Part 1.1

Sound of this signal sounds very familiar. It's like the click sound of the old phones.

Part 1.2 (w/ using the properties)

($\rightarrow$ this denotes the conversion between time and freq. domain)

a) $x(t) = e^{j2\pi f_0 t} = e^{jw_0 t} \longrightarrow X(jw) = 2\pi\,\delta(w-w_0)$

b) $x(t) = \cos(2\pi f_0 t) = \cos(w_0 t) \longrightarrow X(jw) = \pi(\delta(w-w_0) + \delta(w+w_0))$

c) $x(t) = \text{rect}\left(\frac{t}{T_0}\right) = \begin{cases} 0 & \text{if } |t| > \frac{T_0}{2} \\ 1/2 & \text{if } |t| = T_0/2 \\ 1 & \text{if } |t| < T_0/2 \end{cases} \longrightarrow \int_{-T_0/2}^{T_0/2} 1 \cdot e^{-jwt}\,dt = \left.\frac{e^{-jwt}}{-jw}\right|_{-T_0/2}^{T_0/2}$

from Maple's def.

$= \frac{-(e^{-jw\frac{T_0}{2}} - e^{jw\frac{T_0}{2}})}{jw}$

$= \frac{2\sin(w\frac{T_0}{2})}{w}$

d) $x(t) = \underbrace{e^{j2\pi f_0 t}}_{x_1(t)} \cdot \underbrace{\text{rect}\left(\frac{t}{T_0}\right)}_{x_2(t)} \rightarrow X_1(jw) * X_2(jw)$ (freq. shift)

$= X_2(j(w-w_0)) = \frac{2\sin((w-w_0)T_0/2)}{w-w_0}$

e) $x(t) = \cos(w_0 t)\,\text{rect}\left(\frac{t}{T_0}\right) = \frac{e^{jw_0 t} + e^{-jw_0 t}}{2} \cdot \text{rect}\left(\frac{t}{T_0}\right)$ (freq. shift)

$\longrightarrow \frac{\sin((w-w_0)T_0/2)}{w-w_0} + \frac{\sin((w+w_0)T_0/2)}{w+w_0}$

f) $x(t) = \text{rect}\left(\frac{t-t_0}{T_0}\right) \longrightarrow e^{-jwt_0} \cdot \frac{2\sin(w\frac{T_0}{2})}{w}$ (Time shift)

g) $x(t) = e^{jw_0 t} \cdot \text{rect}\left(\frac{t-t_0}{T_0}\right) \longrightarrow e^{-j(w-w_0)t_0} \cdot \frac{2\sin((w-w_0)T_0/2)}{w-w_0}$

h) $x(t) = \cos(w_0 t)\,\text{rect}\left(\frac{t-t_0}{T_0}\right) = e^{-j(w-w_0)} \cdot \frac{\sin((w-w_0)\frac{T_0}{2})}{w-w_0}$

$+ e^{-j(w+w_0)} \cdot \frac{\sin((w+w_0)\frac{T_0}{2})}{w+w_0}$

* Yes, the frequencies where the peak occur the ones used by DTMF transceivers.

from $w = 2\pi f$ we get

697 Hz → 4380 rad/sec	852 Hz → 5353 rad/sec
770 Hz → 4838 rad/sec	941 Hz → 5912 rad/sec
1209 Hz → 7596 rad/sec	1477 Hz → 9280 rad/sec
1336 Hz → 8394 rad/sec	1633 Hz → 10260 rad/sec

FT function computes one digit of the number.

from square wave properties

Second Digit Calculation: $\underset{second}{x_2(t)} = \begin{cases} x_2'(t) & \text{if } 0.25 \leq t \leq 0.5 \\ 0 & , \text{o.w} \end{cases}$ same for other digits

$x_{..}(t) = (u(t-0.25) - u(t-0.5))$

Third digit calculation: $x_3(t) = x(t)(u(t-0.5) - u(t-0.75))$

Fourth digit calculation: $x_4(t) = x(t)(u(t-0.75) - u(t-1))$

The first method calculates all peaks together but the second method computes peaks one by one. Dialed number can not be determined exactly if peaks not determined clearly.

Part 2

a) let $x(t) = \delta(t)$ we can find $h(t)$ easily because it is a LTI system.

$$h(t) = \delta(t) + \sum_{i=1}^{m} A_i \, \delta(t - t_i)$$

b)

$$h(t) \longrightarrow H(j\omega) = 1 + \sum_{i=1}^{m} A_i e^{-j\omega t_i}$$

c) $Y(j\omega) = X(j\omega) H(j\omega)$, convolution in time domain is multiplication in frequency domain and vice versa.

d) $X(j\omega)$ can be calculated by $\dfrac{Y(j\omega)}{H(j\omega)}$

for the echoed version of the sound, I realize there is an overlap between my voice because of the delay.
for the recovered non-echo version of the sound was sounds almost like the original sound.

