GENERAL SIGNAL GENERATOR

mini project ||

- **Eslam Abdellatif Dyab** **18010333**
- **Eslam Ashraf Muhammed** **18010324**
- **Mayar Gamal Samada** **18011918**
- **Mai Bendary Soliman** **18011912**

The code

```
% read the sound file and play it
[xin ,fs] = audioread('dawn of faith.mpeg');
%sound(xin,fs);

% asking for the value of segma
fprintf('please enter the number of impulseResponse:\nType 1 for H1\nType 2 for H2\nType 3 for H3\nType 4 for H4\n');
impulseResponse = input('The number= ');

sigma=input('please enter the value of sigma: ');

% geting number of channels
channelsNum = size(xin);
channelsNum = channelsNum(2);

% empty vector to concatinate the channels in it
finalOutputSignal = zeros(length(xin),channelsNum);
finalNoiseySignal = zeros(length(xin),channelsNum);
```

- **At first we enter the sound file then we asked the user to enter the sigma and the impulse response he wants to apply on the sound**
- **We created two empty vectors to concatenate the channels in it**

```
i=1;
while (i <= channelsNum)
    [outputSignal,noiseSignal] = communicationSystem(xin(:,1),fs,sigma,i,impulseResponse);
    finalOutputSignal(:,i) = outputSignal;
    finalNoiseySignal(:,i) = noiseSignal;

    i = i+1;
end

% play output sound file
sound(finalNoiseySignal,fs);
pause(25);

%play output sound file
sound(finalOutputSignal,fs);
audiowrite('out.wav', finalOutputSignal, fs);
```

- **We created a while loop to find the final signal of all the channels that we have then play it**

```
function [outputSignal,noiseSignal] = communicationSystem(channel,fs,sigma,i,impulseResponse)
% 1) Transmision
%----------------------------------------%
x = channel;
t_end = length(x)./fs;
t = linspace(0,t_end, t_end*fs);

%plot in time domain
if (i==1)
    figure(1)
    subplot(3,1,1)
    plot(t,x)
    title('Signal in time domain')
end

X = fftshift(fft(x));
Xmg = abs(X);
Xphase = angle(X);
N = length(x);
fvec = linspace(-fs/2,fs/2,N);

%plot in frequency domain
if (i==1)
    subplot(3,1,2)
    plot(fvec,Xmg)
    title('Signal Magnitude in frequency domain')
end

if (i==1)
    subplot(3,1,3)
    plot(fvec,Xphase)
    title('Signal Phase in frequency domain')
end
```

- **We created a function that gives us the output signal and noise signal**
- **Then we plot the signal in time domain and frequency domain.**
- **The if before every plot is because we plot only the first channel**

```
% 2) Channel
%-----------------------------------------%
% A) Delta
% the output signal is the same as the input
if (impulseResponse ==1)
    H1 = [1 zeros(1,N-1)];
    y = conv(x,H1);

    t_end = length(y)./fs;
    t_conv = linspace(0,t_end, t_end*fs);

    y = y';
end
% B) exp(-2pi*5000t)
% the system amplifies the signal volume by approximately 2
if (impulseResponse == 2)
    H2 = exp(-2*pi*5000*t);
    y = conv(x,H2);

    t_end = length(y)./fs;
    t_conv = linspace(0,t_end, t_end*fs);

    y = y';
end
% C) exp(-2pi*1000t)
% the system amplifies the signal volume by approximately 4
if(impulseResponse == 3)
    H3 = exp(-2*pi*1000*t);
    y = conv(x,H3);

    t_end = length(y)./fs;
    t_conv = linspace(0,t_end, t_end*fs);

    y = y';
end
```

```
  % D) The channel has the following impulse response
  % the system somehow disturbe the signal (overlab)
  if (impulseResponse == 4)
      H4 = [2 zeros(1,1*fs -2) 1];
      y = conv(x,H4);

      t_end = length(y)./fs;
      t_conv = linspace(0,t_end, t_end*fs);
  end

% ploting the signal after applying system
if (i==1)
        figure(2)
        subplot(3,1,1)
        plot(t_conv, y)
        title('Signal in time domain after applying system')

        Y = fftshift(fft(y));
        Ymg = abs(Y);
        Yphase = angle(Y);
        N = length(Y);
        fvec = linspace(-fs/2,fs/2,N);

        subplot(3,1,2)
        plot(fvec, Ymg)
        title('Signal Magnitude in freq. domain after applying system ')

        subplot(3,1,3)
        plot(fvec, Yphase)
        title('Signal phase in freq. domain after applying system ')
end
```

```
% 3) noise
%----------------------------------------%
% play sound file after adding noise
z = sigma*randn(length(x),1);
noiseSignal = y(1:length(x)) + z;
x = noiseSignal;
% plot the sound file after adding noise in time domain
t_endN=length(x)./fs;
t=linspace(0,t_endN,t_endN*fs);

if (i==1)
    figure(3)
    subplot(2,1,1)
    plot(t,x)
    title('sound file after adding noise in time domain')
end
% plot the sound file after adding noise in frequency domain
xf=fftshift(fft(x));
xfmg=abs(xf);
N=length(x);
fvec=linspace(-fs/2,fs/2,N);

if (i==1)
    subplot(2,1,2)
    plot(fvec,xfmg)
    title('sound file after adding noise in frequency domain')
end
```

