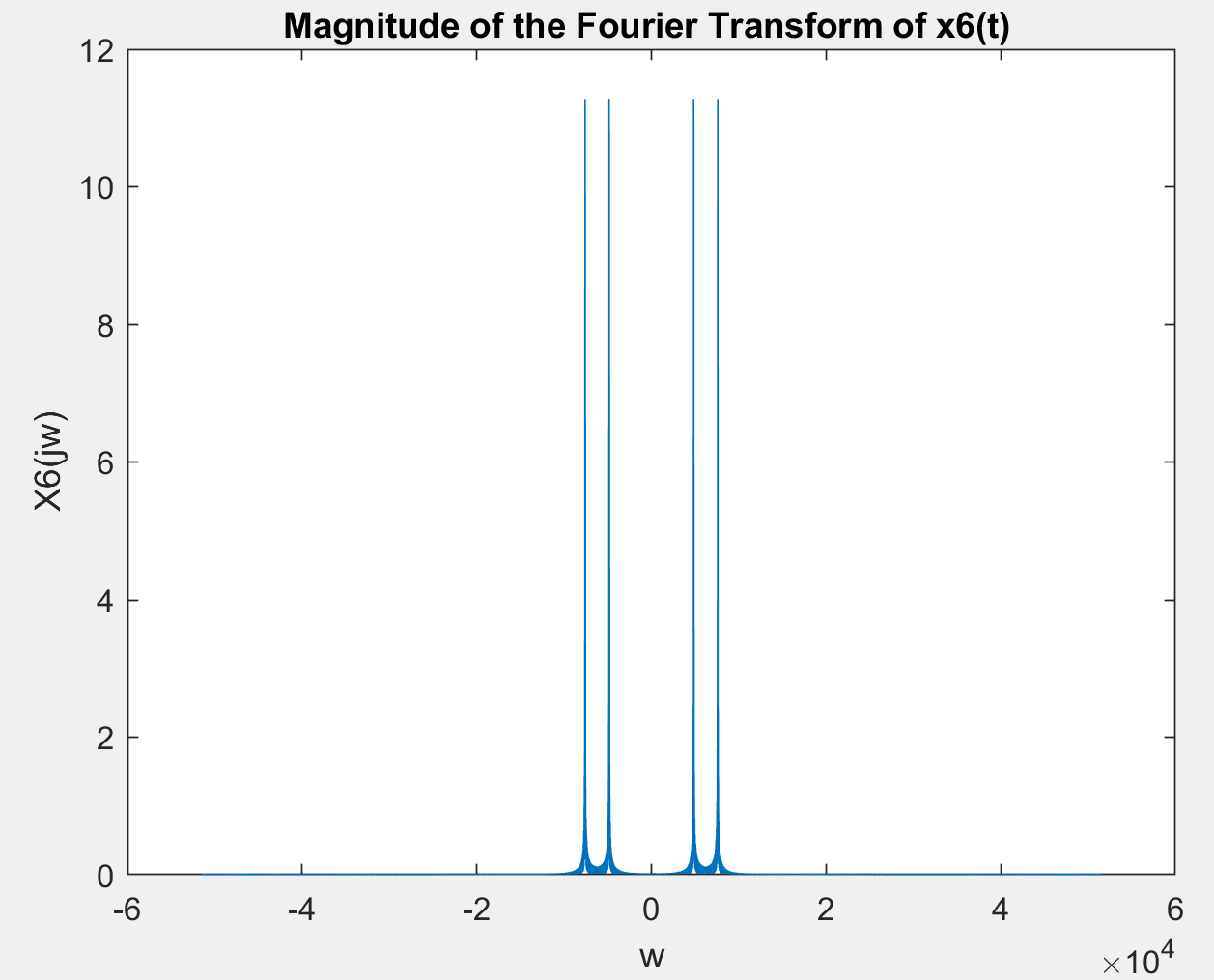
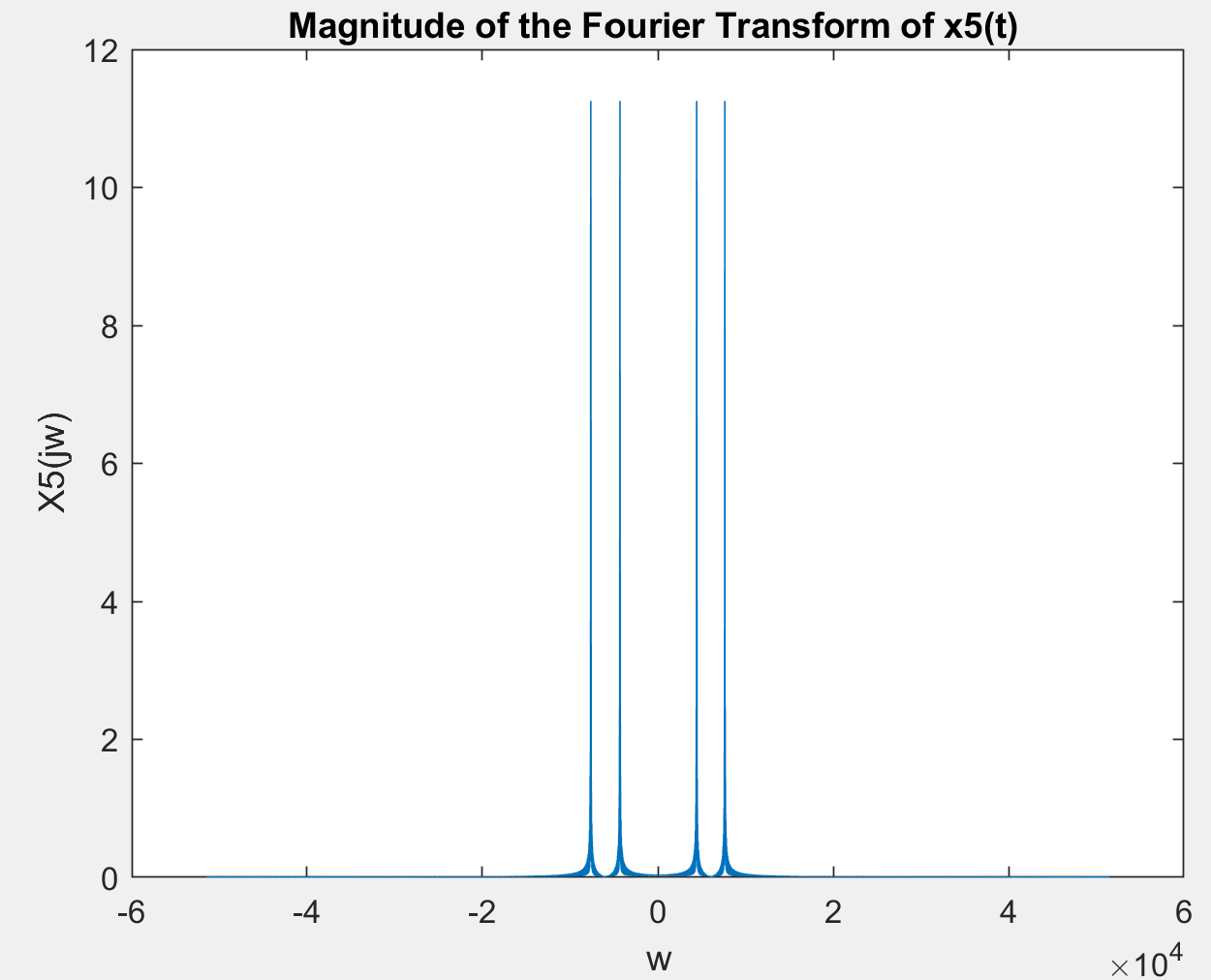
