Eray Gündoğdu 22101962

Signals and Systems Lab Report 3

Part 1)Transmitter

In this part a DTMF transmittor is designed by using the freguency table given below. Also the code for DTMF transmittr can be investigated.

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Table1.1 DTMF frequencies

```
Number1=[0 5 3 6 5 6 8 9 8 3 0];
x1=DTMFTRA(Number1);
soundsc(x1,8192)
```

```
function [x0]=DTMFTRA(Number)


t=[0:1/8192:0.25];
x=0;
y=0;
    for i=1:length(Number)
        if Number(i)==1
            x= cos(2*pi*697*t)+ cos(2*pi*1209*t);
        elseif Number(i)==2
            x= cos(2*pi*697*t)+ cos(2*pi*1336*t);
        elseif Number(i)==3
            x= cos(2*pi*697*t)+ cos(2*pi*1477*t);
        elseif Number(i)==4
            x= cos(2*pi*770*t)+ cos(2*pi*1209*t);
        elseif Number(i)==5
            x= cos(2*pi*770*t)+ cos(2*pi*1336*t);
        elseif Number(i)==6
            x= cos(2*pi*770*t)+ cos(2*pi*1477*t);
       elseif Number(i)==7
            x= cos(2*pi*852*t)+ cos(2*pi*1209*t);
        elseif Number(i)==8
            x= cos(2*pi*852*t)+ cos(2*pi*1336*t);
        elseif Number(i)==9
            x= cos(2*pi*852*t)+ cos(2*pi*1477*t);
        elseif Number(i)==0
            x= cos(2*pi*941*t)+ cos(2*pi*1336*t);
        end
    t=t+length(t);
    if i==1
        x0=x;
```

```
    else
        x0 = [x0,x]; %combining outputs of the function
    end
    end
end
```

The sound of the signal is similar to the sound when the same phone number is typed to the cell phone. In this transmittor 11 DTMF tones with 0.25 seconds apart from each other are transmitted.

Reciever

(a) $x(t) = e^{j2\pi f_0 t}$, $\boxed{X(\omega) = 2\pi\,\delta(\omega - 2\pi f_0)}$

(b) $x(t) = \cos(2\pi f_0 t) = \dfrac{e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}}{2}$, $e^{-j2\pi f_0 t} \xrightarrow{\text{Fourier}} 2\pi\delta(-\omega - 2\pi f_0)$

$= 2\pi\delta(\omega + 2\pi f_0)$, $X(\omega) = \frac{1}{2}\cdot 2\pi\cdot(\delta(\omega - 2\pi f_0)) + \frac{1}{2}\cdot 2\pi\cdot(\delta(\omega + 2\pi f_0))$

$$\boxed{X(\omega) = \pi\left(\delta(\omega - 2\pi f_0) + \delta(\omega + 2\pi f_0)\right)}$$

(c) $x(t) = \sin(2\pi f_0 t) = \dfrac{e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}}{2j} \xrightarrow{F.T}$

$$\overbrace{\frac{2\pi}{2j}(\delta(\omega - 2\pi f_0))}^{\text{from a}} - \overbrace{\frac{2\pi}{2j}(\delta(\omega + 2\pi f_0))}^{\text{from b}} = \boxed{\frac{\pi}{j}\left(\delta(\omega - 2\pi f_0) - \delta(\omega + 2\pi f_0)\right) = X(\omega)}$$

(d) $x(t) = \mathrm{rect}\left(\frac{t}{T_0}\right) = \begin{cases} 1, & |t| < |T_0/2| \\ 0, & \text{otherwise} \end{cases}$

$$X(\omega) = \int_{-\infty}^{\infty} x(t)\cdot e^{-j\omega t}\,dt = \int_{-T_0/2}^{T_0/2} e^{-j\omega t}\,dt = -\frac{1}{j\omega}\left[e^{-j\omega\frac{T_0}{2}} - e^{j\omega\frac{T_0}{2}}\right]$$

$$\boxed{X(\omega) = \frac{2}{\omega}\left(\sin\left(\frac{\omega T_0}{2}\right)\right) \text{ or } T_0\,\mathrm{sinc}\left(\frac{\omega T_0}{2\pi}\right)}$$