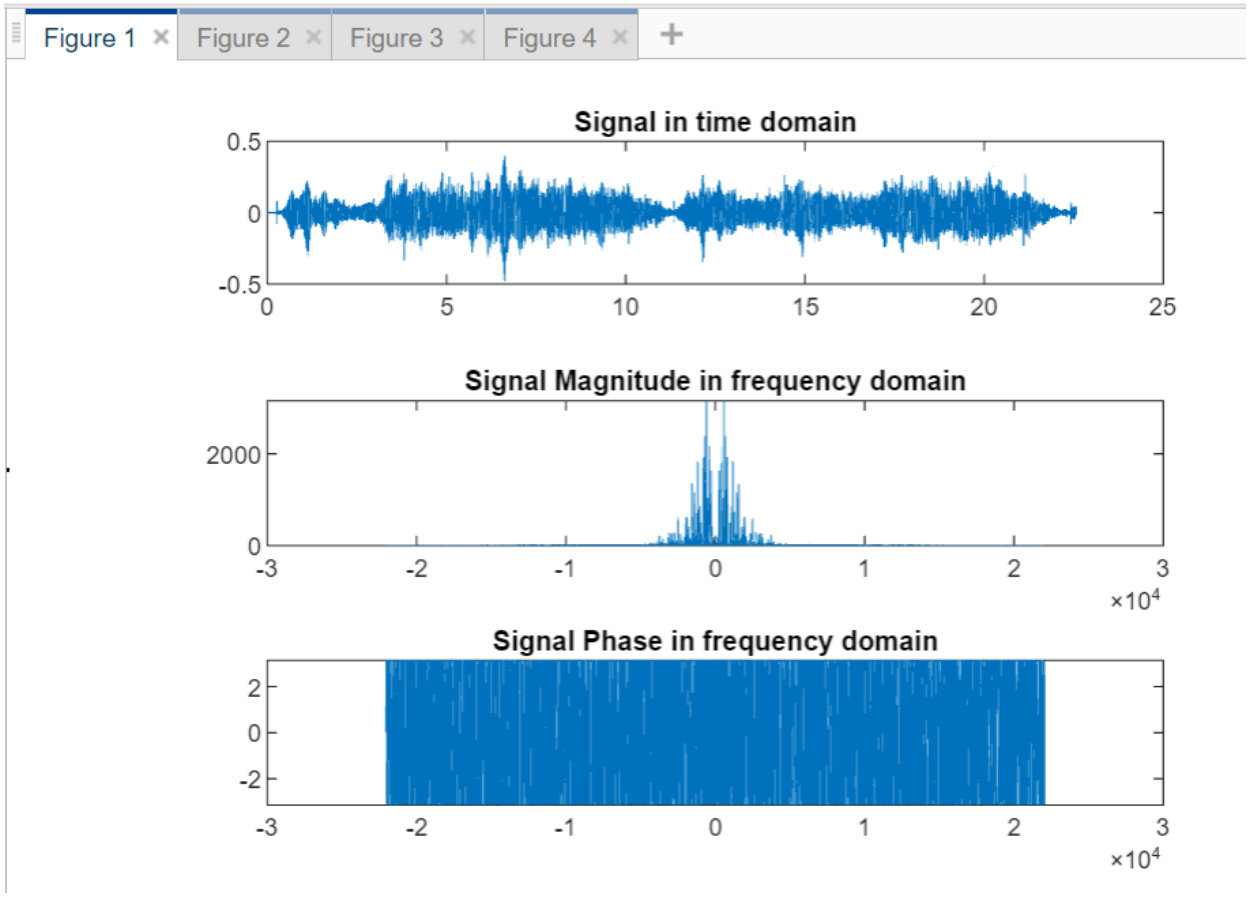
- **The user has already enter the value of sigma we use it in noise equation then we add this noise to the signal**