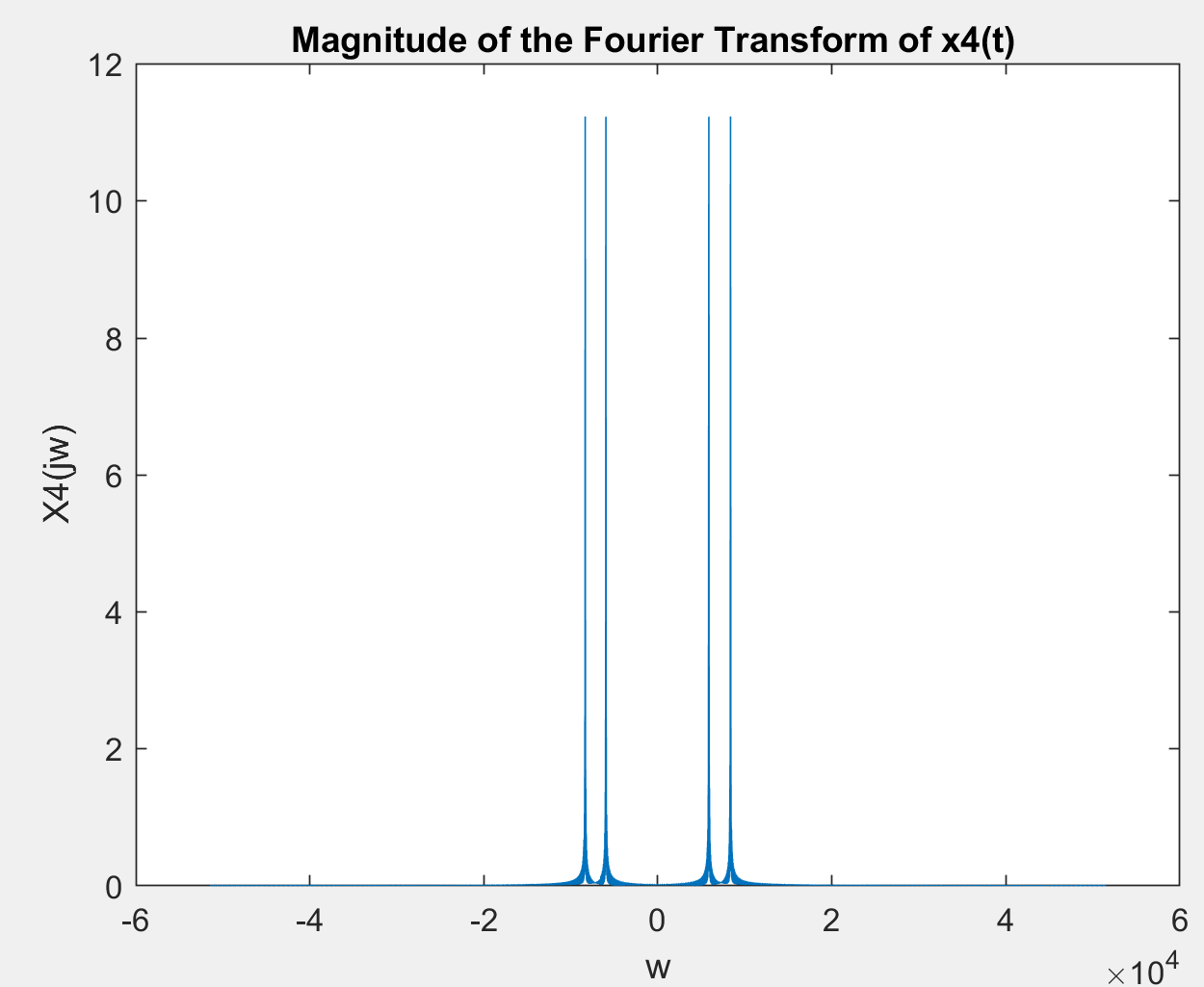
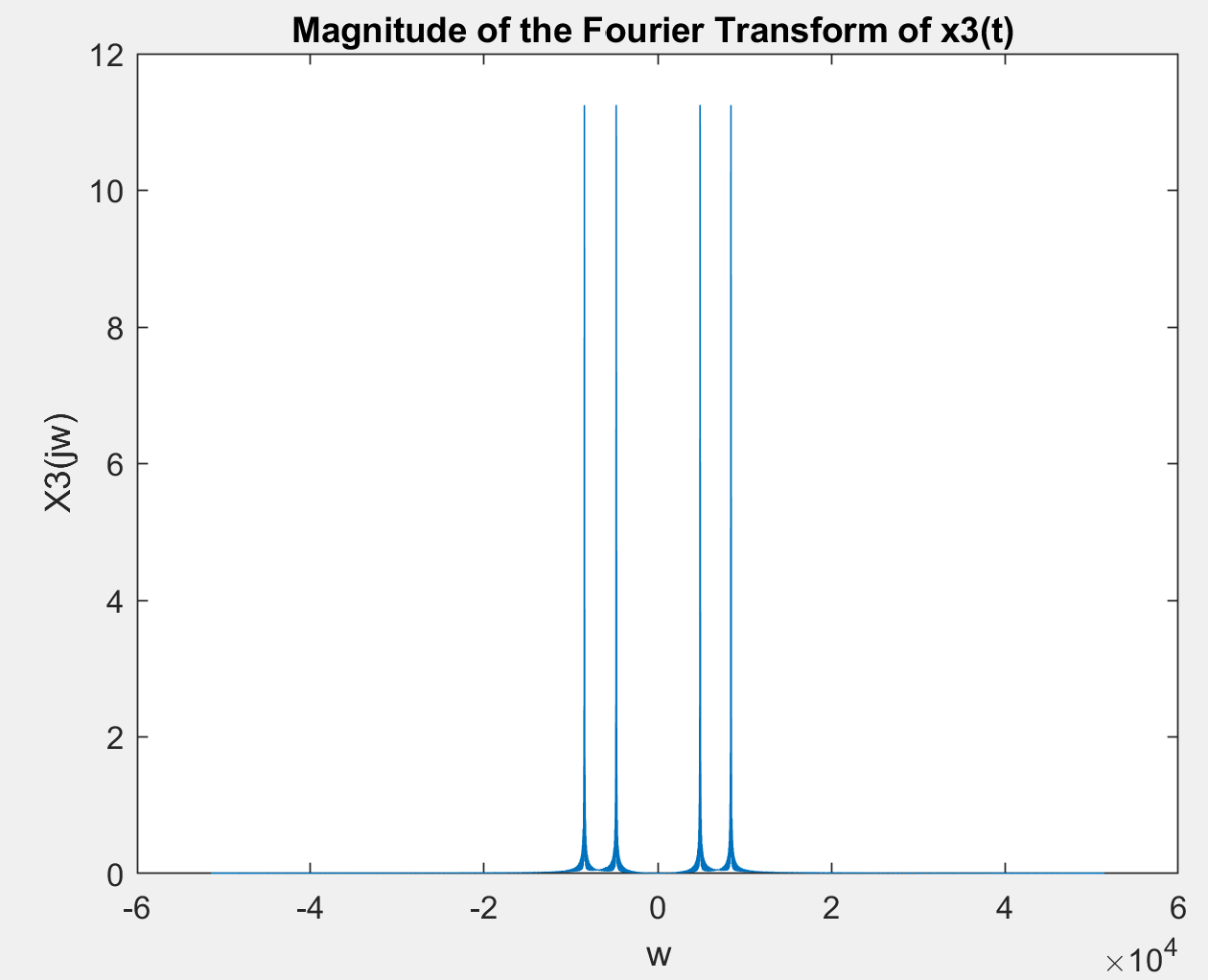
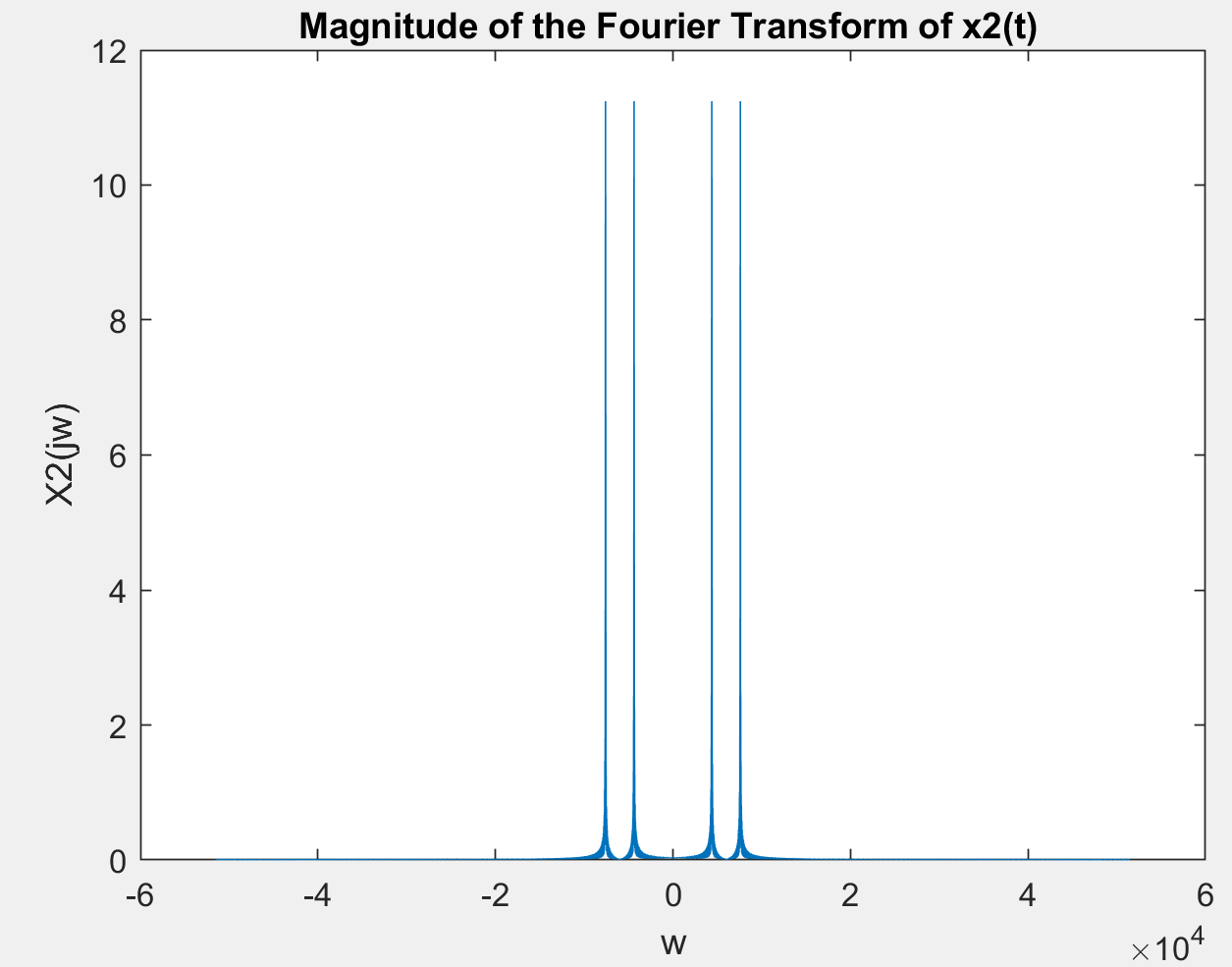