Atahan İyiekici
21801985
EEE321-001
Lab report 3

EEE321 Lab report 3

PLOTS

Part 1.2

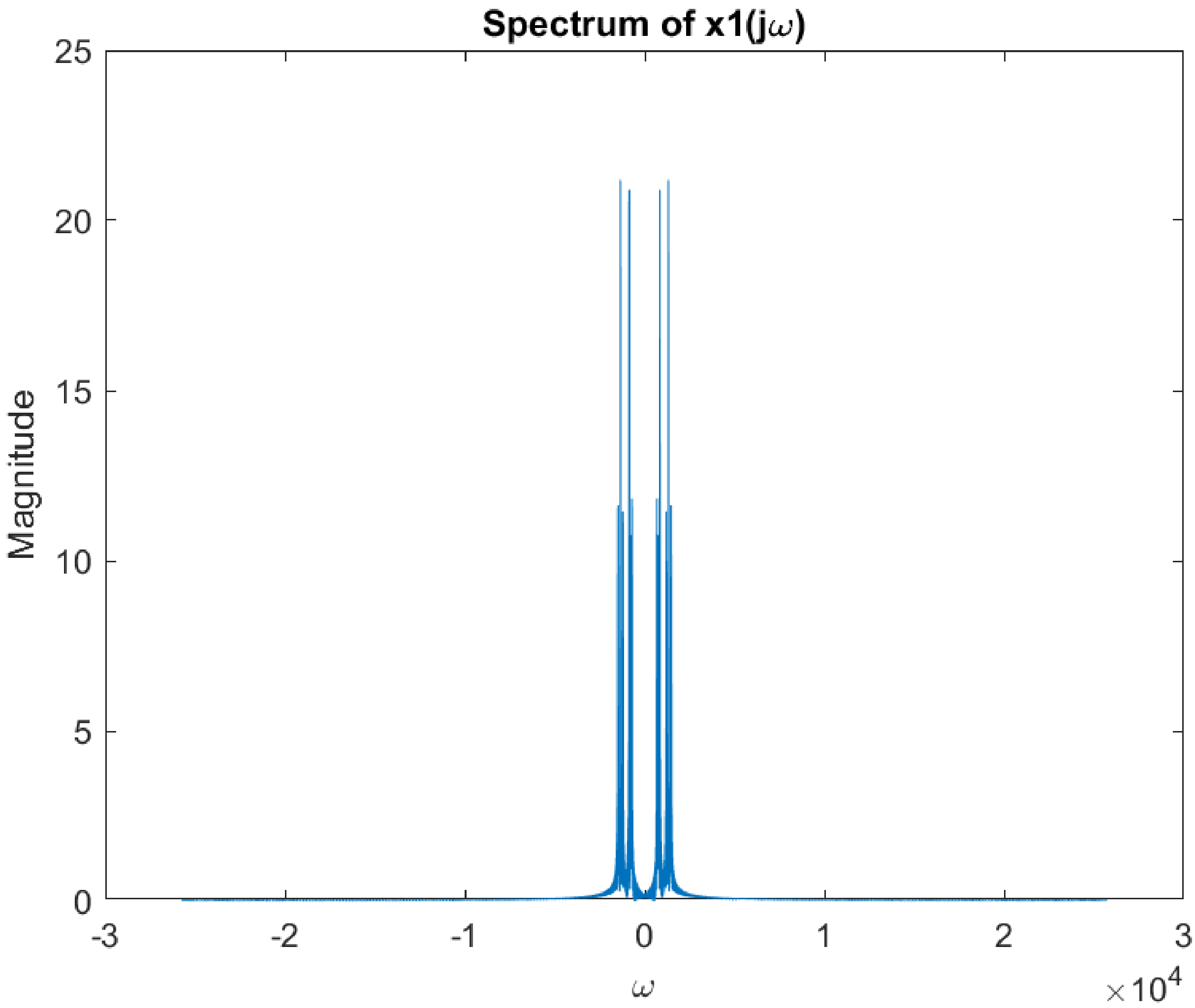

Figure 1: Plot of Spectrum of x1(jω)