- **The if before every plot is because we plot only the %first channel**

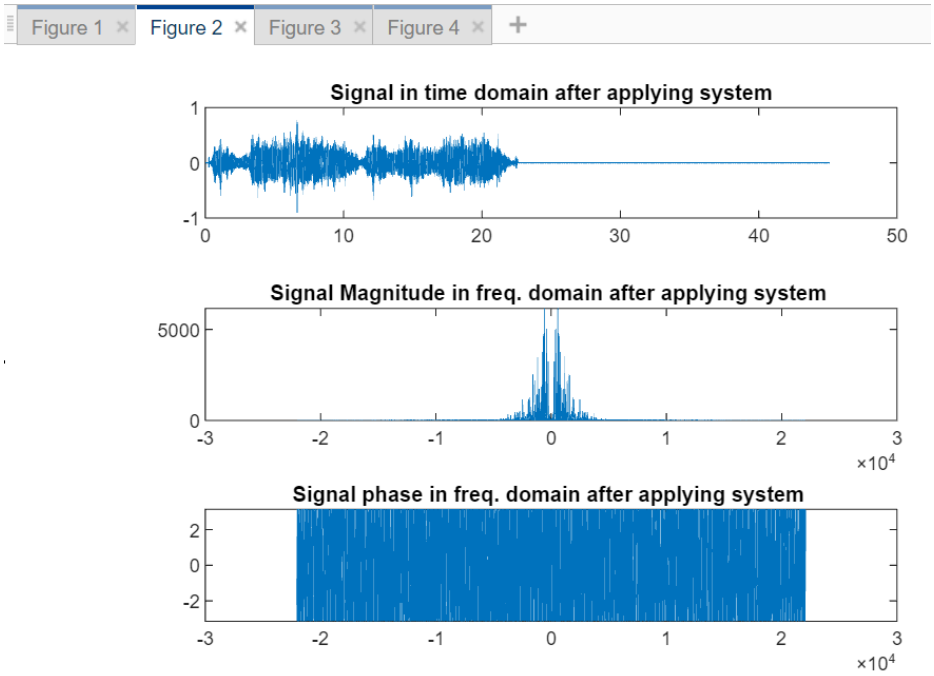
```
% 4) Receiver
%----------------------------------------%
N = length(x);
n = N/fs;
right_band = round((fs/2-3400)*n);
left_band = (N-right_band+1);
xf([1:right_band left_band:N]) = 0;

x = real(ifft(ifftshift(xf)));

% plot the sound file at receiver in time domain
t_end = length(x)./fs;
t=linspace(0,t_end,t_end*fs);

if (i==1)
    figure(4)
    subplot(2,1,1)
    plot(t,x)
    title('sound file at receiver in time domain')
end


% plot the sound file at receiver in frequency
Xmg=abs(xf);
N=length(x);
fvec=linspace(-fs/2,fs/2,N);

if (i==1)
    subplot(2,1,2)
    plot(fvec,Xmg)
    title('sound file at receiver in frequency domain')
end

outputSignal = x;
end
```