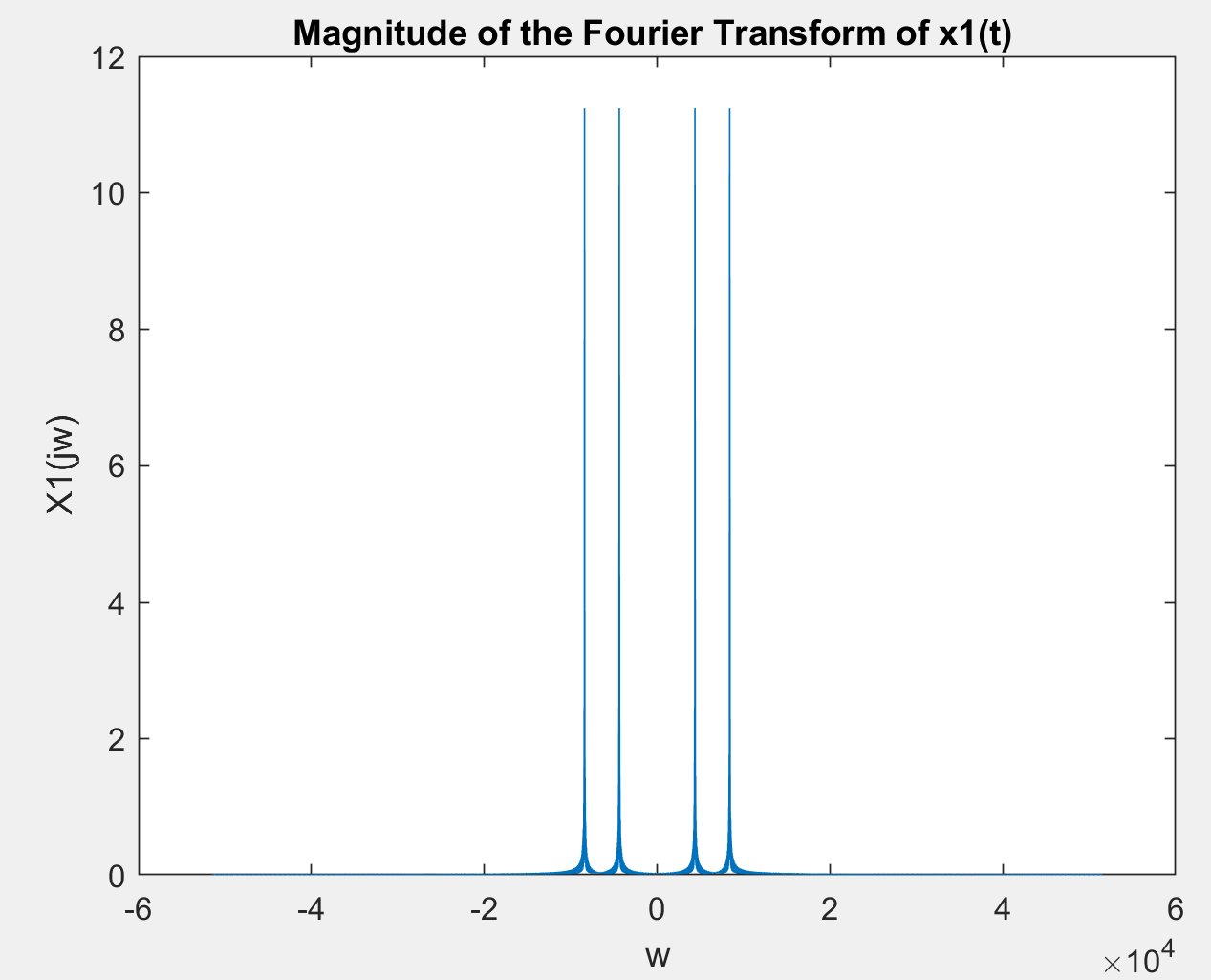
t=exp(j\*pi\*(M-1)/M\*[0:1:M-1]);

