EEE 321 Signals and Systems

Lab3

Sait Sarper Özaslan

22002861 Section 02

Part1.1 DTMF Transmitter

```
function x = DTMFTRA(digits)

        Fs = 8192;

        a1= [1477 1209 1336]; %freq rate for column

        a2= [697 770 852]; %freq rate for row

        t= 0.25;

        digits = [1 3 6 4 6 8 6];%not in original code

        for i = 1:length(digits)

                    k=(t*(i-1):1/Fs:(t*i-1/Fs)); %length of 0.25 sec in sample rate,
                    one is excluded to prevent repetition

              if digits(i) == 0

                    y= cos(2*pi*941*k)+cos(2*pi*1336*k);

              else

                    y=
                    cos(2*pi*a1(mod(digits(i),3)+1)*k)+cos(2*pi*a2(idivide(int32(digi
                    ts(i)),3.3)+1)*k);

              end

              x(1,(2048*(i-1)+1):(2048*i)) =y ; %the data is allocated at x at
              different places for each 0.25sec

        end

        soundsc(x,8192) %the final sound that sounds like dials of the phone, not in
        original code

end
```

The sound heard is same as the number dials on telephones.

Part 1.2 DTMF Receiver

My ID = 22002861 and as I am in section 2, the 5 digits I used in this lab are Digits =[1 6 8 2 2]. The corresponding figures are in order of X(w) vs w, and then each digit is in order of the array.