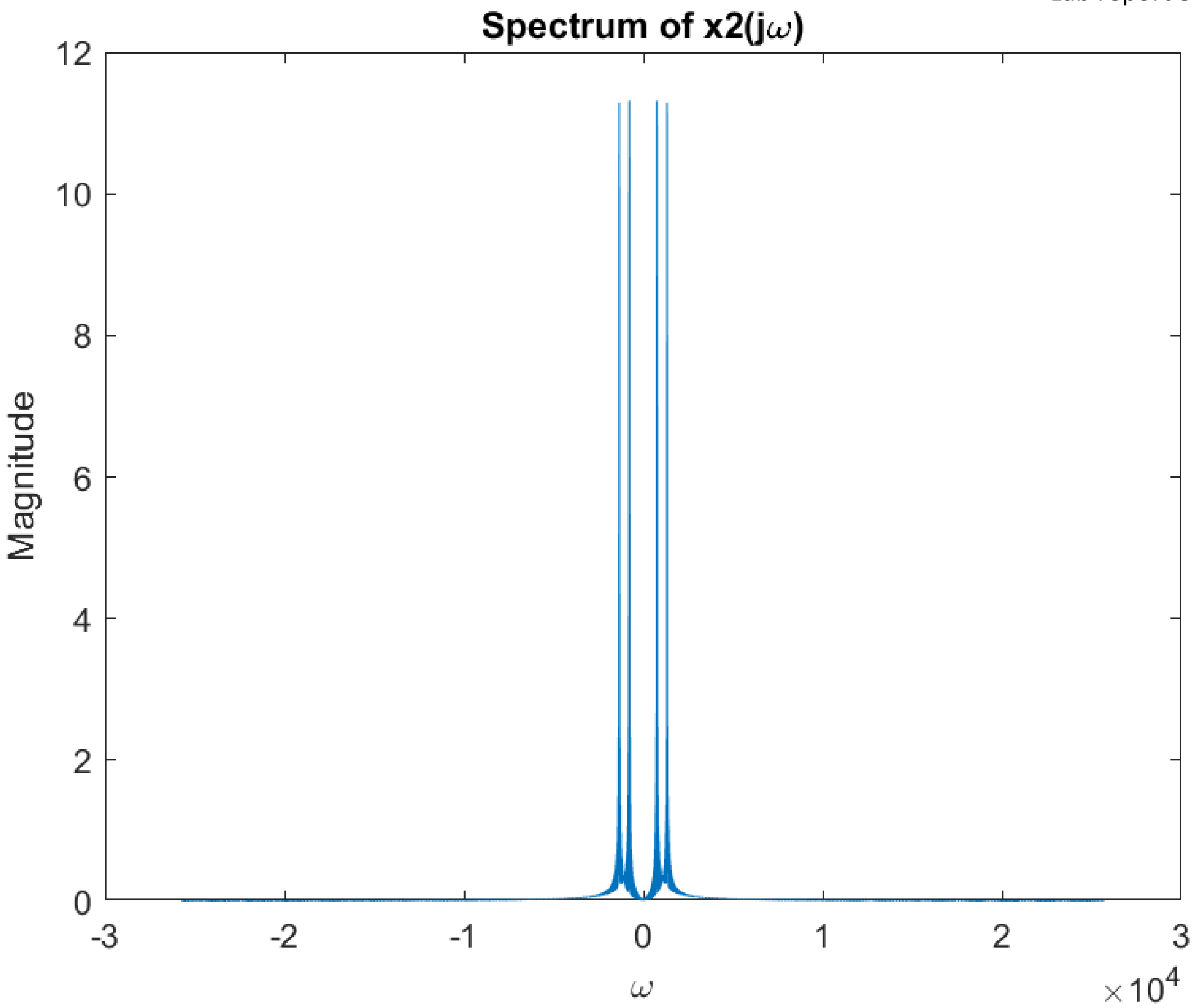

Figure 2: Plot of Spectrum X(jω) = x2(jω)