output=exp(-j\*pi\*(M-1)^2/(2\*M))\*t.\*1/(M)^0.5.\*fft(input.\*t);

end



All of the Spectrum (We can’t tell individual digits from this)



**PART 2**

clc;

clear all;

%%%%%%%%%% PART 2 %%%%%%%%%%%%%

myAudio = audiorecorder(16384,16,1);

recordblocking(myAudio, 10);

x = getaudiodata(myAudio);

t = 0 : 1/16384 :10 - (1/16384);

x = transpose(x); %%% Fixing the x(t) for calculations

figure(1)

plot(t, x);

title("Original Speech x(t) ");

xlabel("t");

ylabel("x(t)");

length = size(x);

row = length(1);

col = length(2);

x1 = [zeros(1, 0.5\*16384) 0.75\*x(1,1:col-0.5\*16384)];

x2 = [zeros(1, 1\*16384) 0.5\*x(1,1:col-1\*16384)];

x3 = [zeros(1, 1.5\*16384) 0.4\*x(1,1:col-1.5\*16384)];

x4 = [zeros(1, 2\*16384) 0.25\*x(1,1:col-2\*16384)];

x5 = [zeros(1, 3\*16384) 0.2\*x(1,1:col-3\*16384)];

x6 = [zeros(1, 4\*16384) 0.1\*x(1,1:col-4\*16384)];

y = x+x1+x2+x3+x4+x5+x6;

figure(2)

plot(t,y); %%%%%%%%%%%%%Echoed signal

title("Echoed Speech y(t) ");

xlabel("t");

ylabel("y(t)");