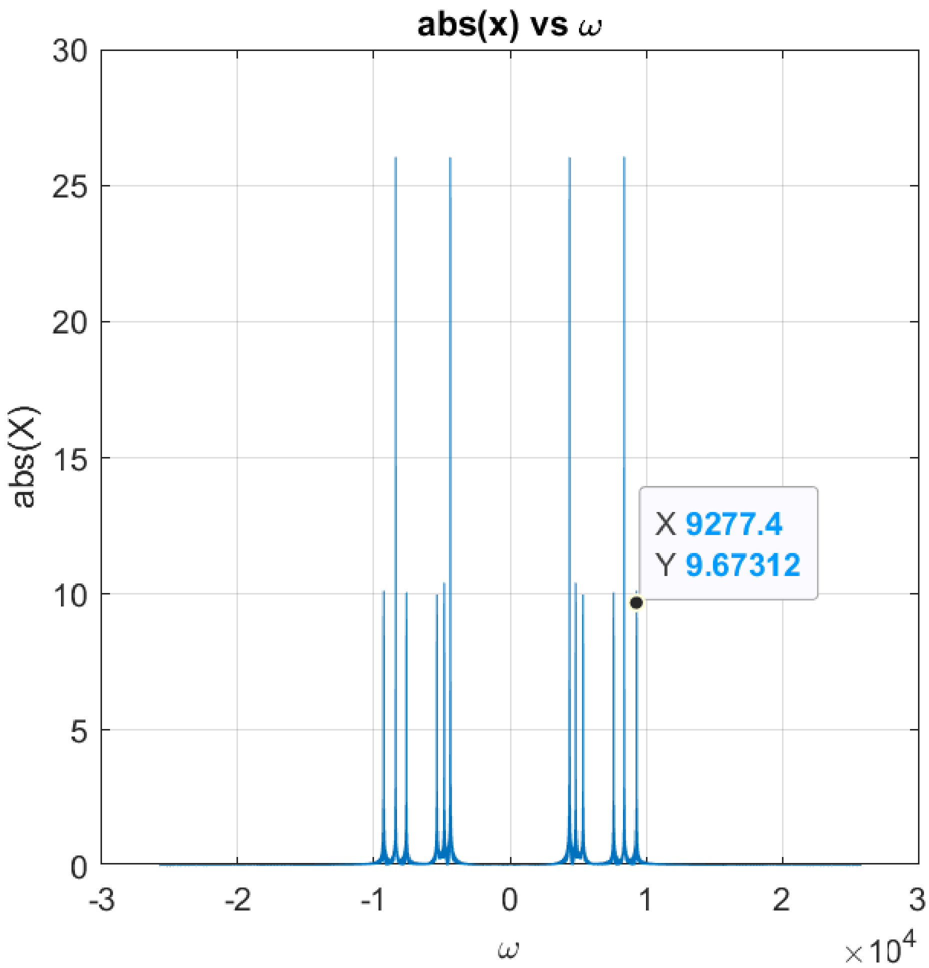


Figure 1: abs(x) vs w

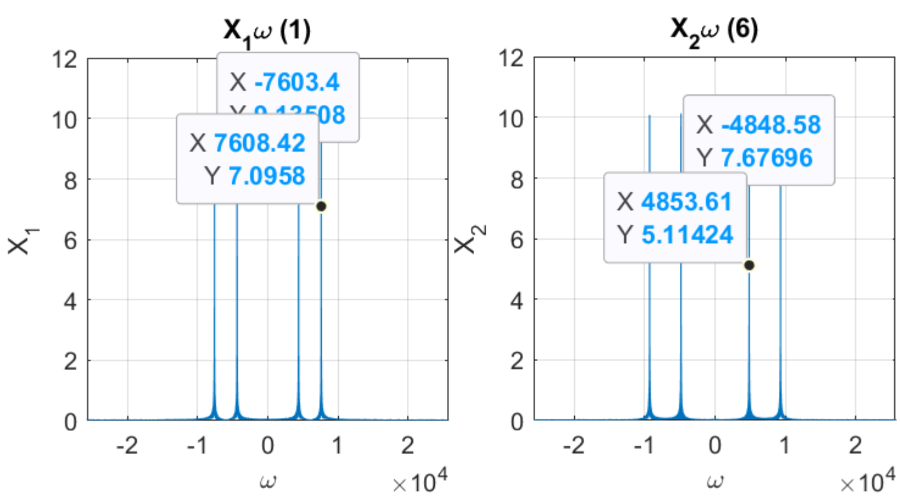


Figure 2: X1 and X2, Digits 1 and 6

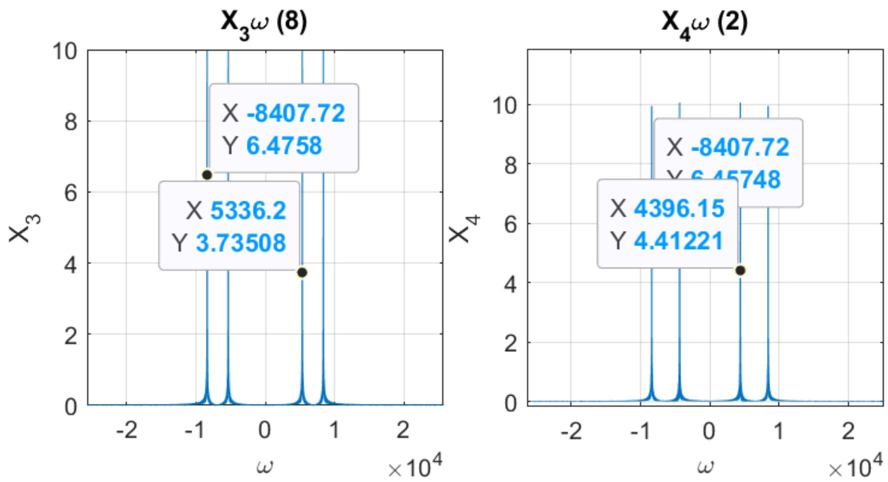


Figure 3: X3 and X4, Digits 3 and 4

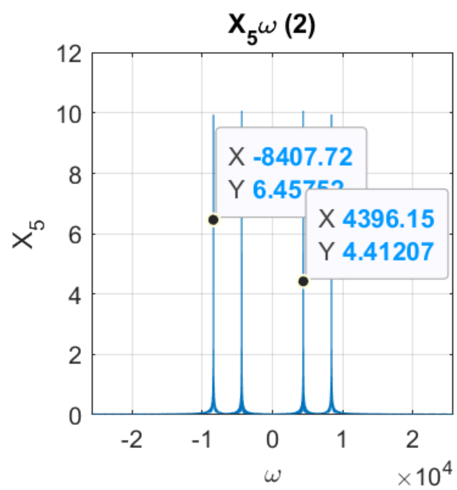


Figure 4: X5, Digit 2

To comment on these figures there are several signals that are mixed up together on figure 1 and in some cases the ones with the same frequencies add up making it harder to observe which digit is pressed. However, we can make sure that digit 6 is pressed as it is the only digit who uses the value 1477Hz in its time domain. This can be seen as 9280 radian/s is one of the values in figure 1. The reason why those values were seen can be proved from the question b solved in part 1.2, in which cosine's Fourier transform gave the result of two symmetrically shifted impulses.

Looking and understanding figure 2 is much easier. For instance, the number 1 uses the frequency "1209Hz" as one of its cosine's frequency values. Hence the expected value in frequency domain is 7596 radian/s. We can confirm a similar value when looked at figure 2 for digit "1". There is also negative version of this values which is due to cosine being an even function. The same case can be said about other digits as well.

Here are the frequency values from each figure and their digits:

- 7600 rad/s (around 1209Hz) and 4380 rad/s (around 697Hz) which corresponds to digit 1
- 4851 rad/s (around 770Hz) and 9281 rad/s (around 1477 Hz) which corresponds to digit 6
- 5336 rad/s (around 852Hz) and 8407 rad/s (around 1336 Hz) which corresponds to digit 8
- 4396rad/s (around 697Hz) and 8407 rad/s (around 1336 Hz) which corresponds to digit 2
- Same for last digit as it is also 2