Atahan İyiekici
21801985
EEE321-001
Lab report 3

Part 2

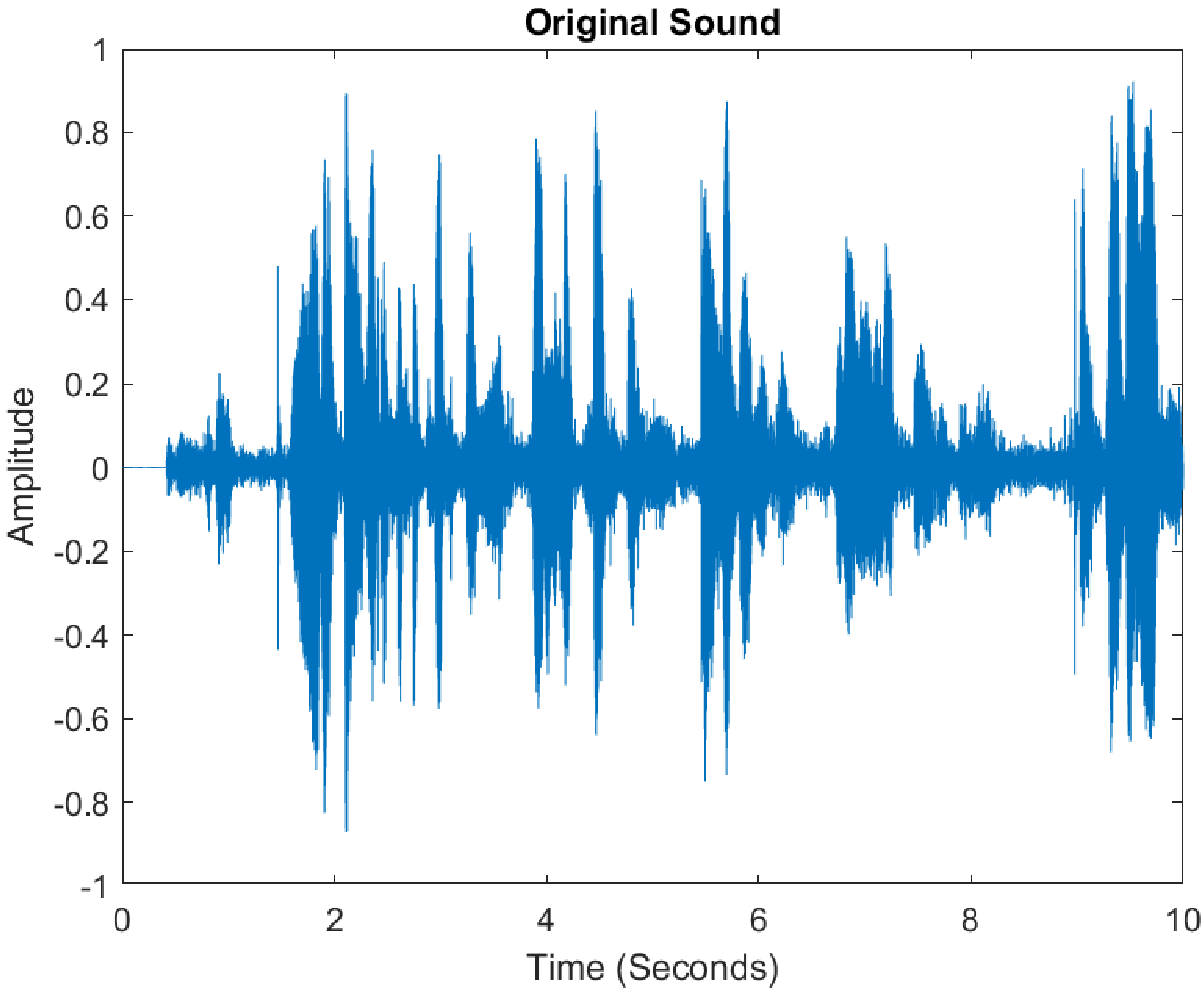

Figure 3: Plot of Amplitude of Original Sound

