Amrita Vishwa Vidyapeetham

Amrita School of Engineering, Amritapuri

Accredited by NACC with “A” Grade

Department of Electronics and Communication Engineering

TEAM – 5

REMOVING THE AWGN NOISE

Team Members:

Sirikonda Venkata Bala Harini - AM.EN.U4ECE18165
Karasani Jyothika - AM.EN.U4ECE18162
Nandigam Mani Srikara Yaswanth - AM.EN.U4ECE18135
Maddala Vamsi Krishna - AM.EN.U4ECE18131
Kamma Madhulatha - AM.EN.U4ECE18124

Acknowledgement:

We would like to express our special thanks of gratitude to our course faculty Anuraj K and Poorna S S as well as our principal Dr. Jyothi S N who gave us the golden opportunity to do this wonderful project on the topic “Removing the AWGN Noise”, which also helped us in doing a lot of Research and also we came to know about so many new things. We are really thankful to them. Secondly, we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

Abstract:

Additive White Gaussian Noise (AWGN) degrades the performance of wireless communication systems. In this paper a method is discussed how to remove the Additive white Gaussian noise of an audio signal and retrace the original audio signal. In order to do this, we are considering an audio signal to that signal we are adding an AWGN and passing it through a low pass filter. Output of the low pass filter is approximately same as the original signal.

Key words: Additive White Gaussian Noise, Low Pass Filter.

Introduction:

Noise is an unwanted signal which interferes with the original signal and corrupts the parameters of the message signal. In any communication system, during the transmission of the signal, or while receiving the signal, some unwanted signal gets introduce into the communication, making it unpleasant for the receiver, questioning the quality of the communication. Such a disturbance is called Noise.

Additive white Gaussian noise is a basic model of noise. It id additive because it is added to any noise that might be intrinsic to the system. White indicates that it has uniform power across the frequency band and white colour has uniform emissions at all frequencies in visible spectrum. It is Gaussian because, it has a normal distribution in the time domain with an average time domain value is zero.

Low Pass Filter (LPF) is a filter that passes signals with a frequency lower than a selected cut-off frequency and attenuates signals with frequency higher than the cut-off frequency. These filters are mainly used to filter noise, since noise is a high frequency signal.

Principle:

In communication systems we are facing a lot of noise in addition to the signal at the receiver end. It is questioning the quality of the communication. In order to remove the extra noise in the signal we are going to use a low pass filter which attenuates signals with frequency greater than the selected cut-off frequency. For an audio signal we would pass noise + audio signal through a low filter with a cut-off frequency of about 20 kHz.

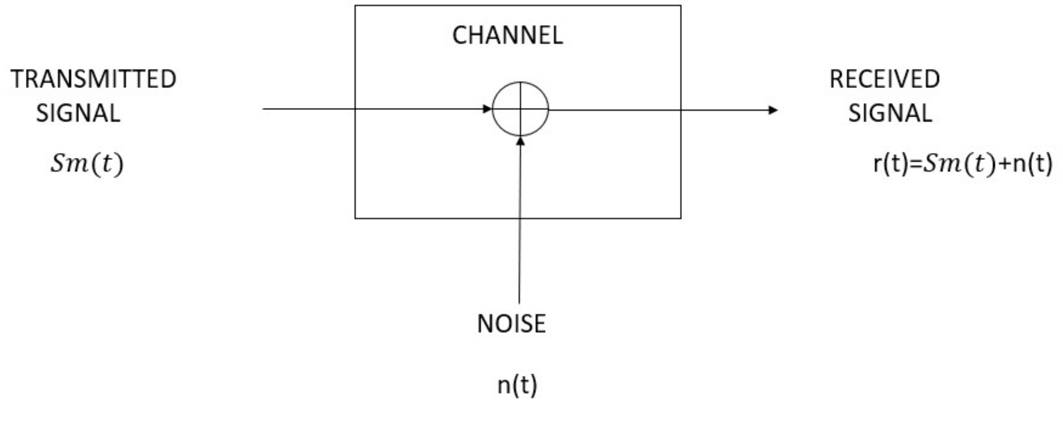


Figure 1 Block Diagram

n(t) denotes the sample function of the AWGN process with power spectral density No/2

Working:

I. Taking any audio or music as the input signal.
   We are taking an audio file from local disc using an inbuilt command audioread. It reads the signal with a sample frequency Fs. By using plot command, we are plotting the audio signal with amplitude on y-axis and samples on x-axis.
   Sound command is used to speak out the original signal on platforms that support sound. Pause command is used to pause the run time to avoid overlapping.

