Department of Electrical, Computer, & Biomedical Engineering
Faculty of Engineering & Architectural Science

Course Number	ELE532
Course Title	Signals and Systems I
Semester/Year	F2025
Instructor	Soosan Beheshti
TA Name	Dhanisha Trivedi

Lab/Tutorial Report No.	4
Report Title	The Fourier Transform: Properties and Applications

Submission Date	November, 22nd 2025
Due Date	November, 23rd 2025

Student Name	Student ID	Signature
Dian Zahid	501253063	D.Z
Hamzah Sahi	501221955	H.S

Introduction

This lab explores how the Fourier Transform is used to analyze, transmit, and recover signals in a basic communication system. We examine the frequency characteristics of the speech signal, two low pass filters, and the channel response. Using this information, we design a simple modulation demodulation scheme in matlab to shift the speech signal into the channel's passband, transmit it, and then recover it at the receiver. The goal is to understand how modulation, filtering, and Fourier analysis work together to enable effective signal transmission.

Part A)

A.1

```
>> N = 100;
PulseWidth = 10;
t = 0:1:(N-1);
x = [ones(1,PulseWidth), zeros(1,N-PulseWidth)];
z = conv(x, x);
tz = 0:(length(z)-1);

figure;
stairs(tz, z);
title('z(t) = x(t) * x(t)');
xlabel('t');
ylabel('Amplitude');
grid on;
>>
```

Figure A.1.1: A.1 Code

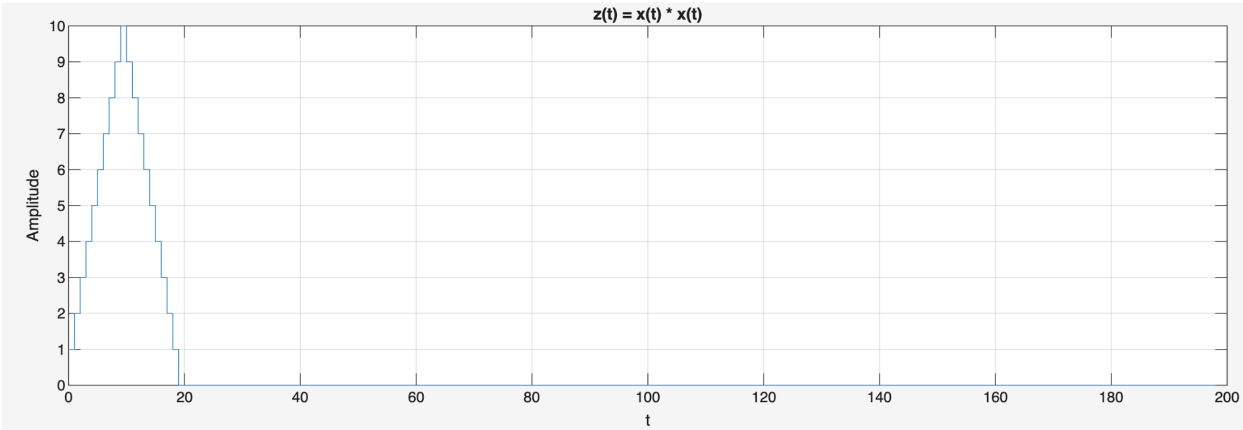

Figure A.1.2: A.1 Plot

A.2

A.3

```
>> f = [-(N/2):(N/2)-1]*(1/N);
figure;
subplot(2,1,1);
plot(f, fftshift(abs(Zf)));
title('Magnitude Spectrum of z(t)');
xlabel('Frequency (Hz)');
ylabel('|Z(f)|');
grid on;

subplot(2,1,2);
plot(f, fftshift(angle(Zf)));
title('Phase Spectrum of z(t)');
xlabel('Frequency (Hz)');
ylabel('∠Z(f)');
grid on;
>>
```

Figure A.3.1: A.3 Code

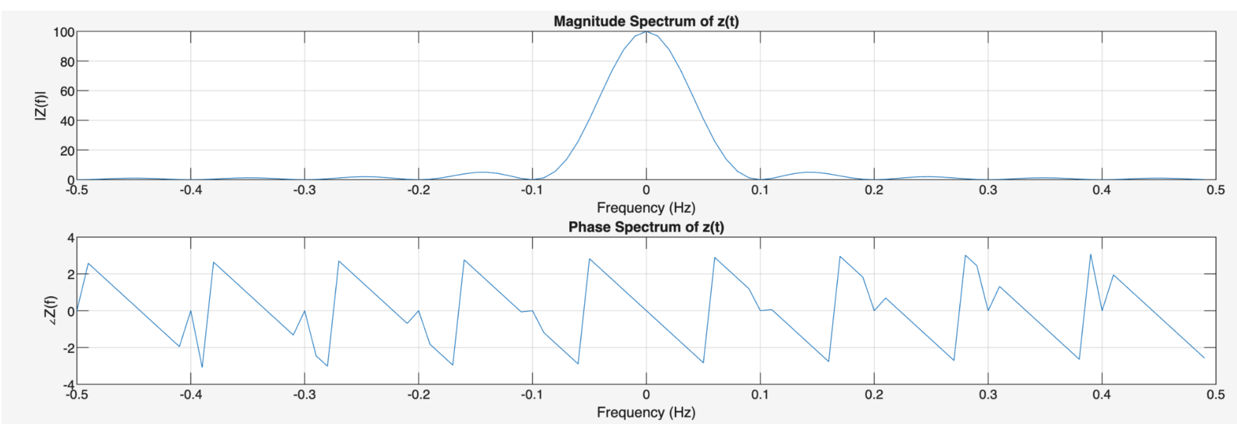

Figure A.3.2: A.3 Plot

A.4

```
>> % Time-domain convolution
z_time = conv(x, x);

% Frequency-domain multiplication
Xf = fft(x);
Zf = Xf .* Xf;
z_freq = ifft(Zf);

z_freq = real(z_freq);
z_freq = [z_freq, zeros(1,length(z_time)-length(z_freq))];

figure;
subplot(2,1,1);
stairs(z_time);
title('z(t) from time-domain convolution');
grid on;

subplot(2,1,2);
stairs(z_freq);
title('z(t) from frequency-domain multiplication');
grid on;
>>
```

Figure A4.1: A4 Code

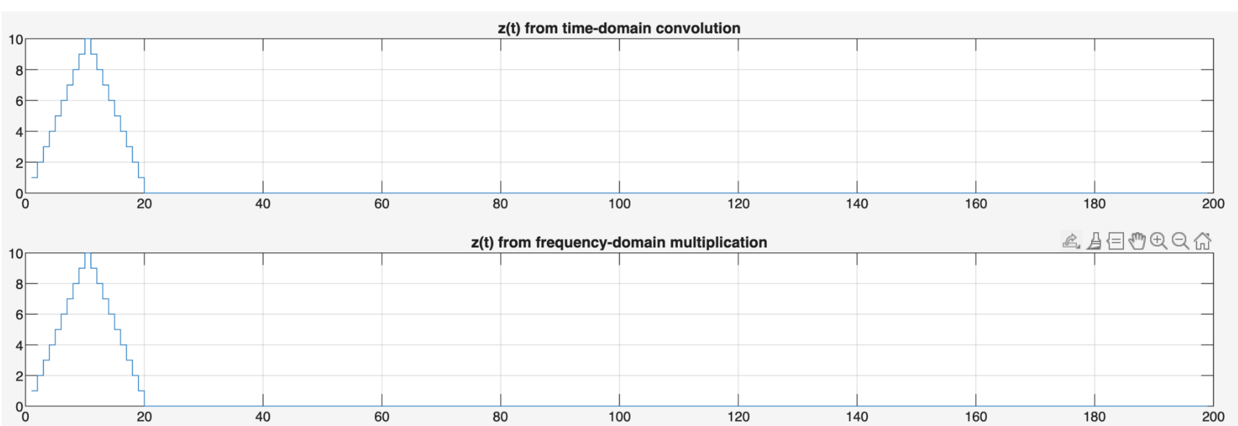

Figure A4.2: A4 Plot

The result from Problem A.1 predicts that $z(t) = x(t) * x(t)$ should be a triangular waveform increasing from 0 to 10 and decreasing back to 0 at $t = 20$. Both matlab implementations, the time-domain convolution and the frequency-domain multiplication produced the same triangular signal with identical amplitude, duration, and shape. Therefore, the matlab results match the result from A.1. This demonstrates the Convolution Theorem of the Fourier Transform.