%%%%%%%%%%%%%%%Calculating Y(jw)%%%%%%%%%%%%%%%

omega=linspace(-16384\*pi,16384\*pi,16384\*10+1);

omega=omega(1:end-1);

Y = FT(y);

figure(3)

plot(omega, Y);

title("Echoed Speech Spectrum");

xlabel("w");

ylabel("Y(jw))");

%%%%%%%%%%%%%%%Calculating H(jw) and h(t)%%%%%%%%%%%%%%%

h1 = 0.75\*exp(-j\*omega\*0.5);

h2 = 0.5\*exp(-j\*omega\*1);

h3 = 0.4\*exp(-j\*omega\*1.5);

h4 = 0.25\*exp(-j\*omega\*2);

h5 = 0.2\*exp(-j\*omega\*3);

h6 = 0.1\*exp(-j\*omega\*4);

H = 1+h1+h2+h3+h4+h5+h6; %%%%%%%H(jw)

h = IFT(H);%%%%%%%h(t)

figure(4)

plot(t, h);

title("Impulse response h(t)");

xlabel("t");

ylabel("h(t)");

%%%%%%%%%%Original Estimation of the function

X = Y./H;

xe = IFT(X);

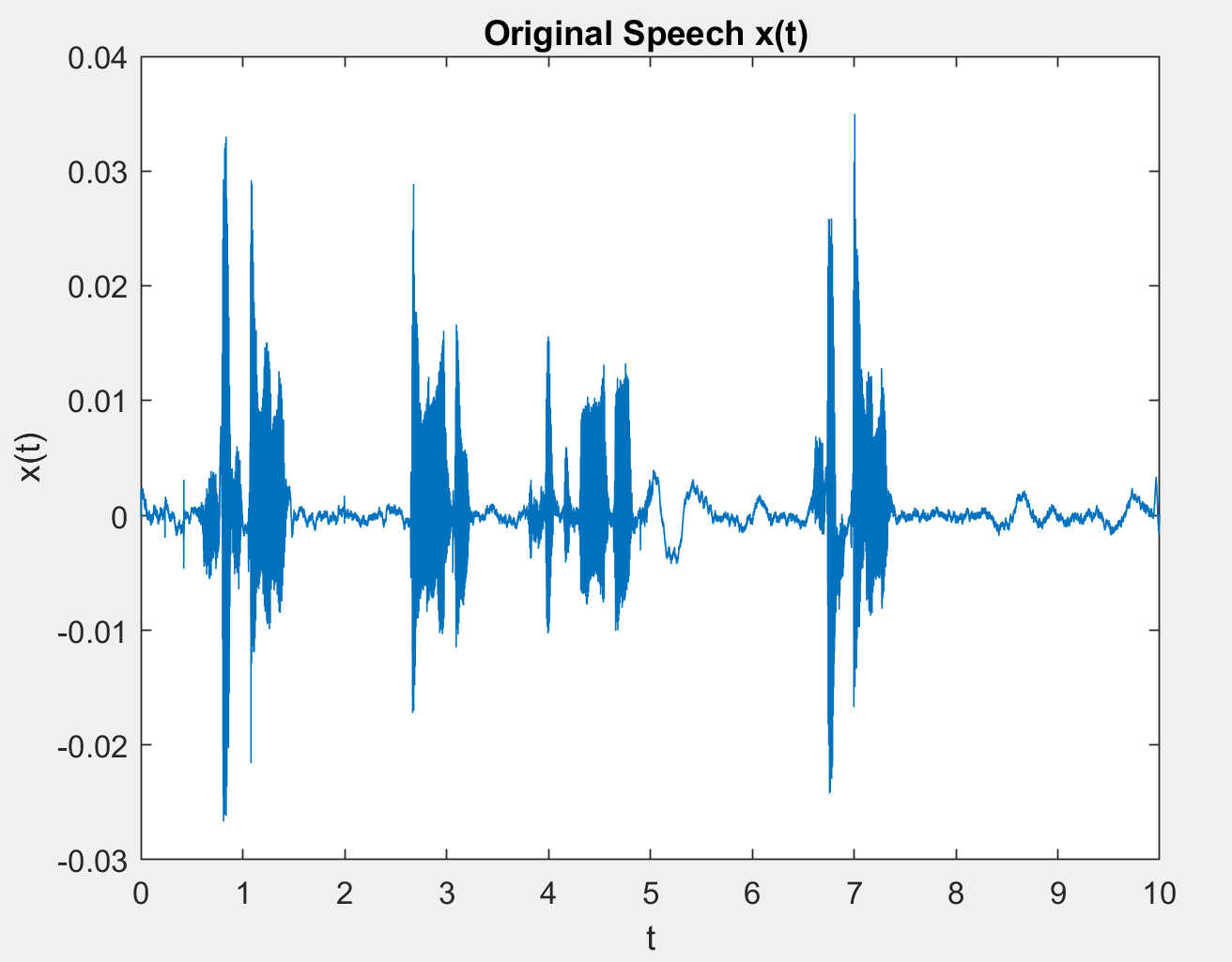
figure(5)

plot(t, xe);