(e) $e^{j2\pi f_0 t} x_1(t) \xrightarrow{FT} X_1(\omega - 2\pi f_0)$, $x_1(t)$ is $\mathrm{rect}\left(\frac{t}{T_0}\right)$

from d $X_1(\omega) = T_0\,\mathrm{sinc}\left(\frac{\omega T_0}{2\pi}\right)$, substituting $\omega - 2\pi f_0$ to ω

$$X(\omega) = X_1(\omega - 2\pi f_0) = \boxed{T_0\,\mathrm{sinc}\left(\frac{\omega T_0}{2\pi} - 1\right)}$$

(f) $x_1(t)\cdot x_2(t) \xrightarrow{FT} \frac{1}{2\pi}\cdot X_1(\omega)\cdot X_2(\omega)$

$x_1(t) = \cos(2\pi f_0 t)$, $x_2(t) = \mathrm{rect}(t/T_0)$, $X_1(\omega) = \pi(\delta(\omega - 2\pi f_0) + \delta(\omega + 2\pi f_0))$

$X_2(\omega) = T_0\cdot\mathrm{sinc}(\omega\cdot T_0/2\pi)$, using property,

$X(\omega) = \frac{1}{2\pi}\cdot\left(\pi(\delta(\omega - 2\pi f_0) + \delta(\omega + 2\pi f_0)) * T_0\,\mathrm{sinc}\left(\frac{\omega T_0}{2\pi}\right)\right)$, by sifting property

$$\boxed{X(\omega) = \frac{T_0}{2}\left(\mathrm{sinc}\left(\frac{\omega_0 T_0}{2\pi} - 1\right) + \mathrm{sinc}\left(\frac{\omega T_0}{2\pi} + 1\right)\right)}$$

(g) time shifting property → $x_1(t-t_0) \xrightarrow{FT} e^{-j\omega t_0}\cdot X_1(\omega)$

$x(t) = rect\left(\frac{t-t_0}{T_0}\right)$, $x_1(t) = rect\left(\frac{t}{T_0}\right)$

$$X(\omega) = e^{-j\omega t_0}\cdot \underbrace{X_1(\omega)}_{\text{from part d}} = e^{-j\omega t_0}\cdot T_0 \, sinc\left(\frac{\omega T_0}{2\pi}\right)$$

(h) $e^{j2\pi f_0 t}\cdot \underbrace{rect\left(\frac{t-t_0}{T_0}\right)}_{X_1 \text{ is known from f}}$, by sifting property due to multiplication with $e^{j2\pi f_0 t}$, $X(\omega) = X_1(\omega - 2\pi f_0)$

$$= e^{-j\omega t_0}\cdot e^{j2\pi f_0 t_0}\cdot T_0 \cdot sinc\left(\frac{\omega T_0 - 2\pi f_0 T_0}{2\pi}\right),$$

$$\boxed{X(\omega) = e^{-j\omega t_0}\cdot T_0\cdot e^{j2\pi f_0 t_0}\cdot sinc\left(\frac{\omega T_0}{2\pi} - 1\right)}$$, using $T_0 \cdot f_0 = 1$

(i) $x_1(t)\cdot x_2(t) = \frac{1}{2\pi}\left(X_1(\omega) * X_2(\omega)\right)$

$x_1(t) = \cos(2\pi f_0 t) \xrightarrow[\text{from part b}]{FT} X_1(\omega) = \pi\left(\delta(\omega - 2\pi f_0) + \delta(\omega + 2\pi f_0)\right)$

$x_2(t) = rect\left(\frac{t-t_0}{T_0}\right) \xrightarrow[\text{from part g}]{FT} X_2(\omega) = e^{-j\omega t_0}\cdot T_0 \cdot sinc(\omega T_0/2\pi)$

using time domain multiplication property and sifting property

$$\boxed{X(\omega) = \frac{T_0}{2}\cdot e^{-j\omega t_0}\left(e^{j2\pi f_0 t_0} sinc\left(\frac{\omega T_0}{2\pi} - 1\right) + e^{-j2\pi f_0 t_0} sinc\left(\frac{\omega T_0}{2\pi} + 1\right)\right)}$$