A.5

```
% Problem A.5
N = 100;
t = 0:(N-1);
f = (-(N/2):(N/2-1))*(1/N);

% Pulse width = 5
PulseWidth = 5;
x5 = [ones(1, PulseWidth), zeros(1, N-PulseWidth)];
X5 = fft(x5);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(X5))); grid on;
title('Magnitude Spectrum, Pulse Width = 5');

subplot(2,1,2);
plot(f, fftshift(angle(X5))); grid on;
title('Phase Spectrum, Pulse Width = 5');

% Pulse width = 10
PulseWidth = 10;
x10 = [ones(1, PulseWidth), zeros(1, N-PulseWidth)];
X10 = fft(x10);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(X10))); grid on;
title('Magnitude Spectrum, Pulse Width = 10');

subplot(2,1,2);
plot(f, fftshift(angle(X10))); grid on;
title('Phase Spectrum, Pulse Width = 10');

% Pulse width = 25
PulseWidth = 25;
x25 = [ones(1, PulseWidth), zeros(1, N-PulseWidth)];
X25 = fft(x25);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(X25))); grid on;
title('Magnitude Spectrum, Pulse Width = 25');

subplot(2,1,2);
plot(f, fftshift(angle(X25))); grid on;
title('Phase Spectrum, Pulse Width = 25');
```