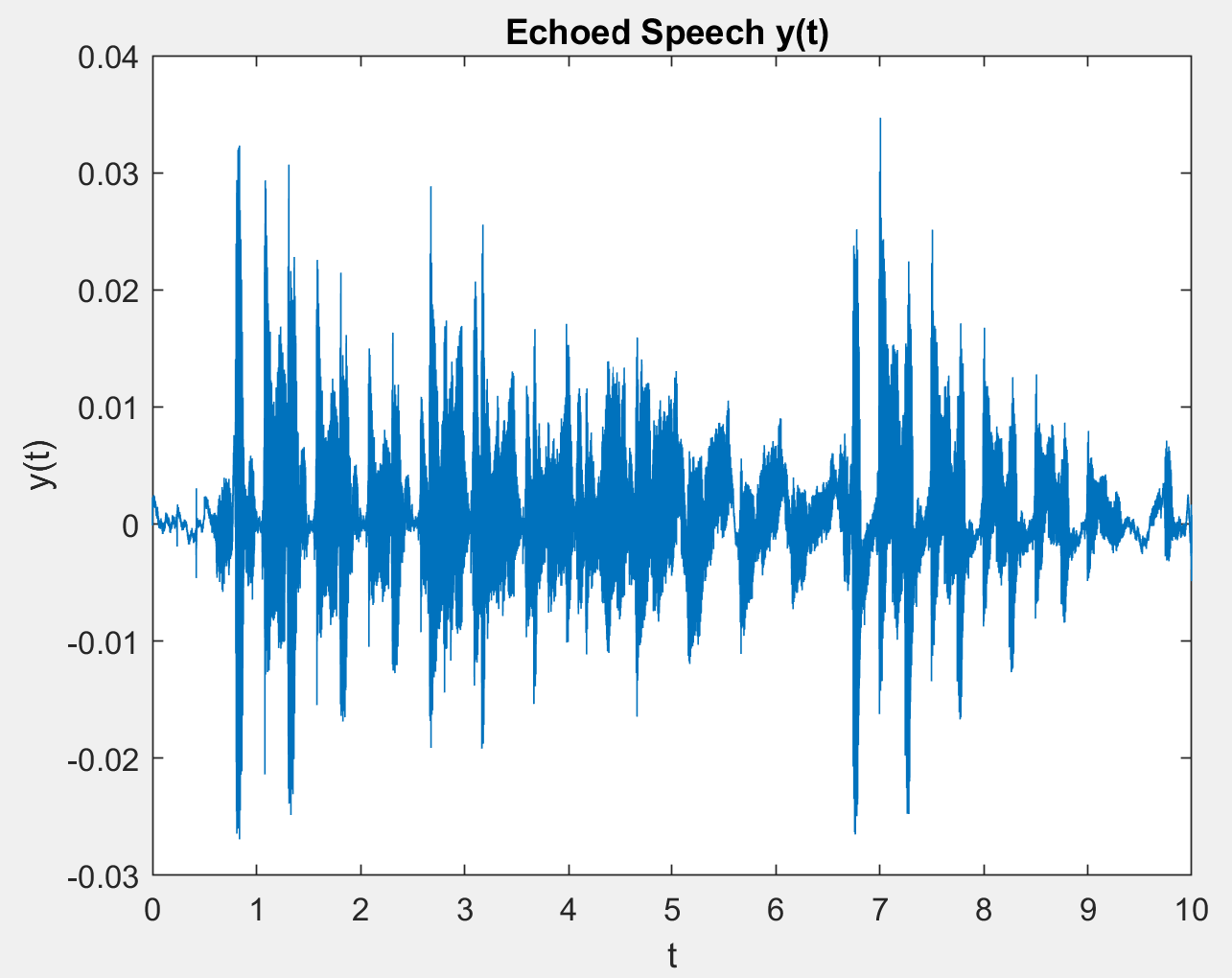
title("Estimation of the original function xe(t)(ECHO ELIMINATED)");

xlabel("t");

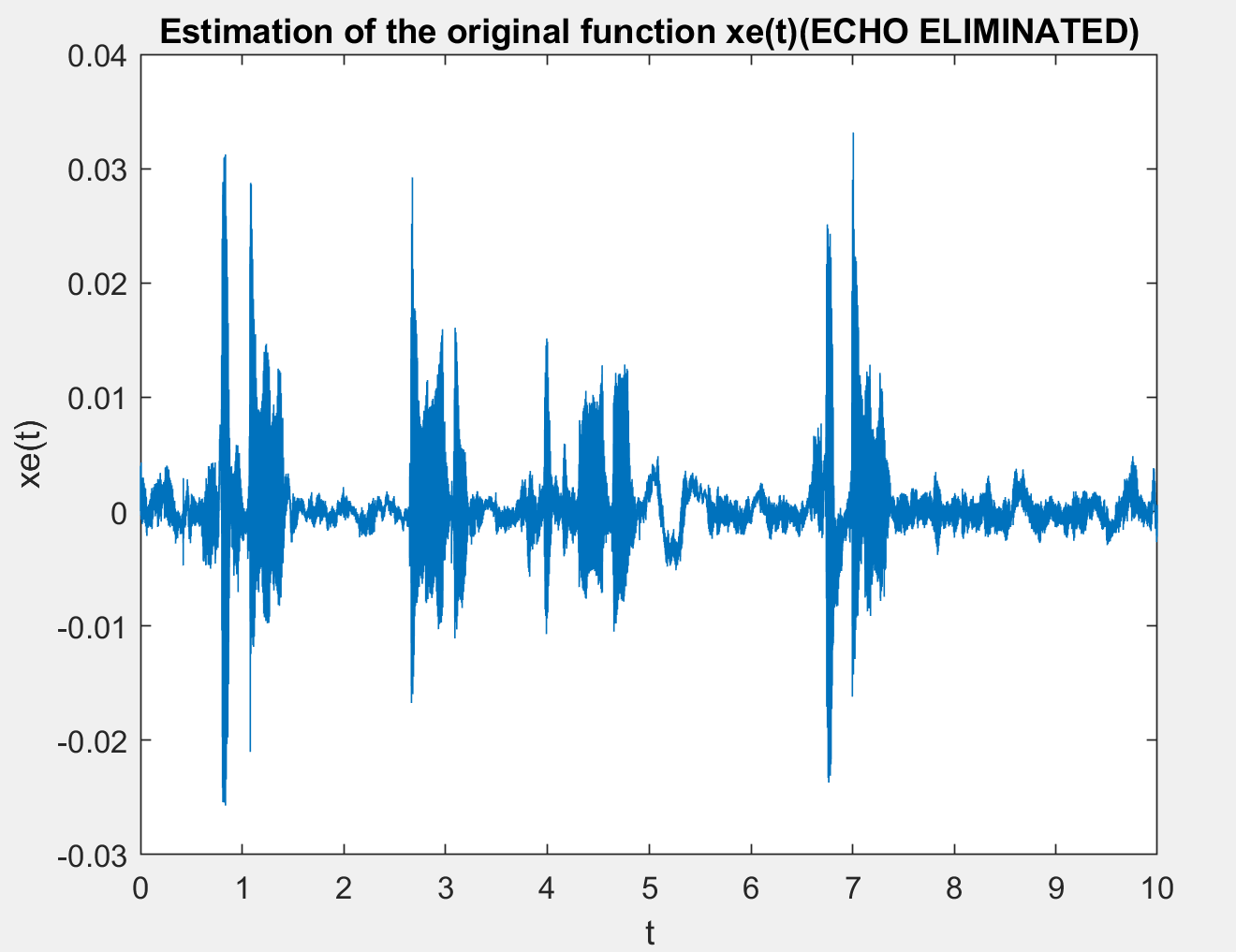
ylabel("xe(t)");