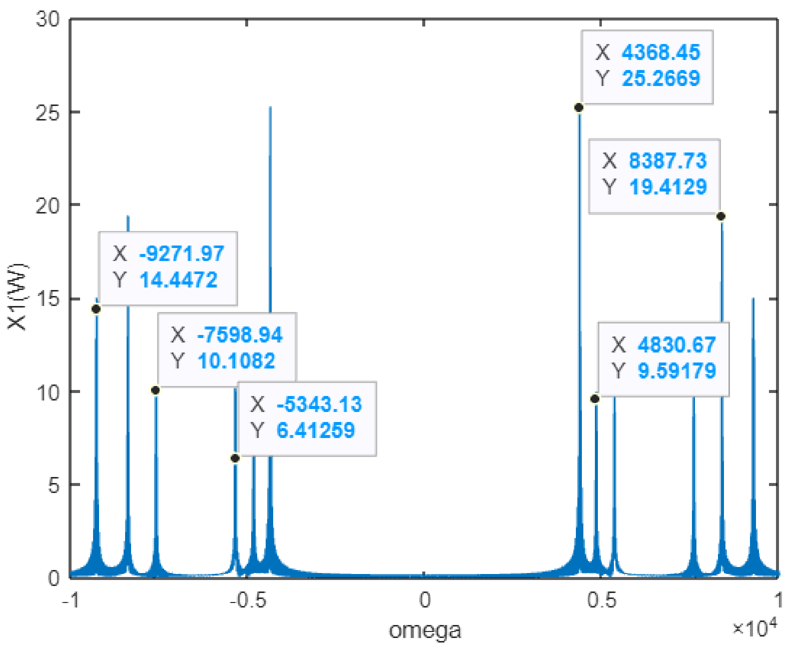

Fig1.1 Fourier Transform of 26912

From the relation $\omega = 2\pi f$, the omega values that can be observed in Fig1.1 is converted to Hz in order to see if the frequencies are precise with the input.

omega(ω)	$f(Hz)$
4368.45	$\cong$697 Hz
4830.67	$\cong$770 Hz
8387.73	$\cong$1336 Hz
5343.13	$\cong$852 Hz
7598.94	$\cong$1209 Hz
9271.97	$\cong$1477 Hz

Table1.2 frequency values

As it can be observed from Table 1.2, frequency content of the fourier transform of 26912 is precise with the corresponding values in Table1.1. This fourier transform only outputs the frequency content of the input hence the order of the dialed number cannot be observed. However by multiplying the $x(t)$ with a rectangular signal in order to decide exactly what the input is. The analytical expression for the rectangular signal to determine first input number is as following.

$$x_1(t) = \begin{cases} x(t), & 0 \leq t \leq 0.25 \\ 0, & otherwise \end{cases}$$

The Matlab code in order to apply fourier transform to the first input number is as following.

```
Number1=[2 6 9 1 2];
x=DTMFTRA(Number1);
rect= zeros(size(x));
rect(1:2049)=ones(1,2049)
x1=rect.*x;
X=FT(x1);
omega=linspace(-8192*pi,8192*pi,10246);
omega=omega(1:10245) ;
plot(omega,abs(X))
xlabel('omega');
ylabel('X1(W)');
xlim([-10000,10000]);
```