Figure A.5: MATLAB Implementation

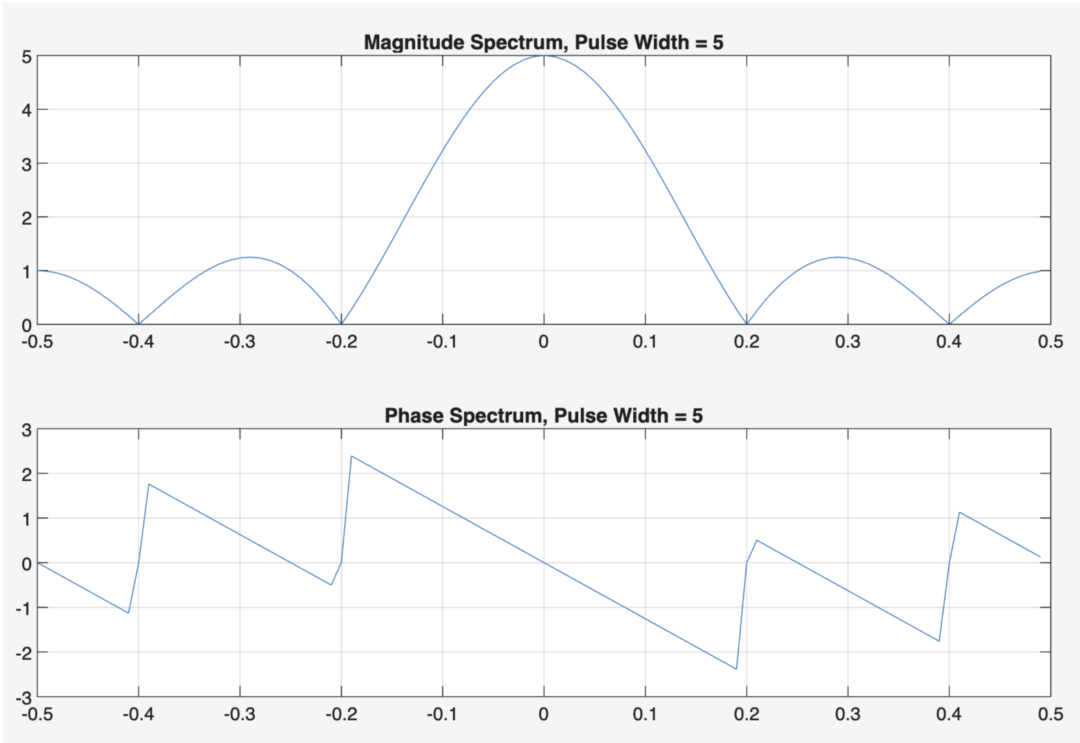

Figure A.5: Plots - Pulse width = 5

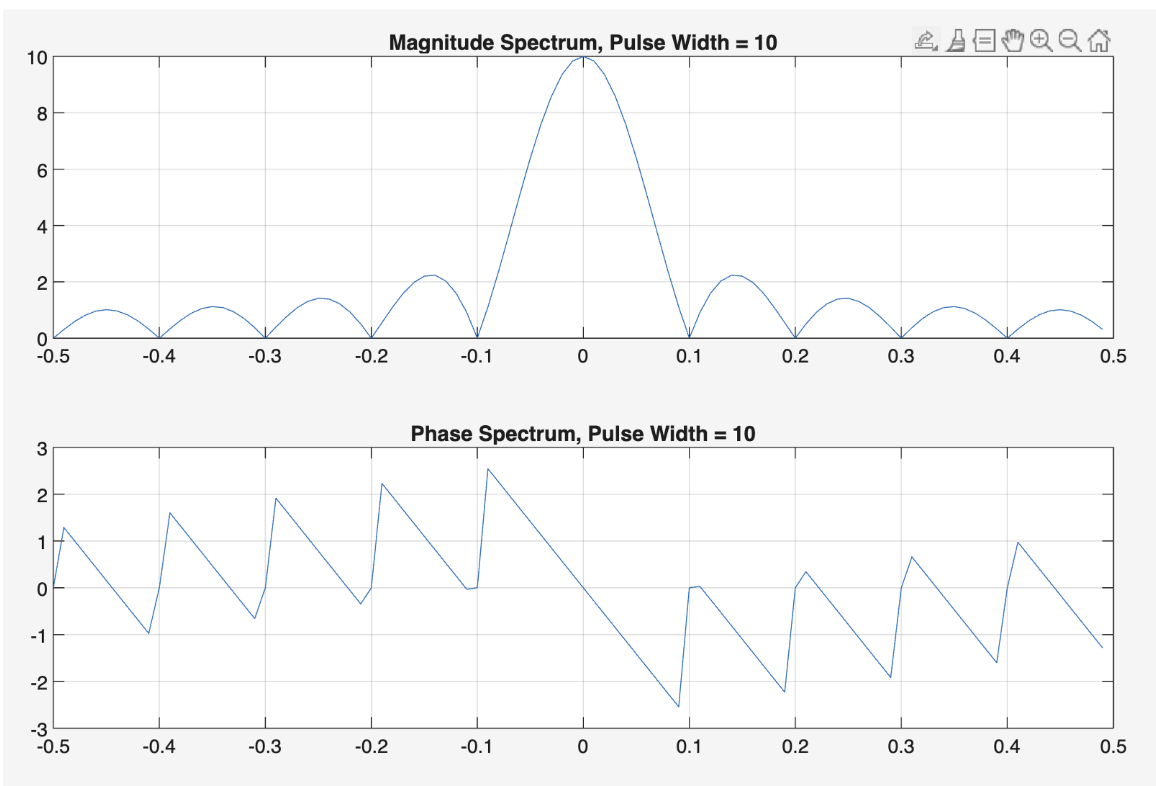

Figure A.5: Plots - Pulse width = 10

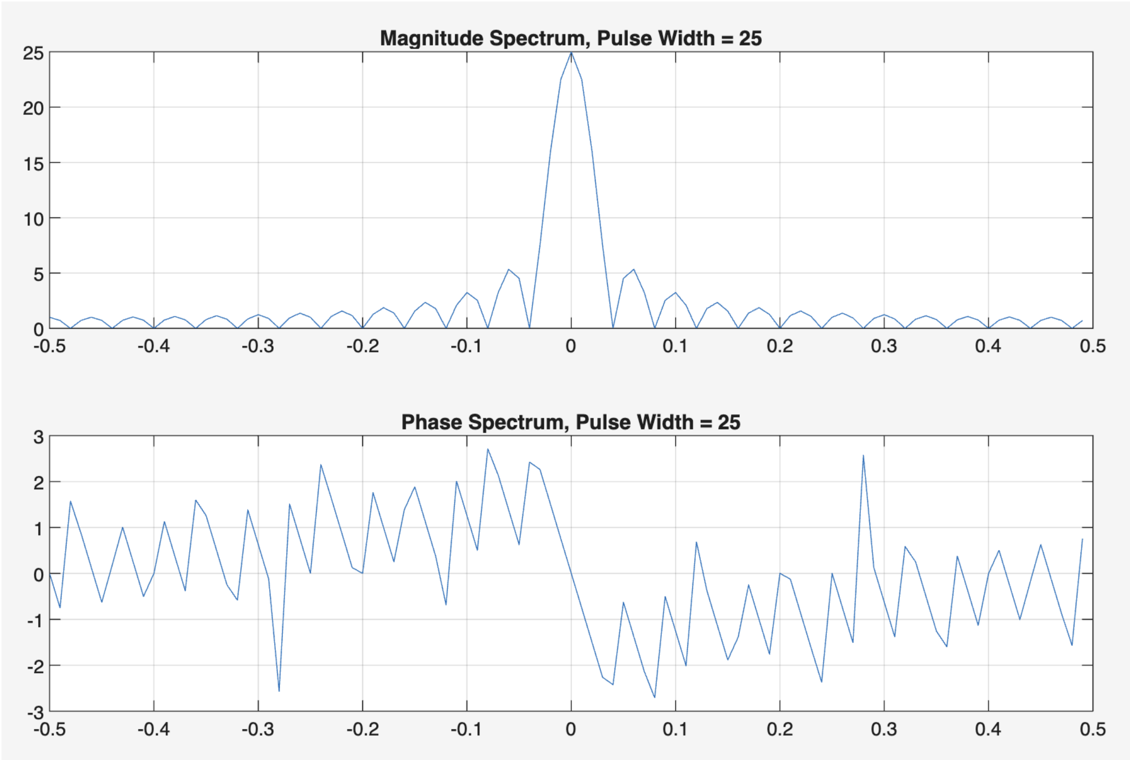

Figure A.5: Plots - Pulse width = 25

A.5 Explain which property of the Fourier Transform you have demonstrated.

Changing the pulse width shows that shorter pulses have much wider spectra, while longer pulses have narrower spectra. The width-5 pulse spreads out the most, and the width-25 pulse is the most concentrated. This happens because fast changes in time create more high-frequency content. This demonstrates the time–bandwidth property: narrow in time → wide in frequency, and wide in time → narrow in frequency.

A.6

```
% Problem A.6
N = 100;
PulseWidth = 10;
t = 0:(N-1);
x = [ones(1, PulseWidth), zeros(1, N-PulseWidth)];
f = (-(N/2):(N/2-1))*(1/N);

% w_plus(t) = x(t)*exp(j*pi/3*t)
w_plus = x .* exp(1j*(pi/3)*t);
W_plus = fft(w_plus);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(W_plus))); grid on;
title('Magnitude Spectrum of w_+(t)');

subplot(2,1,2);
plot(f, fftshift(angle(W_plus))); grid on;
title('Phase Spectrum of w_+(t)');

% w_minus(t) = x(t)*exp(-j*pi/3*t)
w_minus = x .* exp(-1j*(pi/3)*t);
W_minus = fft(w_minus);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(W_minus))); grid on;
title('Magnitude Spectrum of w_-(t)');

subplot(2,1,2);
plot(f, fftshift(angle(W_minus))); grid on;
title('Phase Spectrum of w_-(t)');

% w_c(t) = x(t)*cos(pi/3*t)
w_c = x .* cos((pi/3)*t);
W_c = fft(w_c);

figure;
subplot(2,1,1);
plot(f, fftshift(abs(W_c))); grid on;
title('Magnitude Spectrum of w_c(t) = x(t)cos(\pi t / 3)');

subplot(2,1,2);
plot(f, fftshift(angle(W_c))); grid on;
title('Phase Spectrum of w_c(t)');
```

Figure A.6: MATLAB Implementation

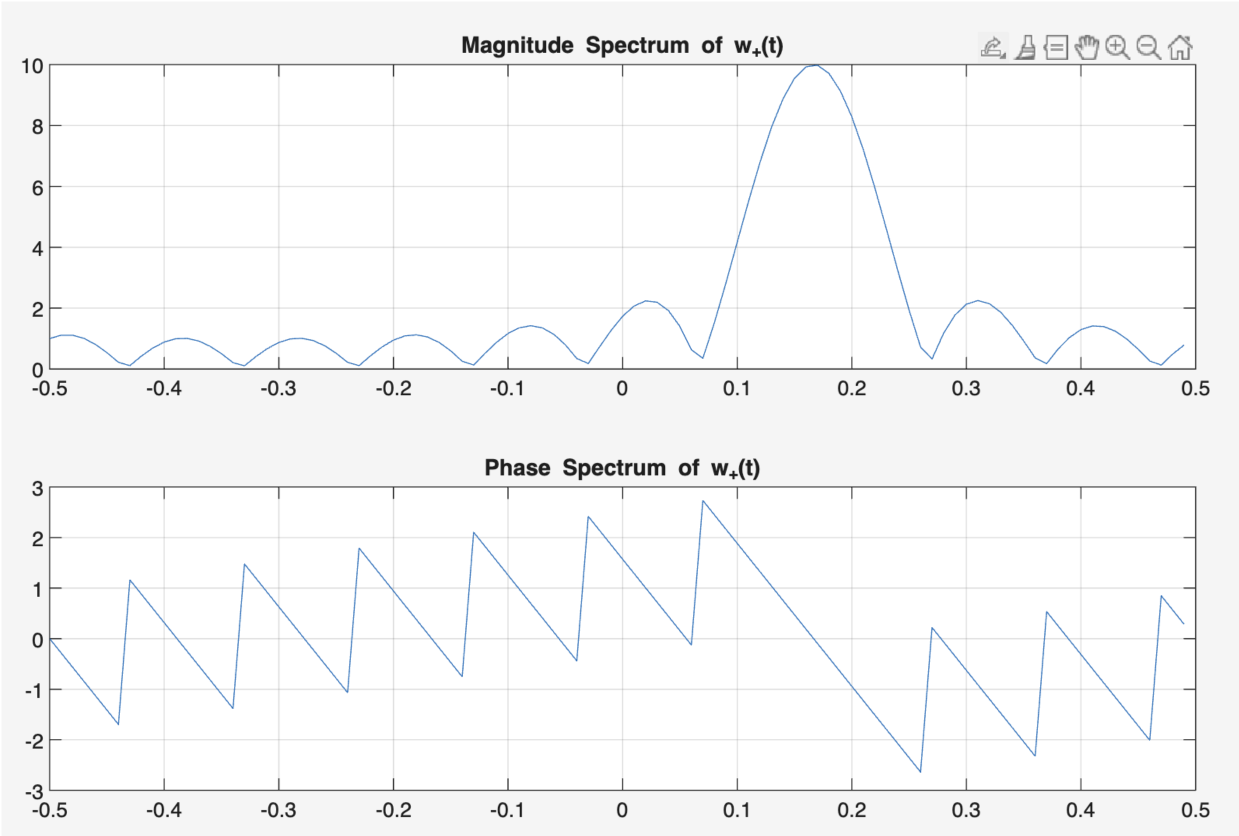

Figure A.6: Plots - w+(t)