The written questions from part 1:

a) $x(t) = e^{j2\pi f_0 t}$, $2\pi f_0 = w_0$

Using duality:

$$F\{\delta(t)\} = 1$$

$$= F\{1\} = 2\pi\,\delta(w)$$

Then add the modulation property

$$F\{1 \cdot e^{jw_0 t}\} = 2\pi\,\delta(w-w_0)$$

b) $x(t) = \cos(w_0 t)$

we can find coefficient using $a_k \frac{2\pi}{w_0} = \int_{T_0} \cos(w_0 t)\, e^{-jw_0 t}$

Or we can use:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\pi(\delta(w-w_0)) + \pi\delta(w+w_0)\right) e^{jwt}$$

$$= \frac{1}{2} \quad e^{jw_0 t} + e^{-jw_0 t} \quad = \cos(w_0 t)$$

Then $X(w) = \pi\,\delta(w-w_0) + \pi\delta(w+w_0)$

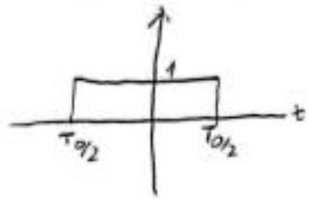
c) $\sin(w_0 t)$ has the same method of solution as b

but considering how there is "$\frac{-}{2j}$" and $-$ in euler relation of $\sin(w_0 t$

$$X(w) = \frac{\pi}{j}\,\delta(w-w_0) - \frac{\pi}{j}\delta(w+w_0)$$

Figure 5: Solutions from a to c

d) $x(t) = rect(t/T_0)$



$$X(\omega) = \int_{-T_0/2}^{T_0/2} 1\, e^{-j\omega t}\, dt$$

$$\left.\frac{e^{-j\omega t}}{+j\omega}\right]_{+T_0/2}^{-T_0/2} = \frac{e^{+j\omega T_0/2} - e^{-j\omega T_0/2}}{j\omega}$$

$$X(\omega) = \frac{2\sin\frac{T_0}{2}\omega}{\omega} = T_0\,\frac{\sin\frac{T_0\omega}{2}}{\frac{T_0\omega}{2}}$$

$$\boxed{= T_0\ sinc\left(\frac{T_0\omega}{2}\right)}$$

e) $x(t) = e^{+j2\pi f_0 t}\, rect\left(\frac{t}{T_0}\right)$, $2\pi f_0 = w_0$

Using the shift property on the result of d) where we shift to right by $2\pi f_0$

$$X(\omega) = T_0\ sinc\left(\frac{(w-w_0)T_0}{2}\right)$$

f) $x(t) = \cos(2\pi f_0 t)\, rect\left(\frac{t}{T_0}\right)$

Writing $\cos(2\pi f_0 t)$ in complex exponential form we get;

$$\cos 2\pi f_0 t = \frac{e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}}{2}$$