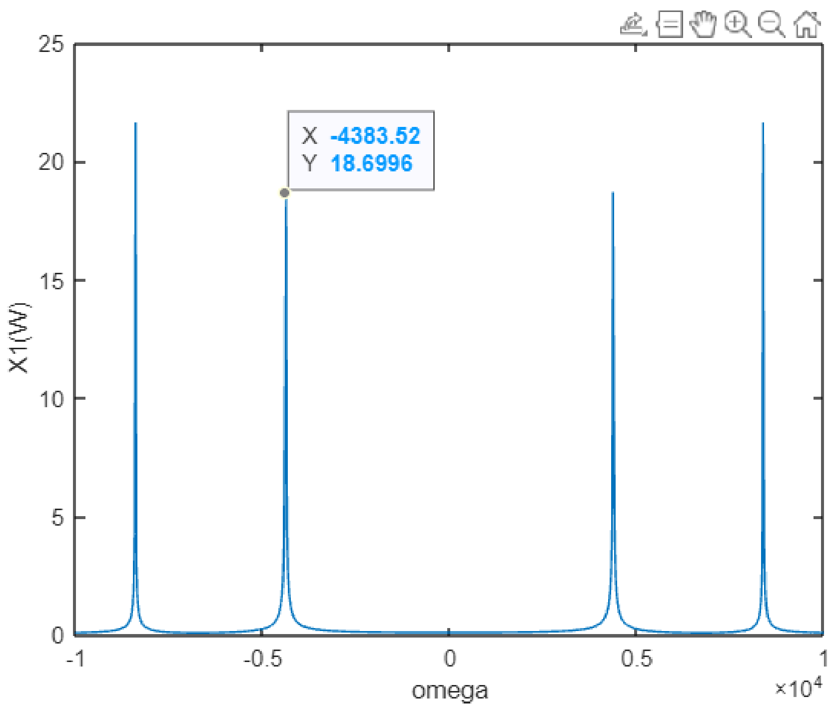

Fig1.2 Fourier Transform of 2

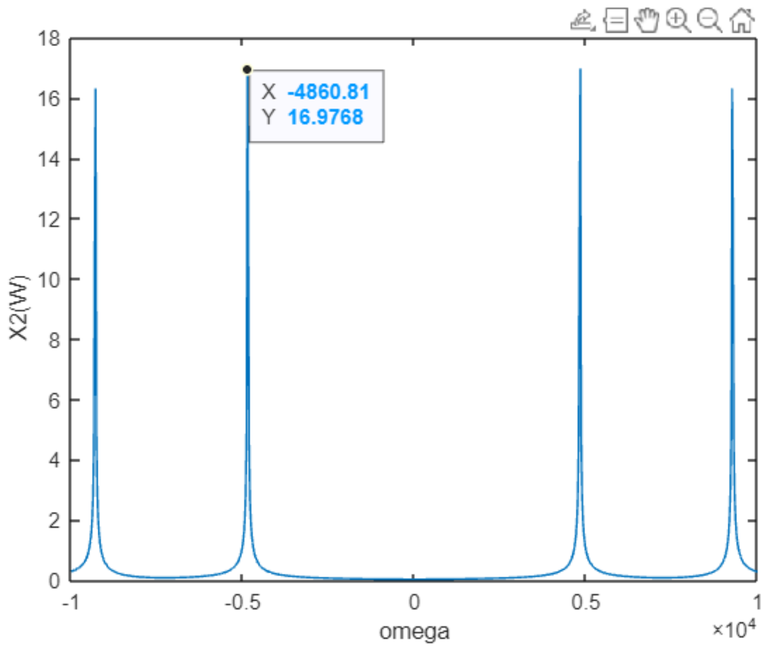

Fig1.2 Fourier Transform of 6

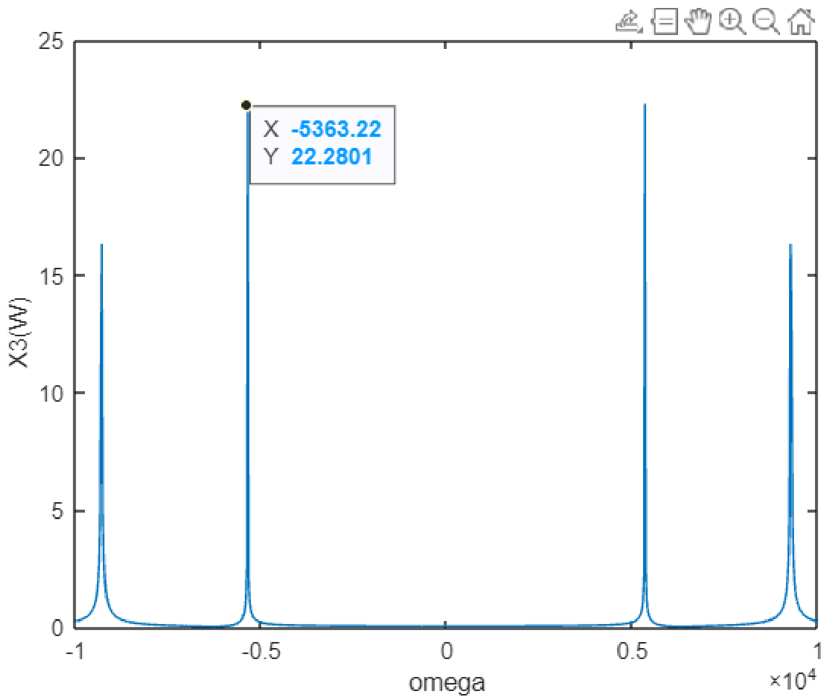

Fig1.2 Fourier Transform of 9

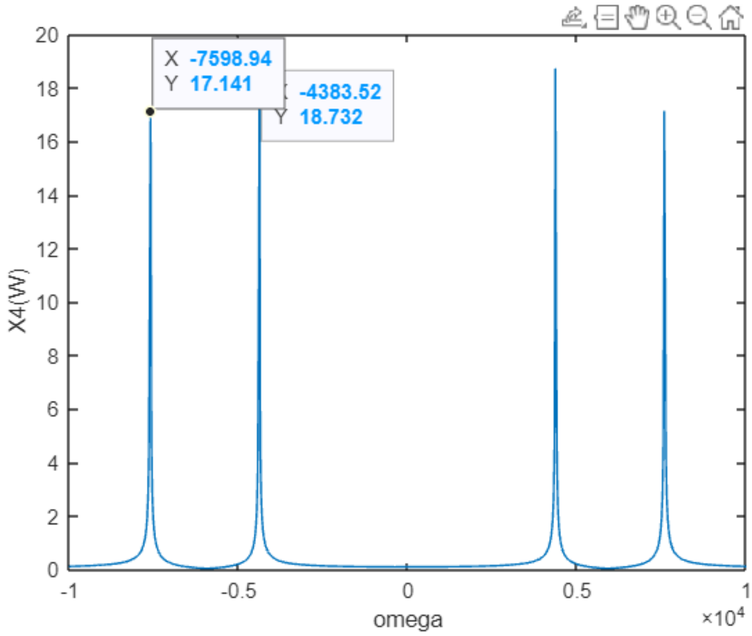

Fig1.2 Fourier Transform of 1

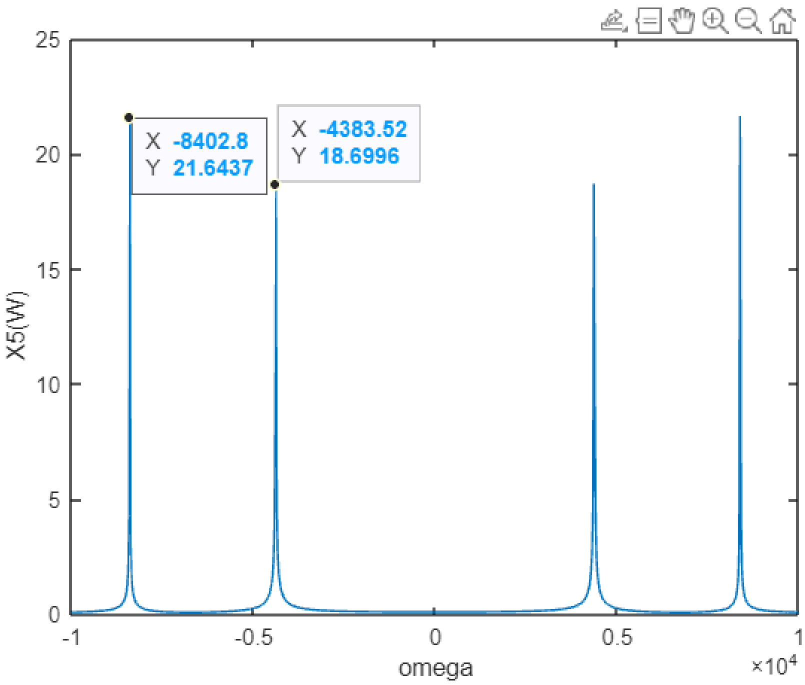

Fig1.2 Fourier Transform of 2

There are 4 peaks in each fourier transform this is a result of numbers being the sum of two cosines with different frequencies. By looking at the 2 peaks with different amplitudes frequency content of the input can be determined hence the input number is determined. In order to verify, number 1 which is the fourth input is taken. $\omega = 2\pi f,\ f_1 = \cong 1209\ Hz$ and $f_2 = \cong 697\ Hz$. After verifying the fourth input it can be concluded as the input can be extracted by looking at fourier transform vs omega graphs. DTMF transmitter and reciever are established successfully.