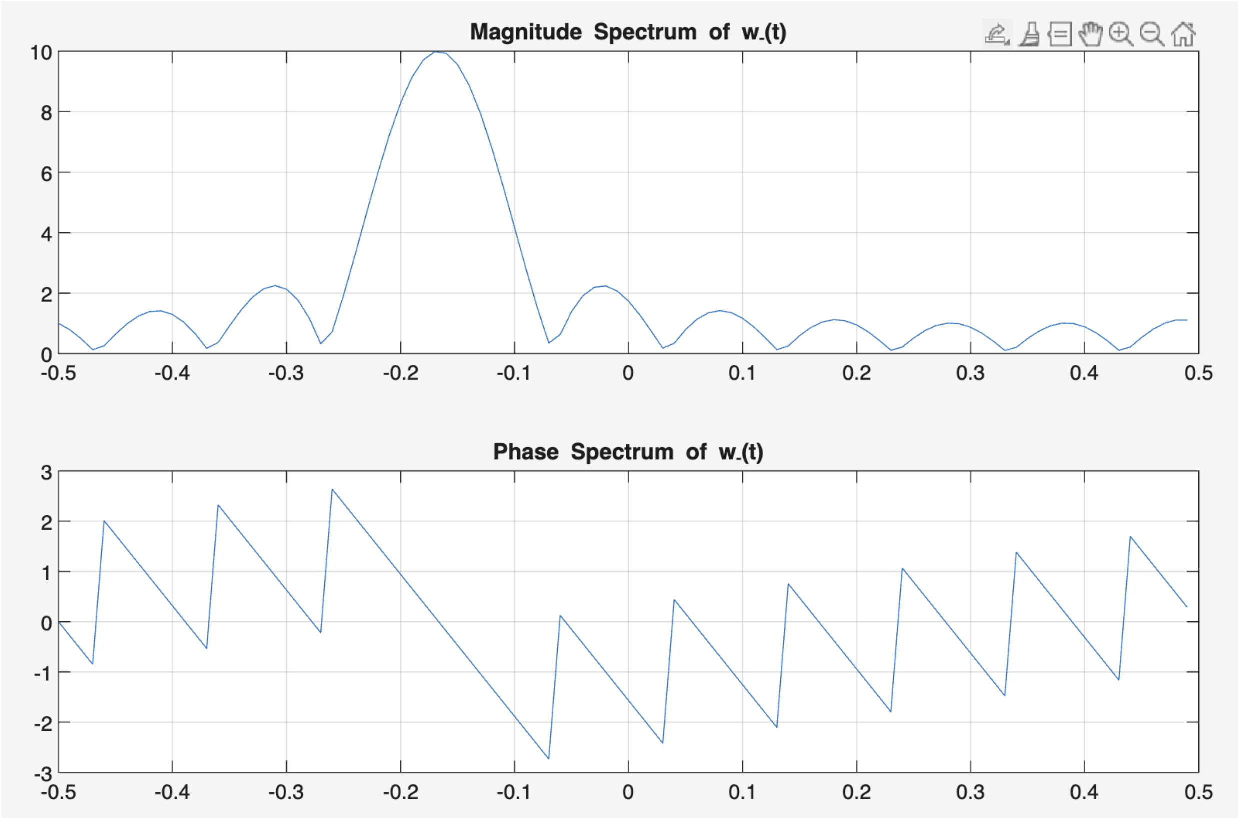

Figure A.6: Plots - w_(t)

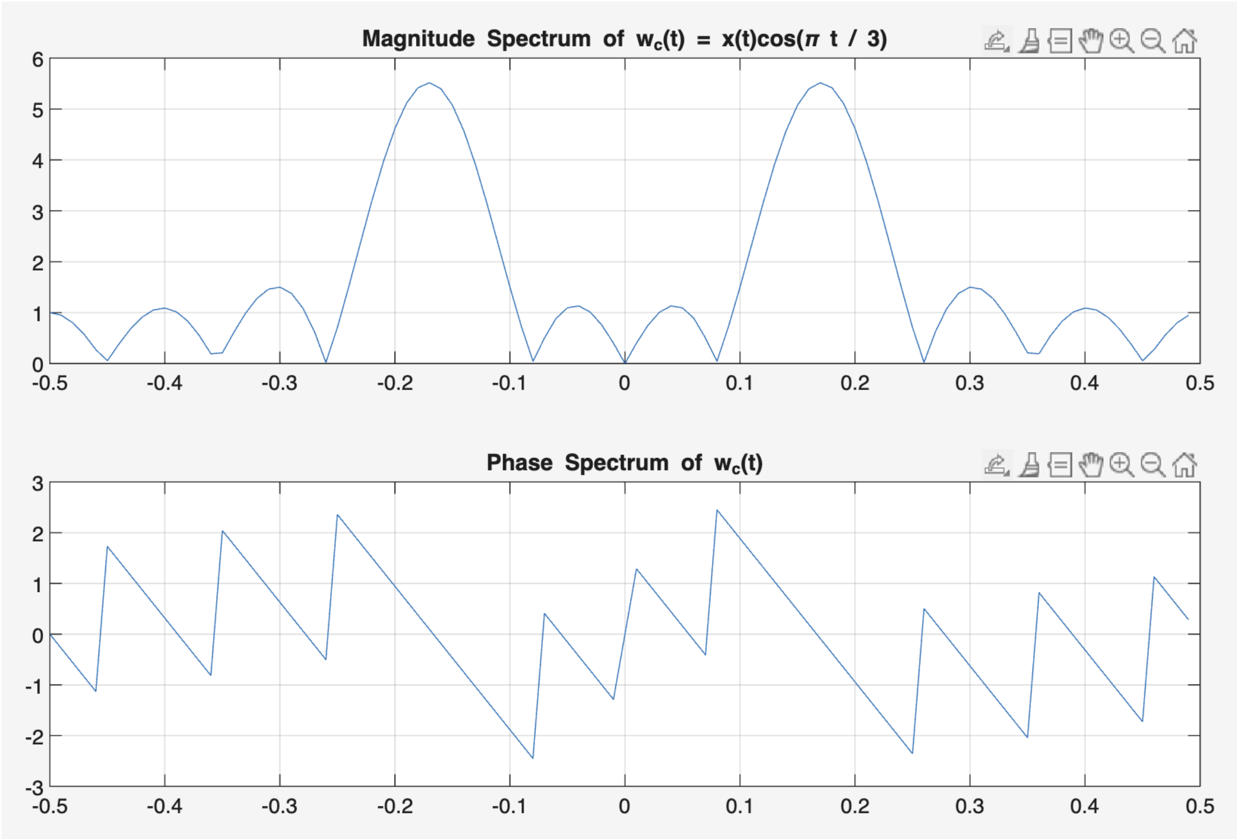

Figure A.6: Plots - Wc(t)

<u>A.6 Explain which property of the Fourier Transform you have demonstrated.</u>

Multiplying the pulse by a complex exponential shifts its spectrum. w+(t) shifts the spectrum to the right, and w−(t) shifts it to the left. Multiplying by a cosine creates two shifted copies since a cosine is made of two exponentials. This demonstrates the modulation property of the Fourier Transform, where multiplying by a sinusoid moves the spectrum in frequency.