We can think of this as conjunction of time shift, linearity properties on $X(\omega)$. Therefore $X(\omega)$ is;

$$X(\omega) = \frac{T_0}{2}\ sinc\left(\frac{(w-w_0)T_0}{2}\right) + T_0\ sinc\left(\frac{(w+w_0)T_0}{2}\right)$$

Figure 6: Solutions d to f

g) Let $x(t) = rect\left(\frac{t-t_0}{T_0}\right)$

we have time shift $x(t)$ to right by t_0 using time shift property and result in d) we get

$$e^{-j\omega t_0} \cdot T_0 \, sinc\left(\frac{\omega T_0}{2}\right)$$

h) Let $x(t) = e^{j2\pi F_0 t} \, rect\left(\frac{t-t_0}{T_0}\right)$

This is a combination of result of d, e, g using time shift on $x(t)$ and $X(\omega)$

$$X(\omega) = e^{-j(\omega-\omega_0)t_0} \frac{T_0 \, sinc\left(\frac{T_0(\omega-\omega_0)}{2}\right)}{\omega-\omega_0}$$

i) $x(t) = \cos(2\pi f_0 t) \, rect\left(\frac{t-t_0}{T_0}\right)$, $2\pi f_0 = \omega_0$

We just do time shift on $x(t)$ on top of the result we found in F)

$$X(\omega) = \frac{1}{2}\left(e^{-j(\omega-\omega_0)t_0}\left(\frac{sinc\left(\frac{T_0(\omega-\omega_0)}{2}\right)}{\omega-\omega_0}\right) + e^{-j(\omega+\omega_0)t_0}\left(\frac{sinc\frac{T_0(\omega+\omega_0)}{2}}{\omega+\omega_0}\right)\right)$$

Figure 7: Solutions part g to i

Part 2

In this part, I recorded my voice straight through the function called "audiorecorder". From this function I recorded a 12 second value with the sampling rate of 8192Hz.

The words I used for this part were "Hello I am Sarper Özaslan My ID is 2 2 0 0 2 8 6 1 , This is signals and systems course That's it"

The written questions for this part were:

PART2

$$y(t) = x(t) + \sum_{i=1}^{M} A_i \delta(t-t_i)$$

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau$$

$$Y(\omega) = X(\omega) \cdot H(\omega) \Rightarrow \cancel{X(\omega)} + \sum_{i=1}^{M} A_i e^{-j\omega t_i} \cdot \cancel{X(\omega)} = \cancel{X(\omega)}\, H(\omega)$$

The relation is expressed at part c

$$1 + \sum_{i=1}^{M} A_i e^{-j\omega t_i} = H(\omega) \Rightarrow \text{Frequency response}$$

To find h(t) we use synthesis

$$h(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} H(\omega)\, e^{j\omega t}\, d\omega$$

$$h(t) = \underbrace{\frac{1}{2\pi} \int_{-\infty}^{\infty} 1 e^{j\omega t}\, d\omega}_{\frac{1}{2\pi} 2\pi \delta(\omega)} + \underbrace{\frac{1}{2\pi} \int_{-\infty}^{\infty} \sum_{i=1}^{M} A_i e^{-j\omega t_i} \cdot e^{+j\omega t}\, d\omega}_{\frac{1}{2\pi} 2\pi}$$

a) $\boxed{h(t) = \delta(\omega) + \sum_{i=1}^{M} A_i\, \delta(\omega - \omega_i)}$