Atahan İyiekici
21801985
EEE321-001
Lab report 3

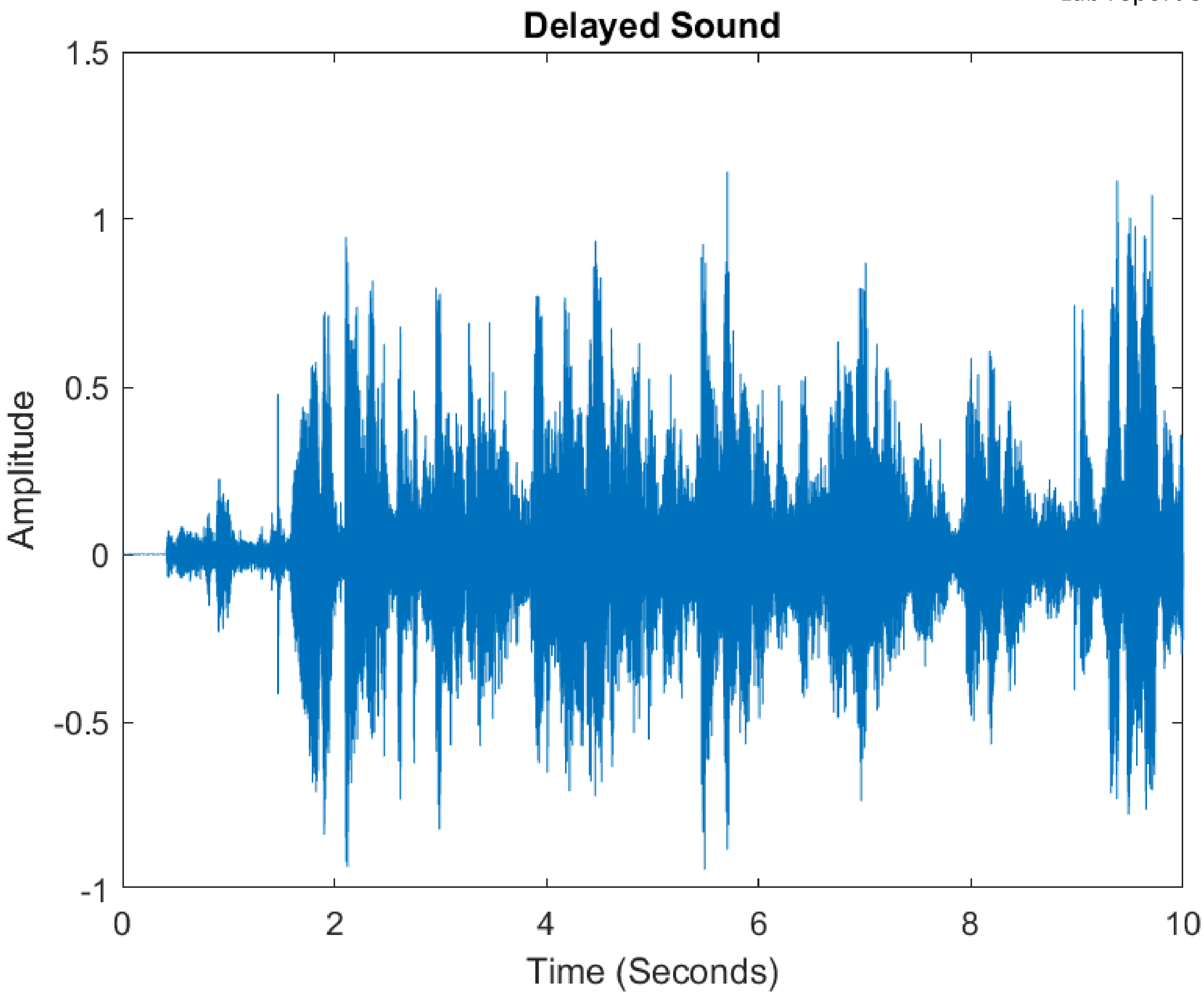

Figure 4: Plot of Amplitude of Delayed Sound