Part 2)

The code for recording a sound and plotting is written as following.

```
      t=[0:1/8192:12-1/8192];

recObj = audiorecorder(8192, 16, 1);
recordblocking(recObj, 12);


x = getaudiodata(recObj);
desired_length = 8192 * 12;
if length(x) > desired_length
    x = x(1:desired_length);
end


soundsc(x,8192);
plot(t,x);
xlabel('time');
ylabel('x');
title('plot of sound record')
```

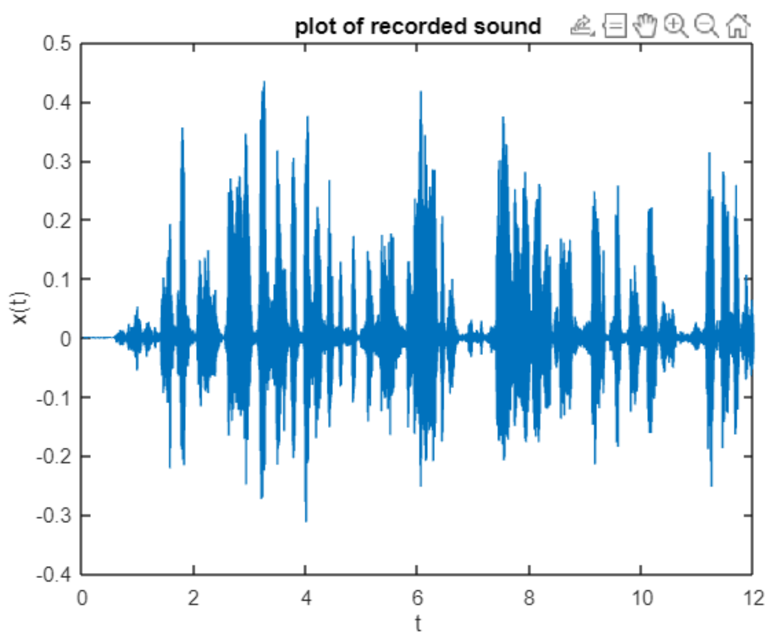

Fig2.1 $x(t)\ vs\ t$

Now it is desired to generate a new signal $y(t)$. The relationship between $y(t)$ and $x(t)$ is given below. From this relation it can be interpreted that $y(t)$ is sum of shifted $x(t)'s$ which is echoed version of $x(t)$.

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

(a) $h(t)$ is the output when $\delta(t)$ is the input (impulse response,

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

(b) $h(t) \xrightarrow[FT]{} \boxed{H(\omega) = 1 + \sum_{i=1}^{M} A_i \cdot e^{-j\omega t_i}}$

FT of $\delta(t)$

$\delta(t-t_0) \xrightarrow[FT]{} e^{-j\omega t_0}$

(c) $y(t) = x(t) * h(t)$, $\boxed{Y(\omega) = X(\omega) . H(\omega)}$

Convolution in time domain corresponds to multiplication in Fourier domain.

(d) From c, $Y(\omega) = X(\omega) . H(\omega)$, $\boxed{X(\omega) = \dfrac{Y(\omega)}{H(\omega)}}$

The specifications for $y(t)$ is given(M and ti array) in the lab report. It is desired to obtain the original speech from the echoed speech. First fourier transform of $h(t)$ is obtained by using the property in d. After that by using inverse fourier transform $h(t)$ is obtained which is classified as impulse response. Finally the estimated speech is obtained again by using the property in d and then applying IFT to the result. The matlab code for the outputs is given at the end of the report and the desired outputs can be observed in the following figures.