Essentially an impulse train with weight A_i's and goes from $\omega = \omega_0$ to $\omega = \omega_M$

b) $H(\omega) = 1 + \sum_{i=1}^{M} A_i e^{-j\omega t_i}$

Figure 8: a and b

c) This is called the convolution property

$$y(t) = x(t) * h(t) \Longleftrightarrow Y(\omega) = X(\omega) \cdot H(\omega)$$

$$y(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau$$

$$\Downarrow F\{\ \}$$

$$Y(\omega) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau \right] e^{-j\omega t}\, dt$$

Interchanging the order of integration

$$\int_{-\infty}^{\infty} x(\tau) \underbrace{\left[\int_{-\infty}^{\infty} \underbrace{h(t-\tau)}_{\text{there is time shift}} e^{-j\omega t}\, dt \right]}_{e^{-j\omega\tau} H(\omega)} d\tau$$

$$Y(\omega) = \int_{-\infty}^{\infty} H(\omega)\, x(\tau)\, e^{-j\omega\tau}\, d\tau$$

$$Y(\omega) = H(\omega) \cdot X(\omega)$$

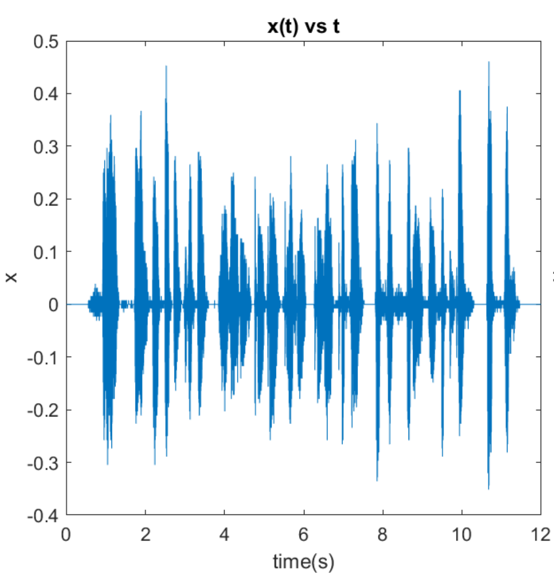
d) Since $y(t) = h(t) * x(t)$

Then $Y(\omega) = H(\omega) \cdot X(\omega)$

$$X(\omega) = \frac{Y(\omega)}{H(\omega)}$$

Figure 9: c and d Solutions

The plots of x(t) vs t, and y(t) vs t are as follows:



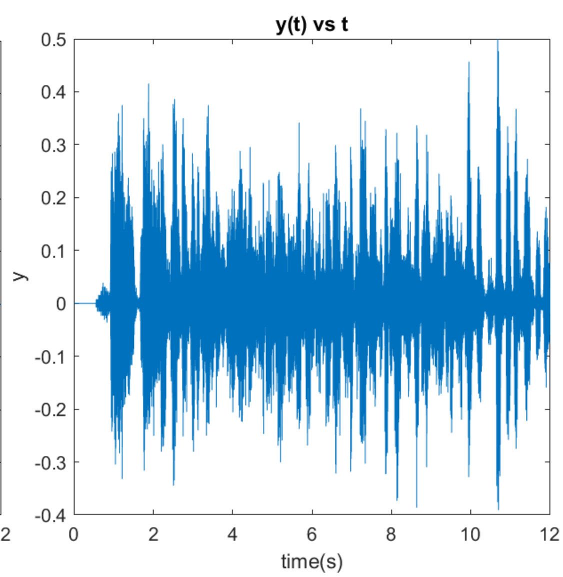


Figure 10: Plots of x(t) vs t and y(t) vs t

When soundsc was used in y(t) the audio was almost inaudible

The Plots of h(t) vs t, and |H(w)| vs w can be seen below:

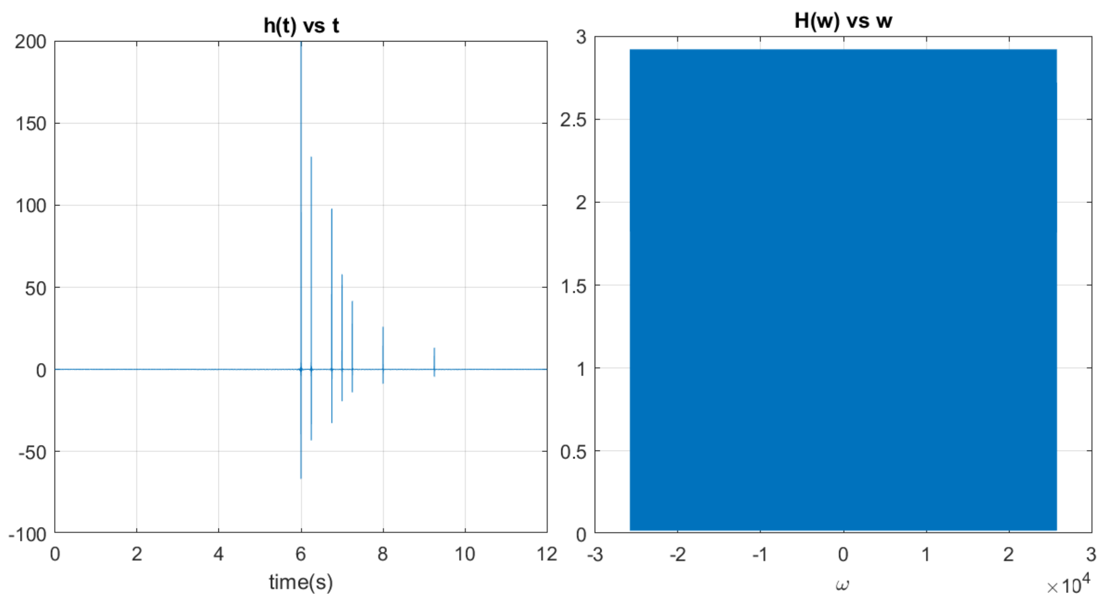


Figure 11: h(t) vs t, H(w) vs w Plots

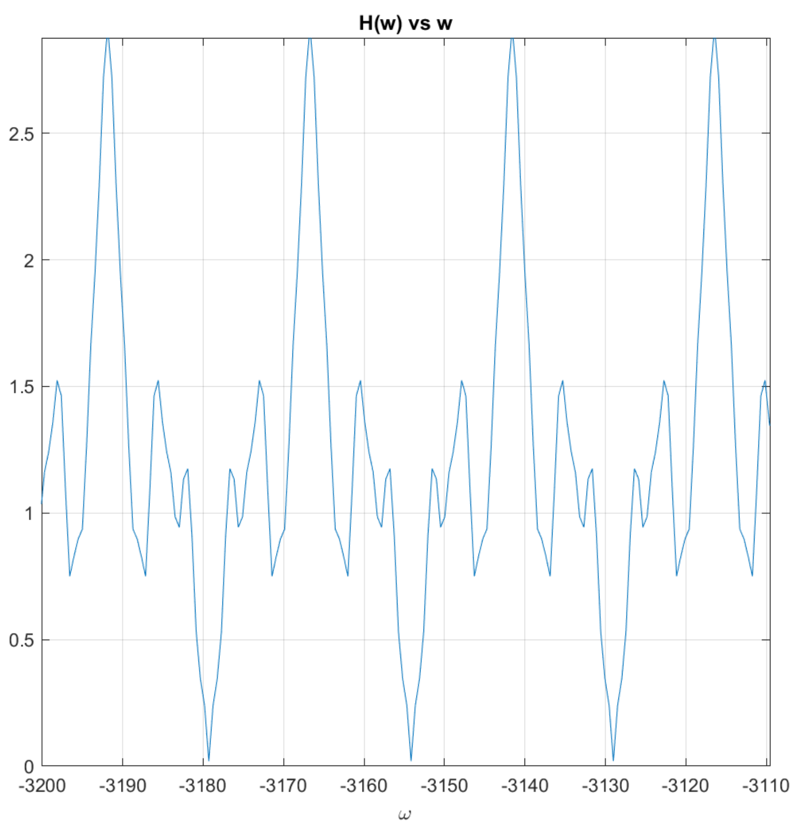


Figure 12: H(w) vs w Zoomed in Version

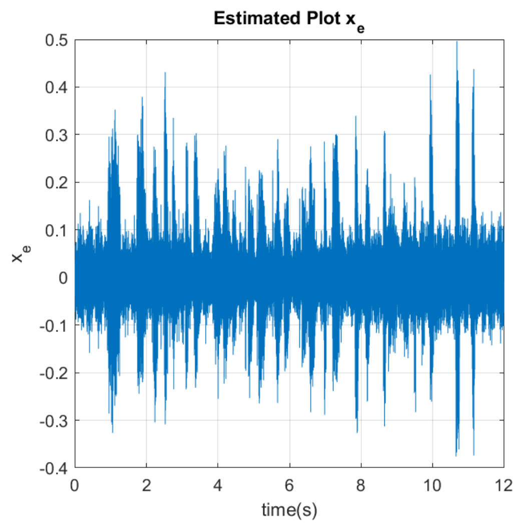


Figure 13: Estimated plot xe versus time

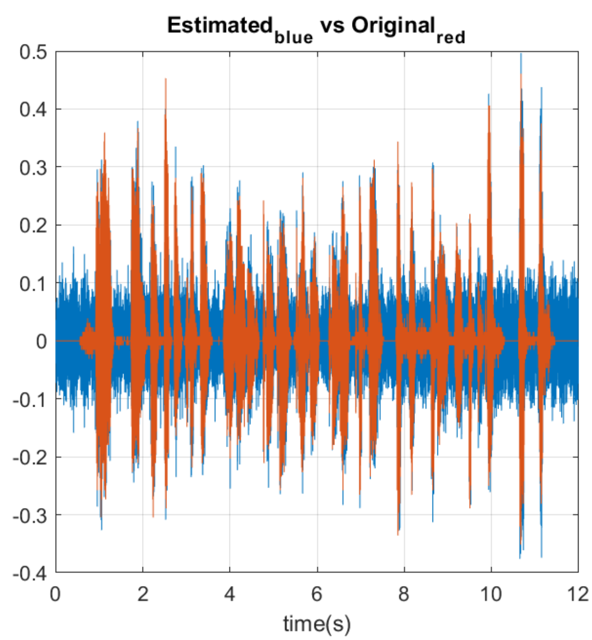


Figure 14: Estimated versus Original plot vs time

When y(t) was written using the function defined in the report as,

$$y(t) = x(t) + \sum_{i=1}^{M} A_i x(t - t_i) \quad ,$$

Figure 15: y(t)'s relation with x(t)

In which the delayed signals were cropped. The soundsc function had a lot of echos. This can also be confirmed from the comporison of y(t) vs t and x(t) vs t plots in figure 10 in which y(t) has a lot of distortions compared to our original sample x(t). This is due to y(t) being a combination of x(t) and delayed x(t)s with weights Ai.

However, we can still extract the original signal through the usage of impulse function, h(t), and convolution. As we had found what H(w) was in the written section of part 2. All I needed to do was to write it appropriately to the MATLAB and divide y(t)'s frequency domain version "Y(w)" with "H(w)". From the relation expressed in "part 2 d)" we now have Xe(w) which is on frequency domain. From here all we need to do is to use the given IFT function and find the approximated function xe(t).

The soundsc() command in this case still had a noise but as one can see in figure 14, it is much closer to original sample x(t). Overall, the voice quality highly improved from something inaudible to something easily audible. Moreover, we may need more samples to get more accurate data as well.