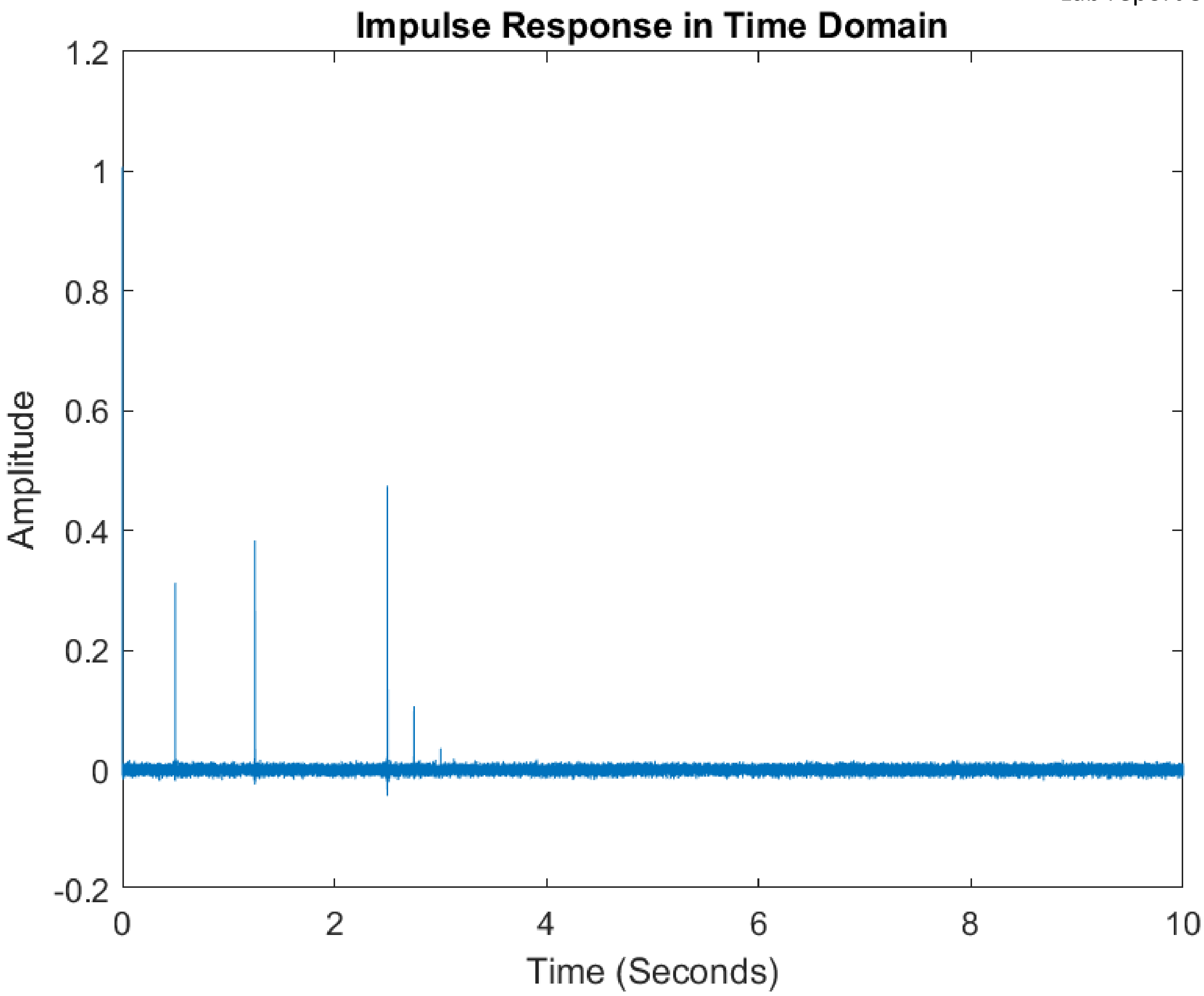

Figure 5: Plot of Impulse Response in Time Domain

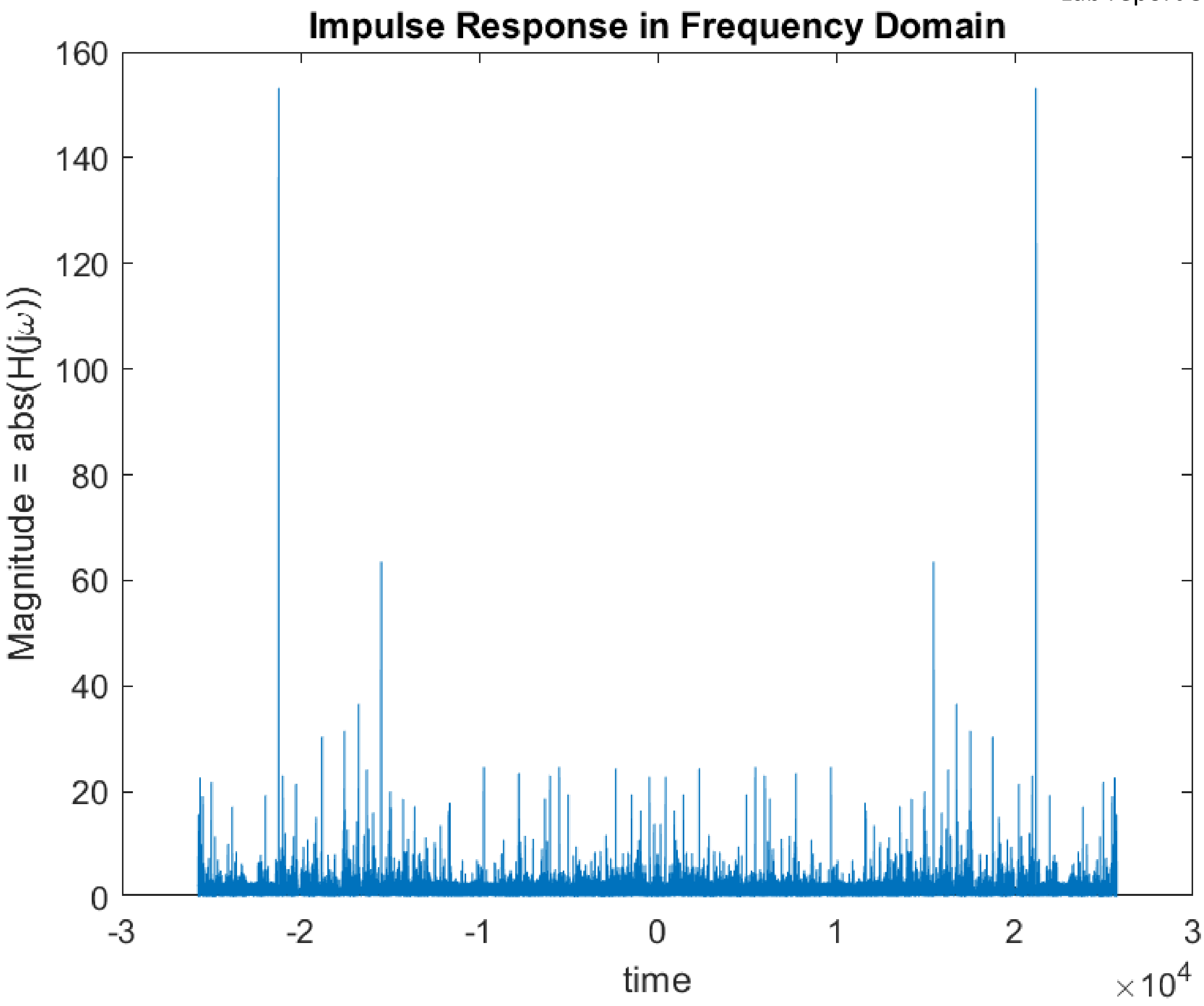