Part B)

```matlab
% Part B
clear; close all; clc;

load('Lab4_Data.mat');

xspeech   = xspeech(:);
hLPF2000  = hLPF2000(:);
hLPF2500  = hLPF2500(:);
hChannel  = hChannel(:);

Fs = 32000;

figure(1);
MagSpect(xspeech);
title('xspeech');

figure(2);
MagSpect(hLPF2000);
title('hLPF2000');

figure(3);
MagSpect(hLPF2500);
title('hLPF2500');

figure(4);
MagSpect(hChannel);
title('hChannel');

C1 = osc(3500, 80000);
Y1 = C1 .* xspeech(1:80000);

figure(5);
MagSpect(Y1);
title('Y1 = Modulated Signal');

pass = conv(Y1, hChannel);

figure(6);
MagSpect(pass);
title('pass = Y1 * hChannel');

C2 = osc(3500, length(pass));
Y2 = C2 .* pass;

figure(7);
MagSpect(Y2);
title('Y2 = Demodulated Signal');

result1 = conv(Y2, hLPF2000);
result2 = conv(Y2, hLPF2500);

figure(8);
MagSpect(result1);
title('Recovered Output (LPF2000)');

figure(9);
MagSpect(result2);
title('Recovered Output (LPF2500)');
```

Figure B: MATLAB Implementation

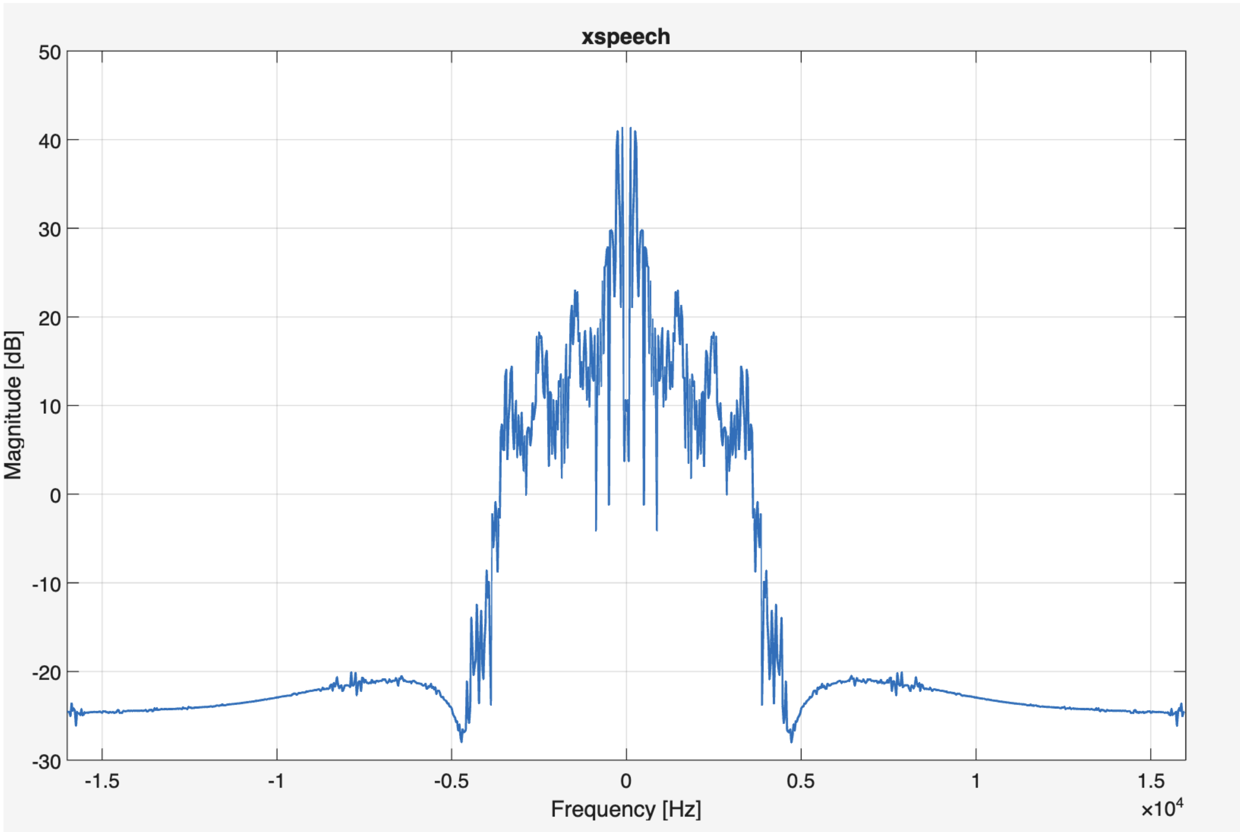

Figure B: Plots - xspeech

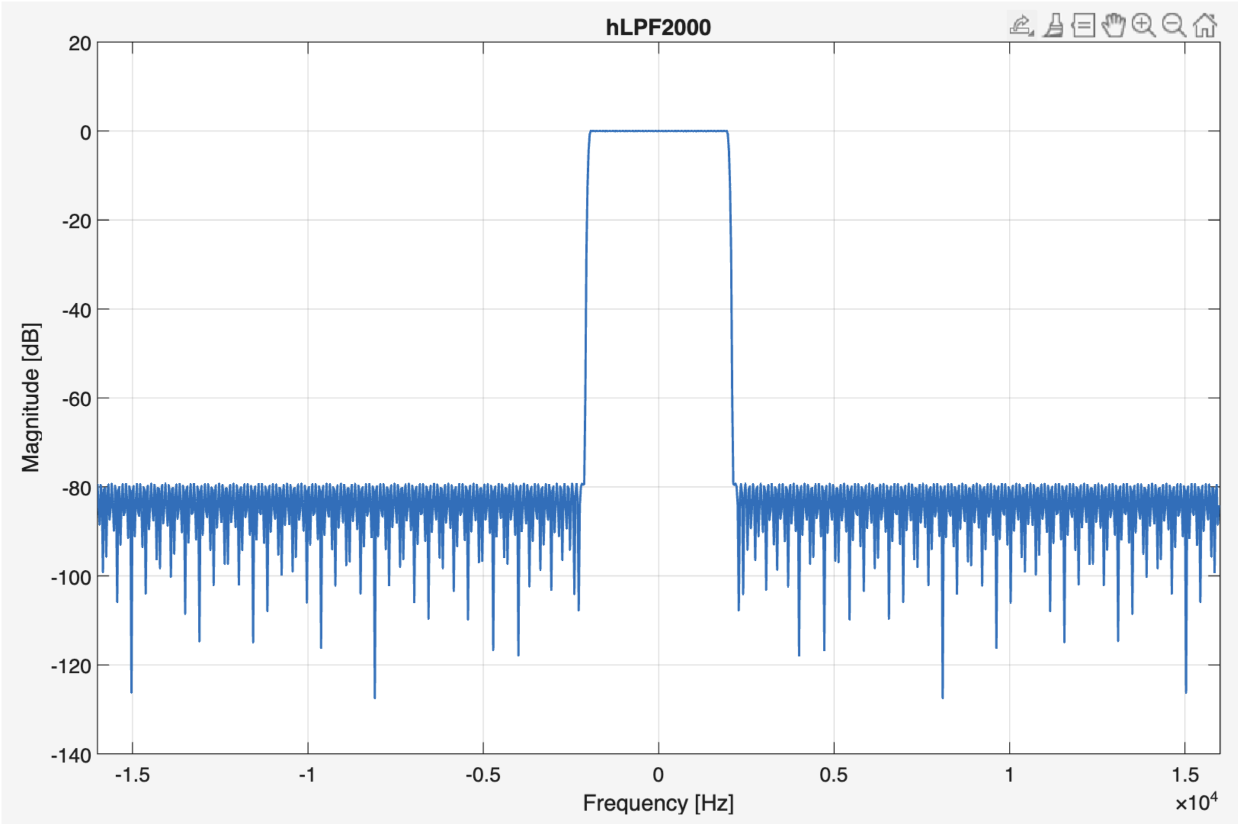

Figure B: Plots - hLPF2000

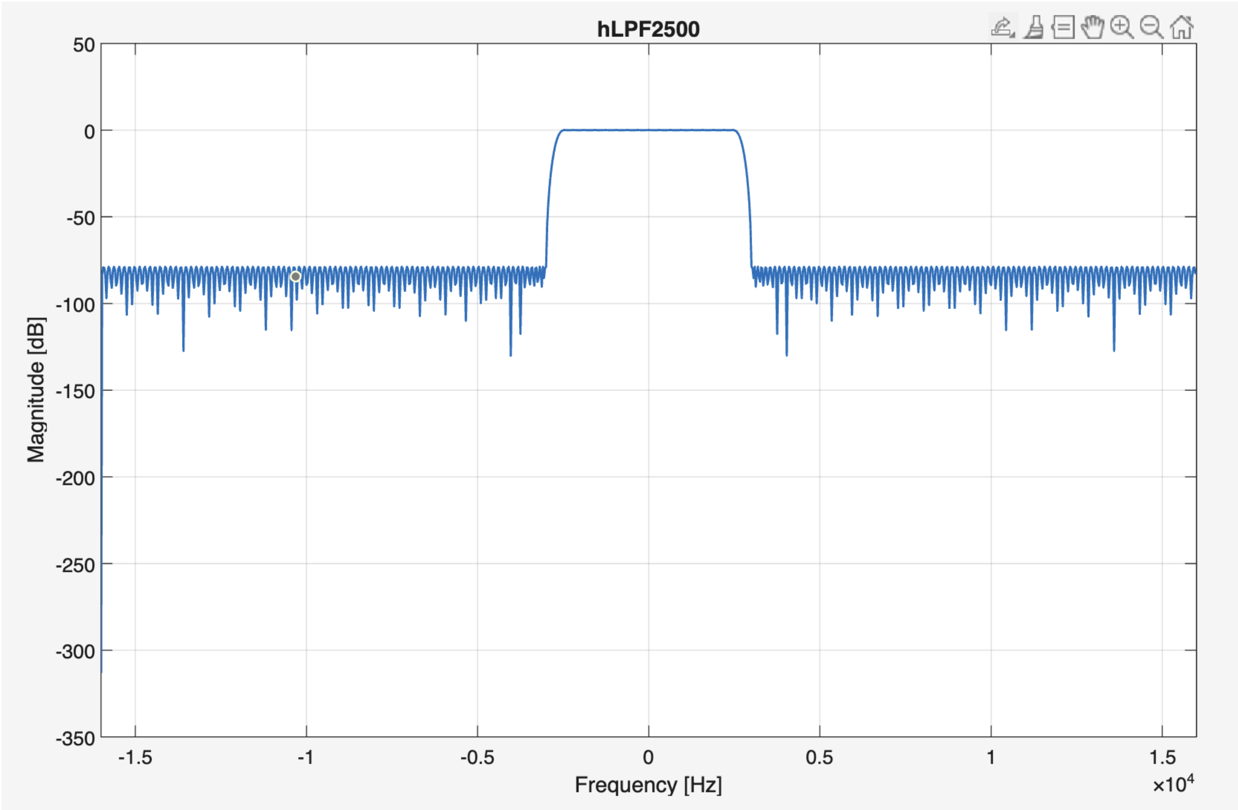

Figure B: Plots - hLPF2500

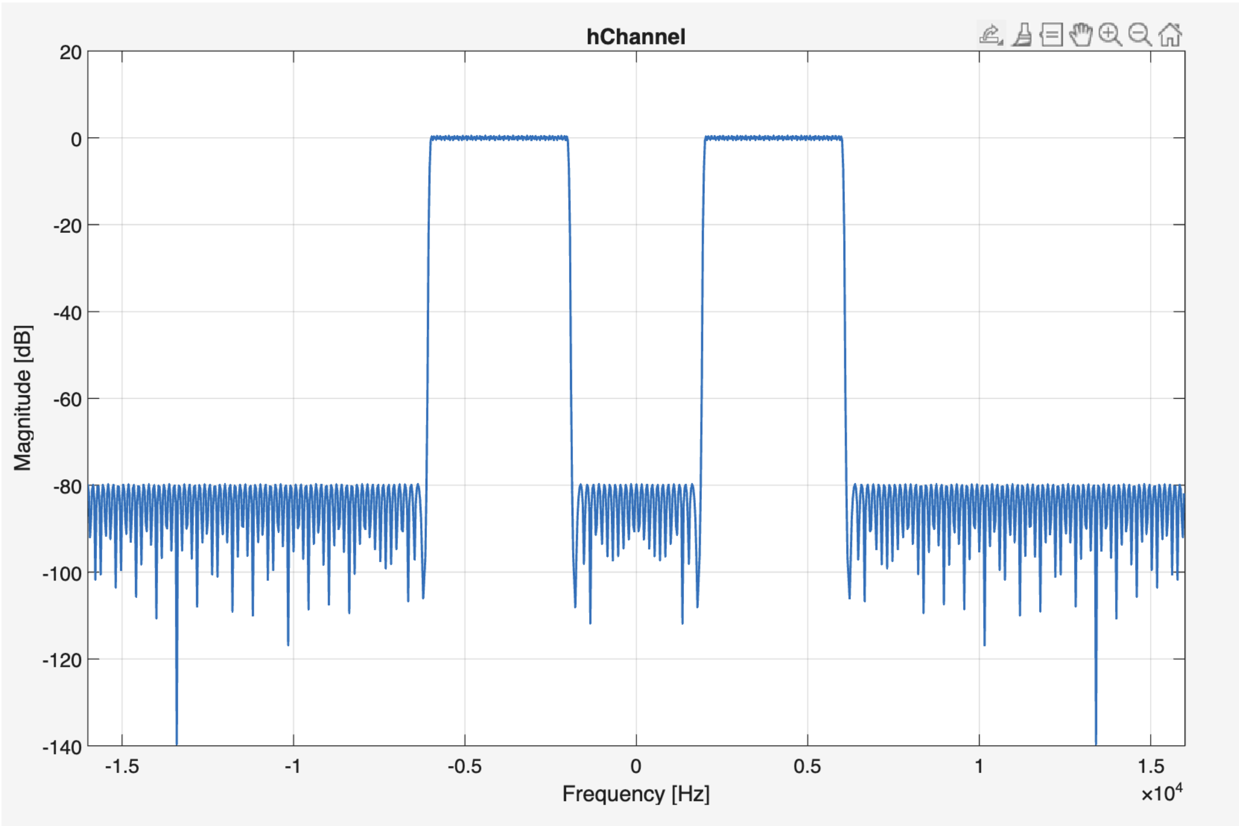

Figure B: Plots - hChannel

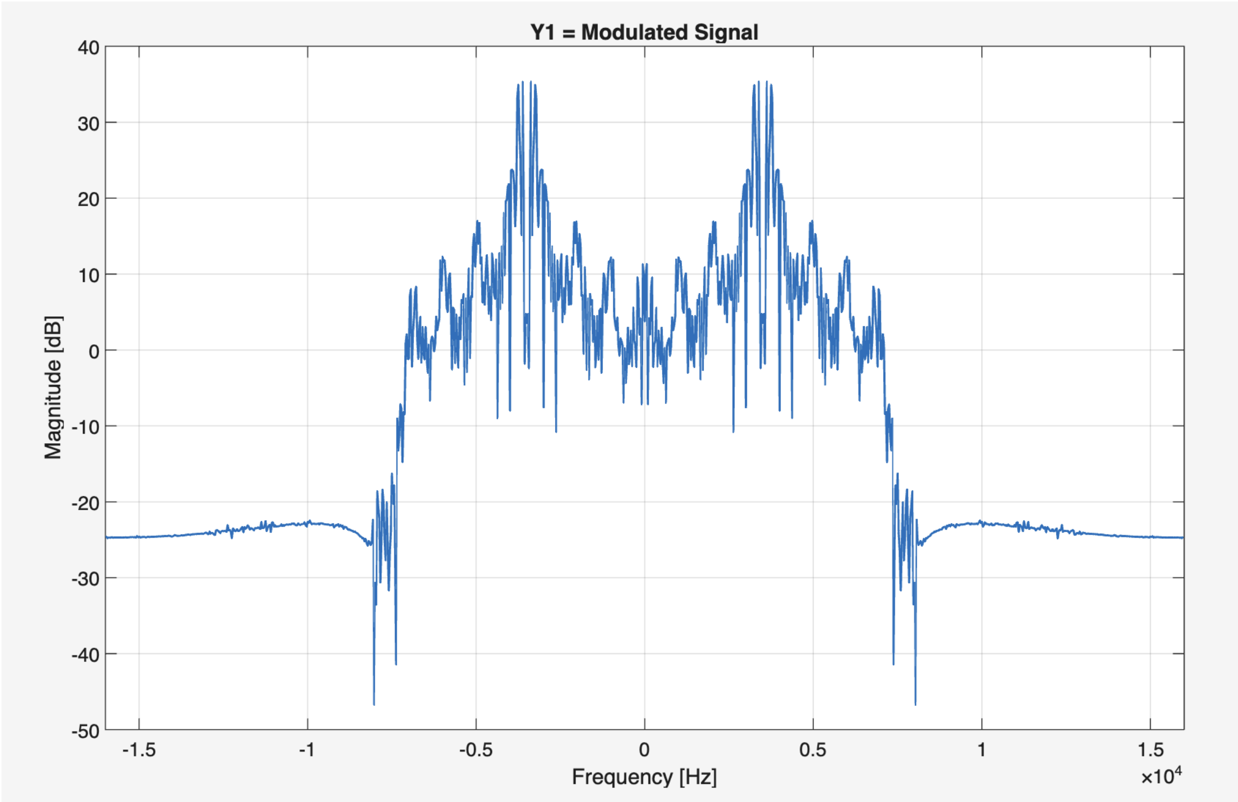

Figure B: Plots - Y1 = Modulated Signal

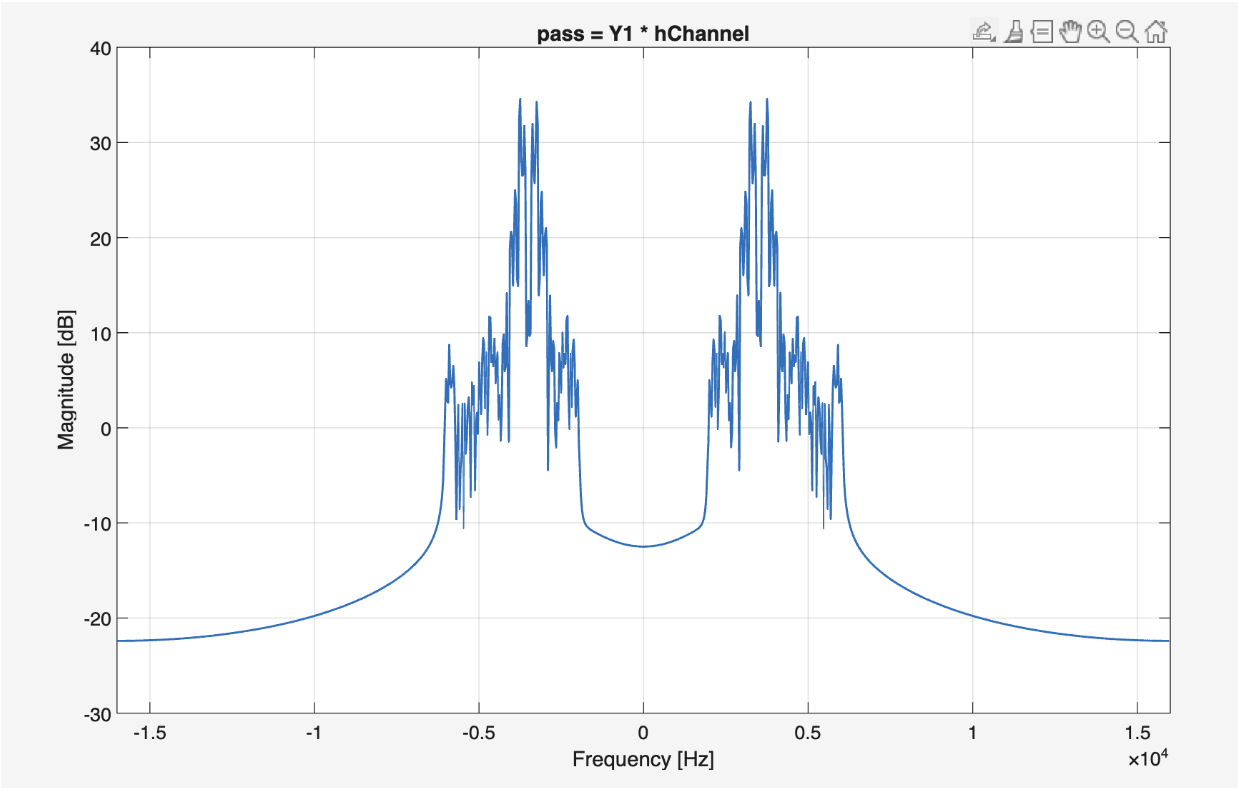

Figure B: Plots - pass = Y1 * hChannel