The Plots

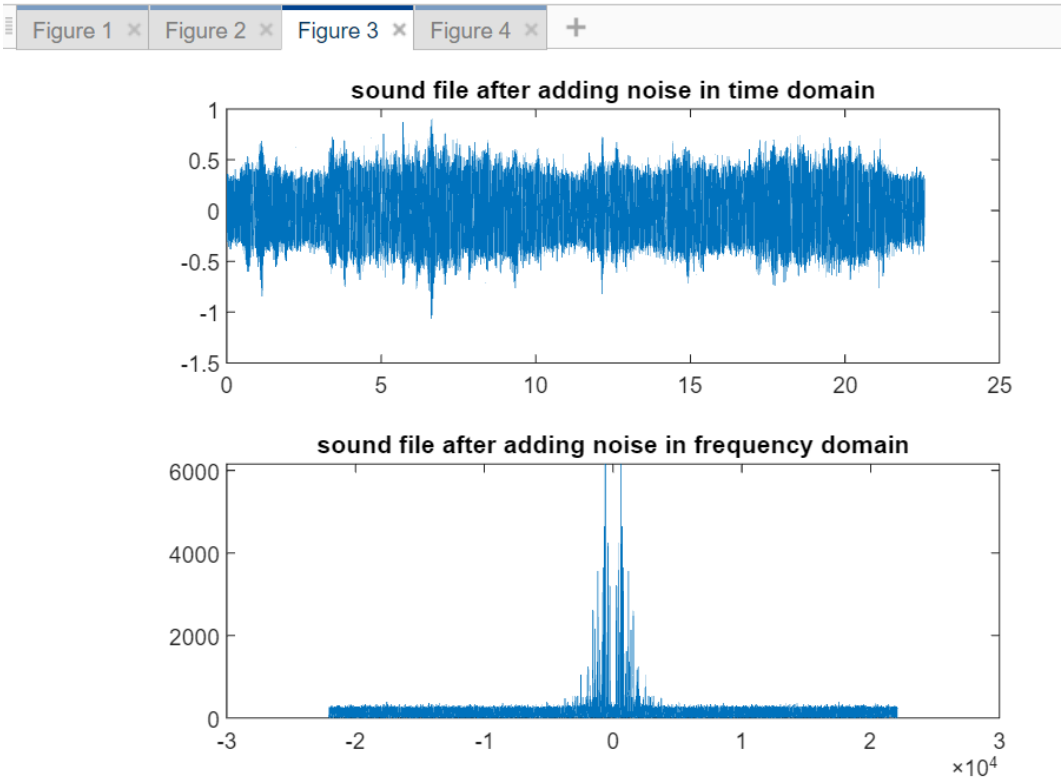
Original signal in time domain and frequency domain :



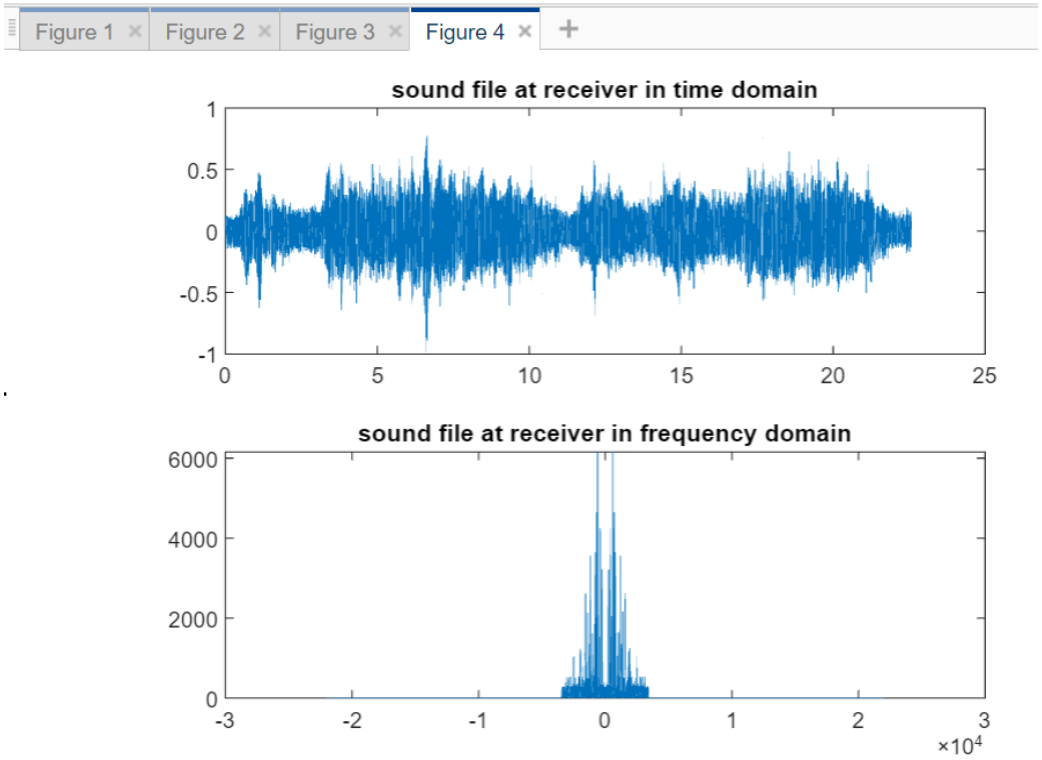
Signal after applying the second system in time domain and frequency domain :



Signal after adding noise :



Signal at receiver :



Signal after applying all systems in time domain