Figure 6: Plot of Impulse Response in Frequency Domain

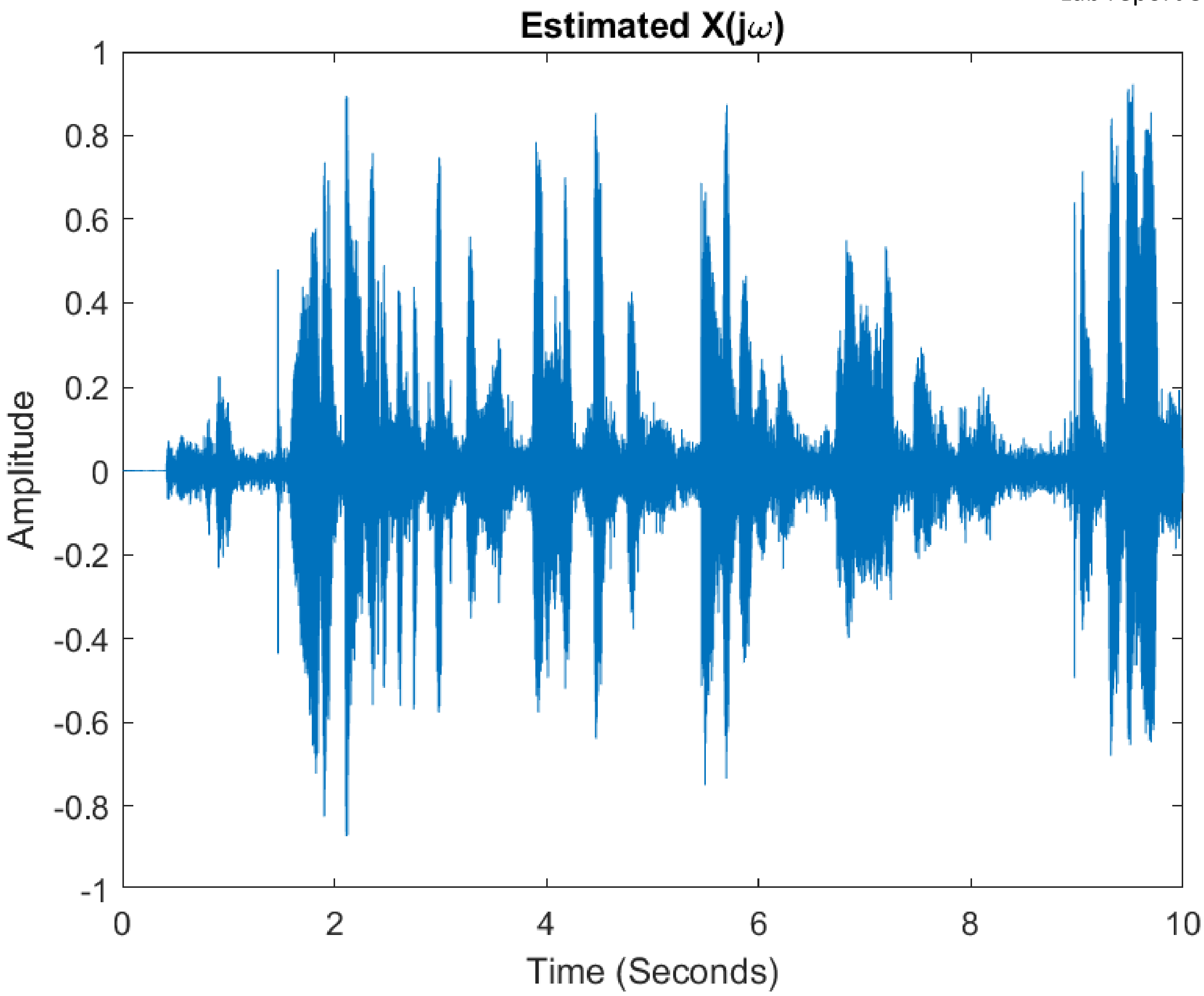

Figure 7: Plot of Estimated X(jω)