Whole code can be found in Appendix

Appendix

```
%part 1.1 code
function x = DTMFTRA(digits)
    Fs = 8192;
    a1= [1477 1209 1336]; %freq rate for column
    a2= [697 770 852]; %freq rate for row
    t= 0.25;
        for i = 1:length(digits)
            k=(t*(i-1):1/Fs:(t*i-1/Fs)); %length of 0.25 sec in sample rate,
            one is excluded to prevent repetition
        if digits(i) == 0
            y= cos(2*pi*941*k)+cos(2*pi*1336*k);
        else
            y=
            cos(2*pi*a1(mod(digits(i),3)+1)*k)+cos(2*pi*a2(idivide(int32(digi
            ts(i)),3.3)+1)*k);
        end
    x(1,(2048*(i-1)+1):(2048*i)) =y ; %the data is allocated at x at different
    places for each 0.25sec
    end
End
```

```
%Part 1.2
%ID 22002861 section 2
Fs=8192;
digits = [1 6 8 2 2];
x = DTMFTRA(digits);
soundsc(x,8192); %Original dial sound
X=FT(x);
omega = linspace(-Fs*pi,Fs*pi,10240); %points for frequency domain
omega = omega(1:10240);
```

```
figure (1),plot (omega,abs(X)),title("abs(x) vs \omega"), xlabel('\omega'),
ylabel('abs(X)'), axis square, grid on

t = 0.25;

figure (18)

for g= 1:5 %different square waves are created in the outer loop

    for i=1:5 %data is loaded to rec in the inner loop

        m=(t*(i-1):1/Fs:(t*i-1/Fs));

        if i == g %1,2,3,4,5 corresponds to different squarevawe

        rec=1*x(1,(2048*(i-1)+1):(2048*i));

        else

        rec=0*x(1,(2048*(i-1)+1):(2048*i));

        end

        x1(1,(2048*(i-1)+1):(2048*i))=rec;

    end

    o=FT(x1); %plots are created here after every squarewave formation

    str1= ['X_' num2str(g) '\omega (' num2str(digits(g)) ')'];

    str2 = ['X_' num2str(g)];

    figure(18), subplot(1,5,g), plot(omega,abs(o)),title(str1),
    xlabel('\omega'),ylabel(str2), axis square, grid on

End


%Part 2

Fs = 8192;

r = audiorecorder(Fs, 8, 1);

recordblocking(r,12); % speak into microphone...

x = getaudiodata(r);

x = x.';%transposed to be able to form into an usable array

soundsc(x,Fs)

time=0:1/Fs:12-1/Fs;

M=6;

Ai = [0.65 0.50 0.30 0.22 0.15 0.1];

delay = [0.25 0.75 1 1.25 2 3.25];
```

```
dummy= zeros(1,Fs*12); %in here every shift is seperated and formed into 12 second long audios through the multiplication with sample rate Fs=81292

for i = 1:M

    for ti = time

        if ti >= delay(i)

        dummy(ti*Fs+1) = dummy(ti*Fs+1)+ Ai(i)*x((ti-delay(i))*Fs+1);

        end

    end

end

y = x + dummy;

figure(5), plot(time, y), title('y(t) vs t'), xlabel('time(s)'), ylabel('y'), axis square

figure(6), plot(time, x), title('x(t) vs t'), xlabel('time(s)'), ylabel('x'), axis square

% soundsc(y, 8192)

Y=FT(y);

omega=linspace(-8192*pi,8192*pi,98305);

omega=omega(1:98304);

H = zeros(size(omega)); %the result found in part b is used to form frequency response H(W)

for ii = 1:M

    H = H + Ai(ii)*exp(-1i*omega*delay(ii));

end

H = H+1; % addition of the impulse at omega=0

h=IFT(H);

figure(7), plot(time,h), title('h(t) vs t'), xlabel('time(s)'), axis square, grid on

figure(8), plot(omega,abs(H)), title('H(w) vs w'), xlabel('\omega'), axis square, grid on

%calculation of xe

X=Y./H;

xe=IFT(X);

figure(9), plot(time,xe), title('Estimated Plot x_e'), xlabel('time(s)'), ylabel("x_e"), axis square, grid on

% Estimated vs. original
```

```
figure(10), plot(time, xe), hold all, plot(time, x), title('Estimated_b_l_u_e vs
Original_r_e_d'), xlabel('time(s)'),axis square, grid on

soundsc(xe,8192);
```