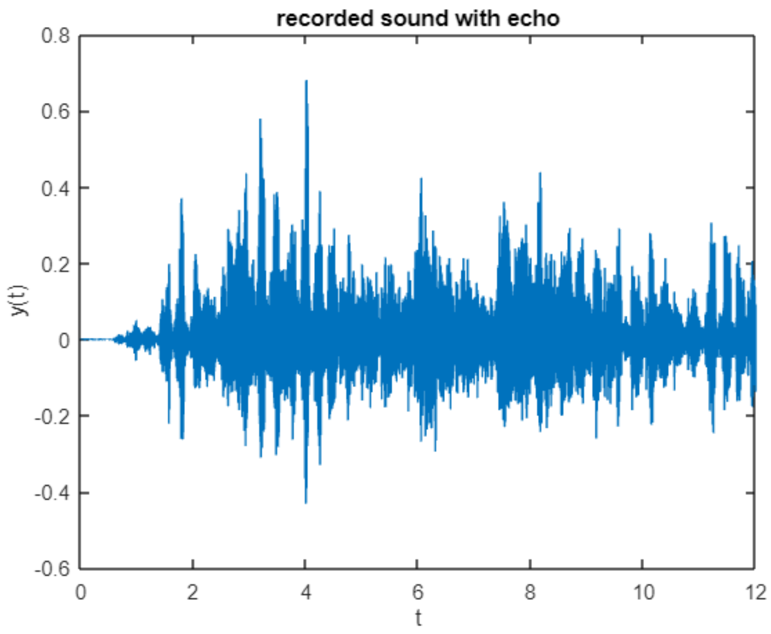

Fig2.2 $y(t)\ vs\ t$

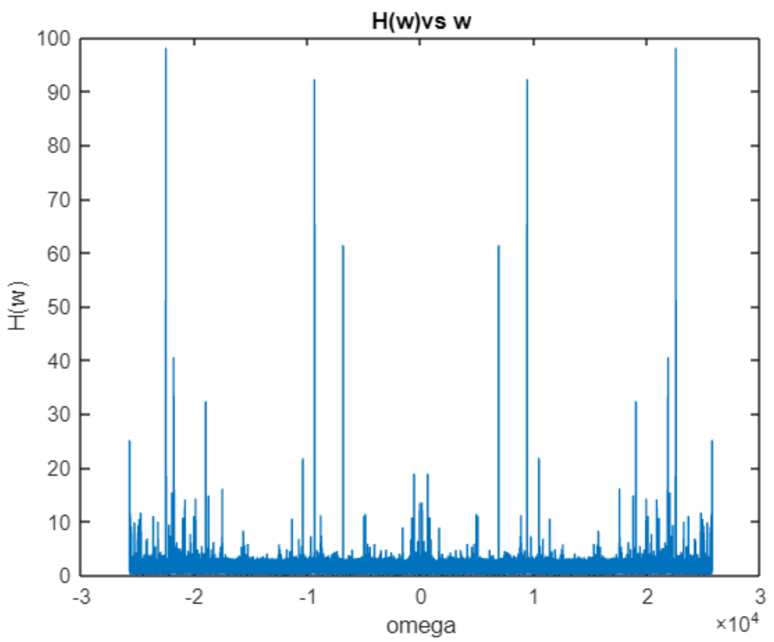

Fig2.3 Fourier Transform of $h(t)$ by using the propeety in d

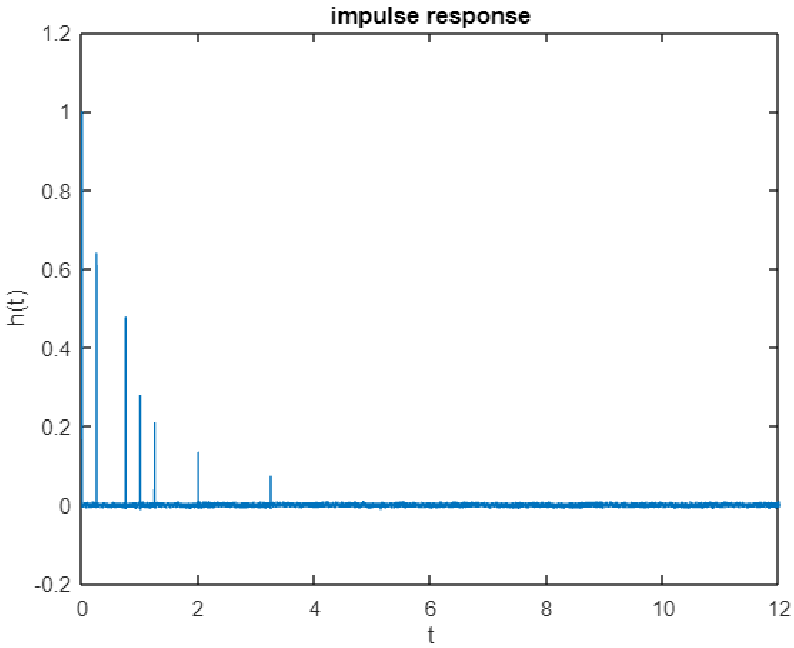

Fig2.4 $h(t)\ vs\ t$

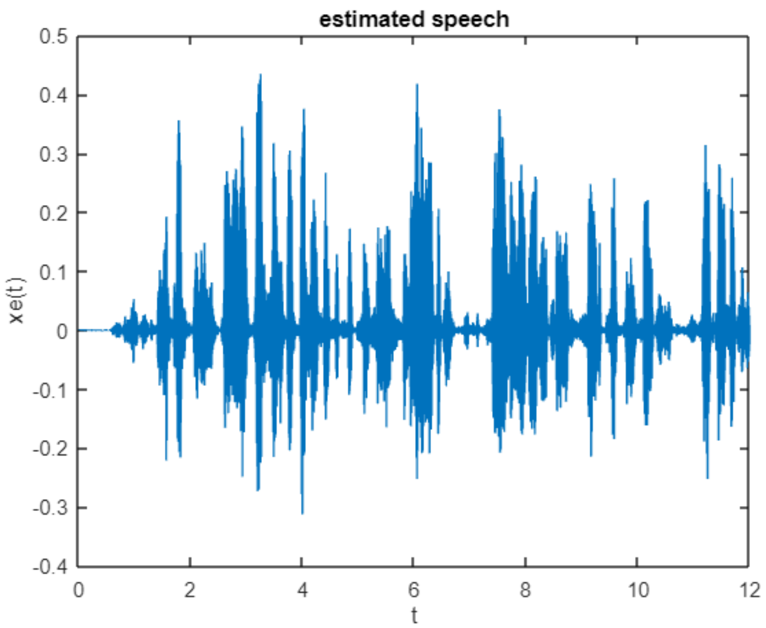

Fig2.5 estimated speech vs t

After listening the estimated speech, the original speech is significantly redeemed. Also it can be observed that plot of the estimated speech is similar to the original speech plot which is another indicator that the operation is successful.

Added code to the initial code which is for recording and plotting the speech:

```
M = 6;
Ai = [0.65, 0.50, 0.30, 0.22, 0.15, 0.1];
ti = [0.25, 0.75, 1, 1.25, 2, 3.25];
y = x;
new_signal = zeros(size(x));
for i = 1:M
    delay1 = ti(i) * 8192;
    new_signal(delay1+1:end) = x(1:end-delay1);
    y = y + Ai(i) * new_signal;
end


Y=FT(y);
X=FT(x);
H=FT(y)./FT(x);
omega=linspace(-8192*pi,8192*pi,98305);
omega=omega(1:98304);
h=IFT(H);
H=FT(h);
Xe=FT(y)./FT(h);
xe=IFT(Xe);

figure;
plot(t,x);
ylabel('x(t)');
xlabel('t');
title('plot of recorded sound');

figure;
plot(t,y);
ylabel('y(t)');
xlabel('t');
title('recorded sound with echo');

figure;
plot(omega,abs(H));
ylabel('H(w)');
xlabel('omega');
title('H(w)vs w');

figure;
plot(t,h);
ylabel('h(t)');
xlabel('t');
```

```
title('impulse response');

figure;
plot(t,xe);
ylabel('xe(t)');
xlabel('t');
title('estimated speech');

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
```