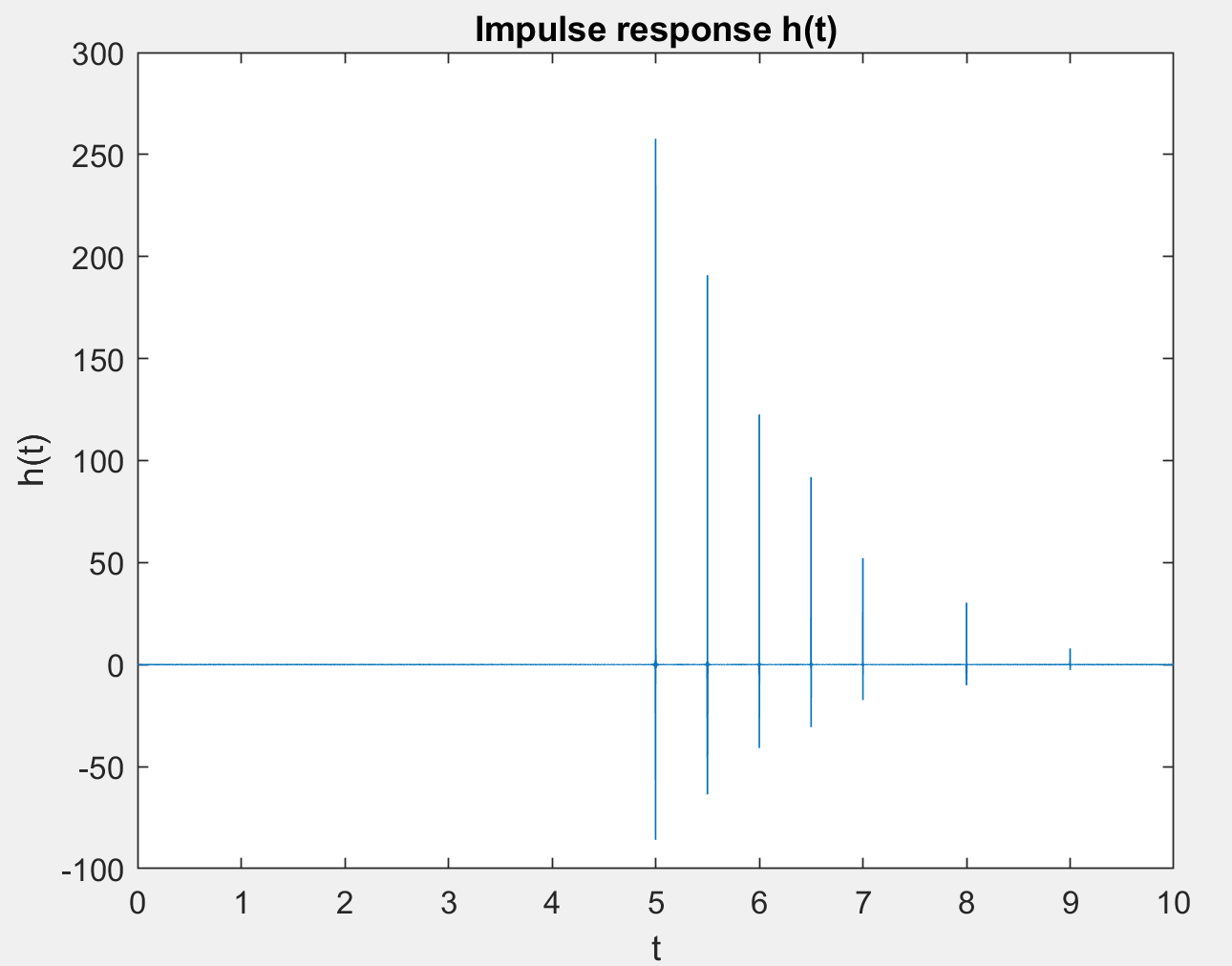
The input signal recorded with MATLAB



Artificial echo is added to my speech



Echo is eliminated – Original signal is recovered



**ON LAB**

**1)**

interval = 0:(1/8192):2.5-(1/8192);

g = gaussian(0.1,interval,1.25); %%%%%%%%% Generated Gaussian Function

plot(interval, g)

function[x]=gaussian(a,t,t0)

x= exp((-(t-t0).^2)/(2\*a.^2));

end

**3,4)**

clc;

CONSTANT = 8192;

x = on\_lab\_assignment\_3;