Atahan İyiekici
21801985
EEE321-001
Lab report 3

MATLAB CODES

```
%% PART 1.1 %%
Number = [0 5 3 8 0 6 3 9 4 9 7];
x = DTMFTRA(Number);
%% PART 1.2 %%
clear all;
Number = [5 8 9 1]; %21801985
x=DTMFTRA(Number);
x1= FT(x);
omega= linspace(-8192 * pi, 8192 * pi, 8193);
omega= omega(1 :8192);
figure;plot(omega,abs(x1));xlabel('\omega');ylabel('Magnitude');title('Spectrum of
x1(j\omega)');
y1= zeros(1,8192);
y1(1:2048)=1;
y1=y1 .* x;
x2 = FT(y1);
figure;plot(omega,abs(x2));xlabel('\omega');ylabel('Magnitude');title('Spectrum of
x2(j\omega)');

%% PART 2
record = audiorecorder(8192, 16, 1);
disp('Start Recording.');
recordblocking(record,10);
disp('End Recording.');
% play(record);
original = getaudiodata(record);
t = [0: 1/8192: 10 - 1 / 8192];
A = 0.35 * delayseq(original, 0.5, 8192);
A = A + 0.5 * (delayseq(original, 1.25, 8192));
A = A + 0.65 * (delayseq(original, 2.5, 8192));
A = A + 0.05 * (delayseq(original, 3, 8192));
A = A + 0.15 * (delayseq(original, 2.75, 8192));
original_Delayed = original + A;
% soundsc(delayed, 8192)
figure(3); plot(t, original); xlabel('Time (Seconds)');
ylabel('Amplitude');title('Original Sound');
figure(4); plot(t, original_Delayed); xlabel('Time (Seconds)');
ylabel('Amplitude');title('Delayed Sound');
Y = FT(original_Delayed);
X = FT(original);
omega = linspace(-8192 * pi, 8192 * pi, 81921);
omega = omega(1 : 81920);
H = Y ./ X;
h = IFT(H);
figure; plot(t, h); xlabel('Time (Seconds)'); ylabel('Amplitude');title('Impulse
Response in Time Domain');
figure; plot(omega, abs(H)); xlabel('time'); ylabel('Magnitude =
abs(H(j\omega))'); title('Impulse Response in Frequency Domain');
Xe = Y ./ H;
xe = IFT(Xe);
figure; plot(t, xe); xlabel('Time (Seconds)');
ylabel('Amplitude');title('Estimated X(j\omega)');
```

Atahan İyiekici
21801985
EEE321-001
Lab report 3

```
%% Functions
function [k]=DTMFTRA(Number)
    k = [];
    t = 0 : 1/8192 : 0.25 - 1 / 8192;
    for num = 1 : length(Number)
        if Number(num) == 1
            cos_sum = cos(2*pi*697*t) + cos(2*pi*1209*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 2
            cos_sum = cos(2*pi*697*t) + cos(2*pi*1336*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 3
            cos_sum = cos(2*pi*697*t) + cos(2*pi*1477*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 4
            cos_sum = cos(2*pi*770*t) + cos(2*pi*1209*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 5
            cos_sum = cos(2*pi*770*t) + cos(2*pi*1336*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 6
            cos_sum = cos(2*pi*770*t) + cos(2*pi*1477*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 7
            cos_sum = cos(2*pi*852*t) + cos(2*pi*1207*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 8
            cos_sum = cos(2*pi*852*t) + cos(2*pi*1336*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 9
            cos_sum = cos(2*pi*852*t) + cos(2*pi*1477*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 0
            cos_sum = cos(2*pi*941*t) + cos(2*pi*1336*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == '*'
            cos_sum = cos(2*pi*941*t) + cos(2*pi*1207*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == '#'
            cos_sum = cos(2*pi*941*t) + cos(2*pi*1477*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 'A'
            cos_sum = cos(2*pi*697*t) + cos(2*pi*1633*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 'B'
            cos_sum = cos(2*pi*770*t) + cos(2*pi*1633*t);
            k = cat(2, k, cos_sum);
        elseif Number(num) == 'C'
            cos_sum = cos(2*pi*852*t) + cos(2*pi*1633*t);
            k = cat(2, k, cos_sum);
        else %D
            cos_sum = cos(2*pi*941*t) + cos(2*pi*1633*t);
            k = cat(2, k, cos_sum);
        end
    end
    x = k;
    soundsc(x,8192);
end
```

Atahan İyiekici
21801985
EEE321-001
Lab report 3

```
function output=FT(input)
M=size(input,2);
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

function output=IFT(input)
M=size(input,2);
t=exp(-j*pi*(M-1)/M*[0:1:M-1]);
output=real(exp(j*pi*(M-1)^2/(2*M))*t.*(M)^0.5.*ifft(input.*t));
end
```

References

https://www.mathworks.com/help/matlab/import_export/record-and-play-audio.html

https://www.mathworks.com/help/phased/ref/delayseq.html