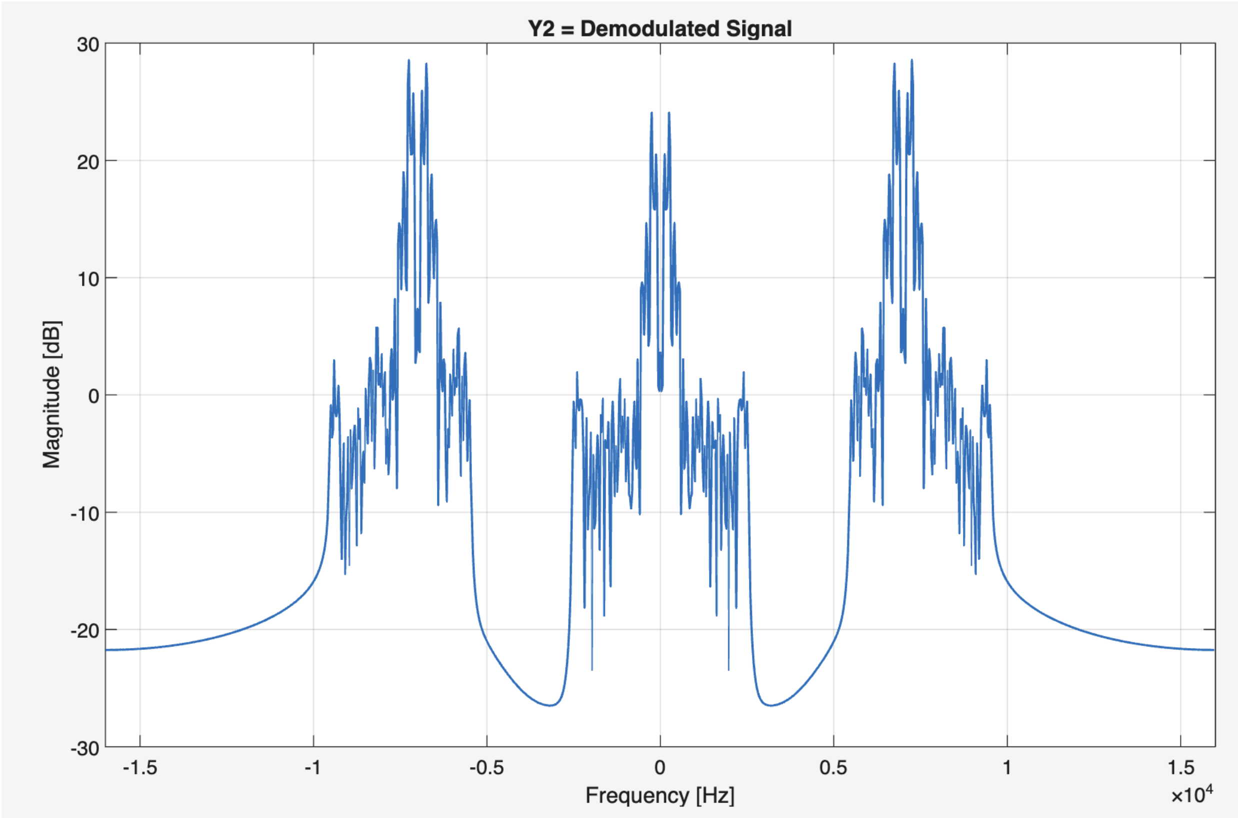

Figure B: Plots - Y2 = Demodulated Signal

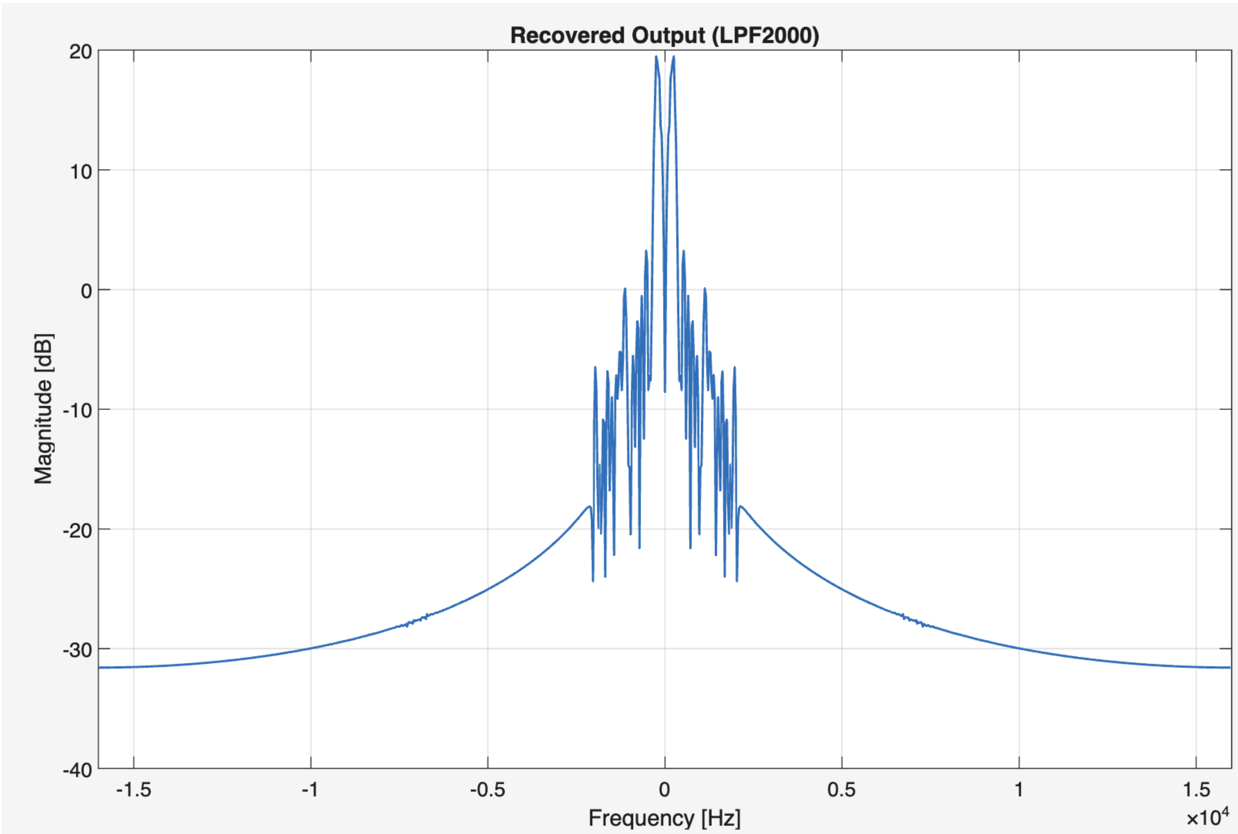

Figure B: Plots - Recovered Output (LPF2000)

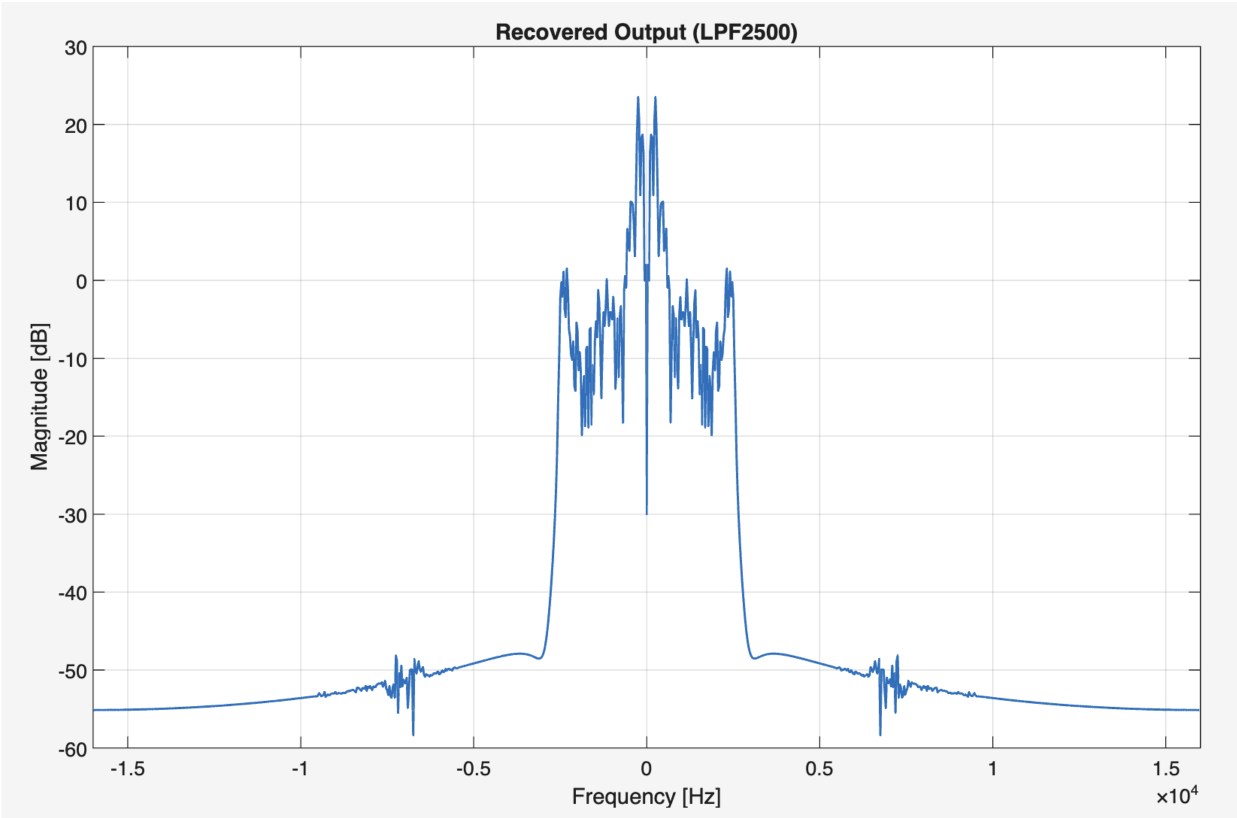

Figure B: Plots - Recovered Output (LPF2500)

Part B: explain the rationale of your design

The approach used in this problem was to modulate the xspeech signal by multiplying it with a cosine carrier at 3500 Hz, which shifts the speech spectrum into the passband of the channel. The modulated signal was then passed through the hChannel filter to simulate transmission. At the receiver, the signal was multiplied by the same carrier to shift it back down to baseband. Finally, the demodulated signal was filtered using the 2.5 kHz low pass filter to remove the high frequency components and recover the speech.

Conclusion

In this lab, the speech signal was successfully transmitted through the channel using cosine based modulation and recovered through demodulation and low pass filtering. The Fourier spectra confirmed how modulation shifts the signal in frequency and how the channel and filters shape the transmitted and received signals. The final recovered output demonstrates the usefulness of frequency domain analysis in communication systems by demonstrating that the designed coder and decoder function as intended.