X = FT(x);

omega=linspace(-8192\*pi,8192\*pi,20481);

omega=omega(1:20480);

interval = 0:(1/8192):2.5-(1/8192);

figure(1)

plot(omega, abs(X));

title("Fourier Transform of plot of the given file ");

xlabel("w");

ylabel("X(jw))");

%%%%%%%%%% First Digit %%%%%%%%%

g1 = gaussian(0.1,interval,0.25);

x1= x .\*g1;

X1 = FT(x1)

figure(2)

plot(omega, abs(X1));

title("Finding the #ONE Digit ");

xlabel("w");

ylabel("X1(jw)");

%%%%%%%%%% Second Digit %%%%%%%%%

g2 = gaussian(0.1,interval,0.75);

x2= x .\*g2;

X2 = FT(x2)

figure(3)

plot(omega, abs(X2));

title("Finding the #TWO Digit ");

xlabel("w");

ylabel("X2(jw)");

%%%%%%%%%% Third Digit %%%%%%%%%

g3 = gaussian(0.1,interval,1.25);

x3= x .\*g3;

X3 = FT(x3)

figure(4)

plot(omega, abs(X3));

title("Finding the #THREE Digit ");

xlabel("w");

ylabel("X3(jw)");

%%%%%%%%%% Fourth Digit %%%%%%%%%

g4 = gaussian(0.1,interval,1.75);

x4= x .\*g4;

X4 = FT(x4)

figure(5)

plot(omega, abs(X4));

title("Finding the #FOUR Digit ");

xlabel("w");

ylabel("X4(jw)");

%%%%%%%%%% Fifth Digit %%%%%%%%%

g5 = gaussian(0.1,interval,2.25);

x5= x .\*g5;

X5 = FT(x5)

figure(6)

plot(omega, abs(X5));

title("Finding the #FIVE Digit ");

xlabel("w");

ylabel("X5(jw)");

%%%%%%% Functions %%%%%%%

function[x]=gaussian(a,t,t0)

x= exp((-(t-t0).^2)/(2\*a.^2));

end

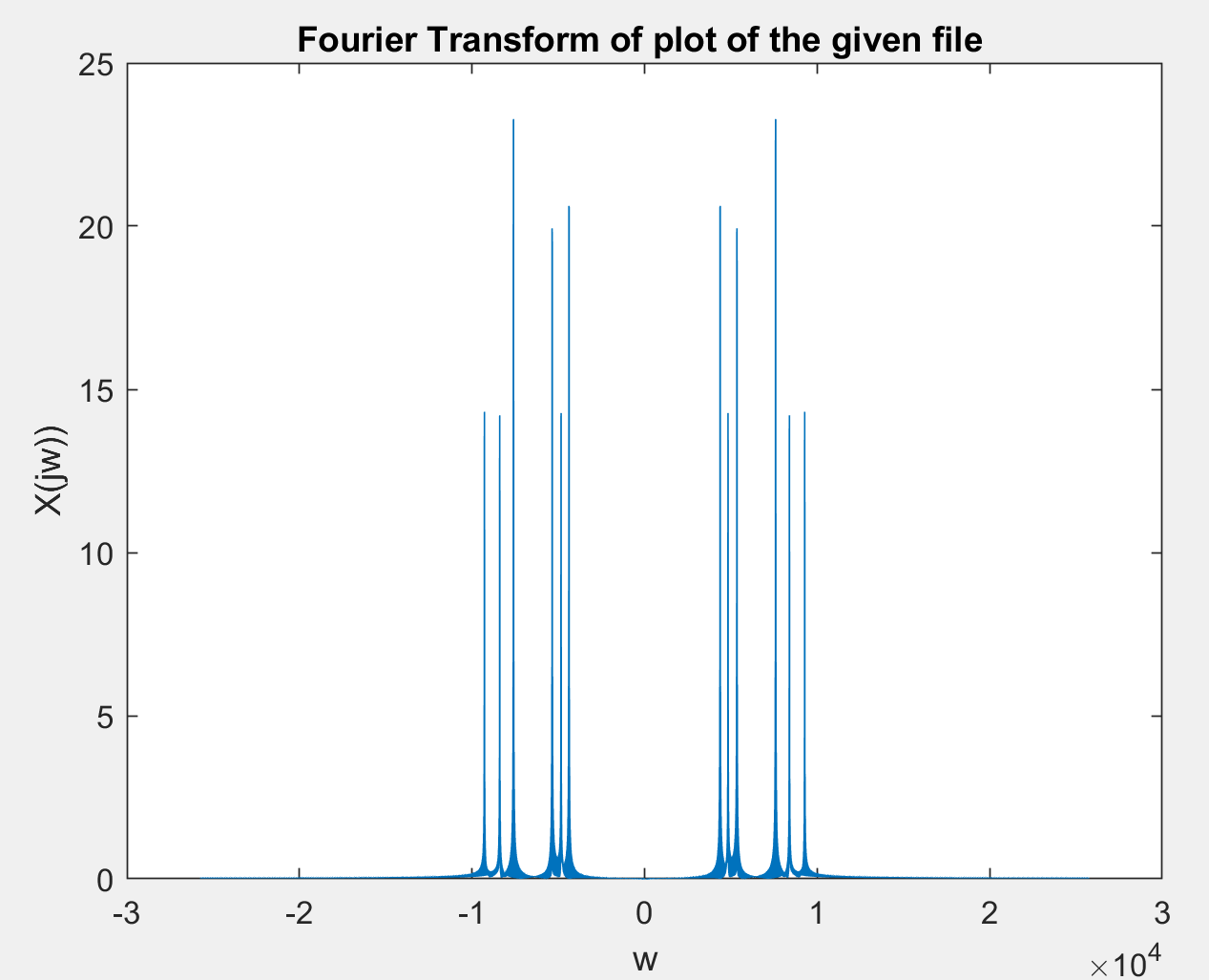
function output=FT(input)

M=length(input);

t=exp(j\*pi\*(M-1)/M\*[0:1:M-1]);

output=exp(-j\*pi\*(M-1)^2/(2\*M))\*t.\*1/(M)^0.5.\*fft(input.\*t);

end



We can’t decide individual digits from this