II. Adding the White Gaussian Noise.
   By using an inbuilt command awgn we adding noise to the original audio signal with some specific Signal to Noise ratio (SNR). By using plot command, we are plotting the noise added signal with amplitude on y-axis and samples on x-axis.
   Sound command is used to speak out the noise added signal on platforms that support sound. Pause command is used to pause the run time to avoid overlapping.

III. Passing the signal through a Low Pass Filter.
   Using a command called fdesign.lowpass by considering passband frequency, stopband frequency, passband ripple and stopband attenuation we are designing a low pass filter function. By using a commands called design and freqz we are extracting the frequency response of the digital filter. By using filter function with frequency and noised signal we are removing high frequency fluctuations in the audio signal. By using plot command, we are plotting the noise removed signal with amplitude on y-axis and samples on x-axis.
   Sound command is used to speak out the noise removed signal on platforms that support sound. Pause command is used to pause the run time to avoid overlapping.

IV. Plotting all the three signals and observing the difference.
   By using subplot commands we are plotting all the three graphs in one plot for easy observation.

Code:

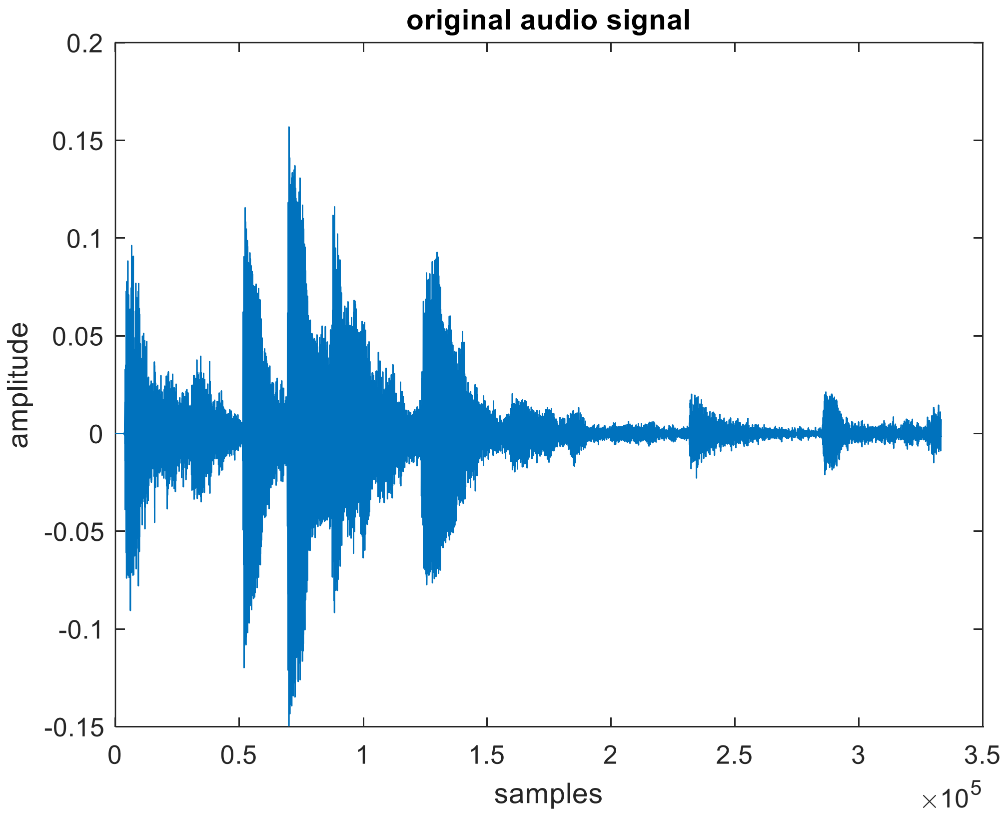
```
%%
% step-1
clc
clear all
[y,Fs]=audioread('C:\Users\91944\Downloads\testvoice1.ogg'); %Taking an audio file from the localdisc
n=1/Fs;
figure,
plot(y);
xlabel('samples');
ylabel('amplitude');
title('original audio signal');
sound(y,Fs) %sound(Y,FS) sends the signal in vector Y (with sample frequency FS) out to the speaker on platforms that support sound.
pause(12)   % pause(n) pauses for n seconds before continuing,
```

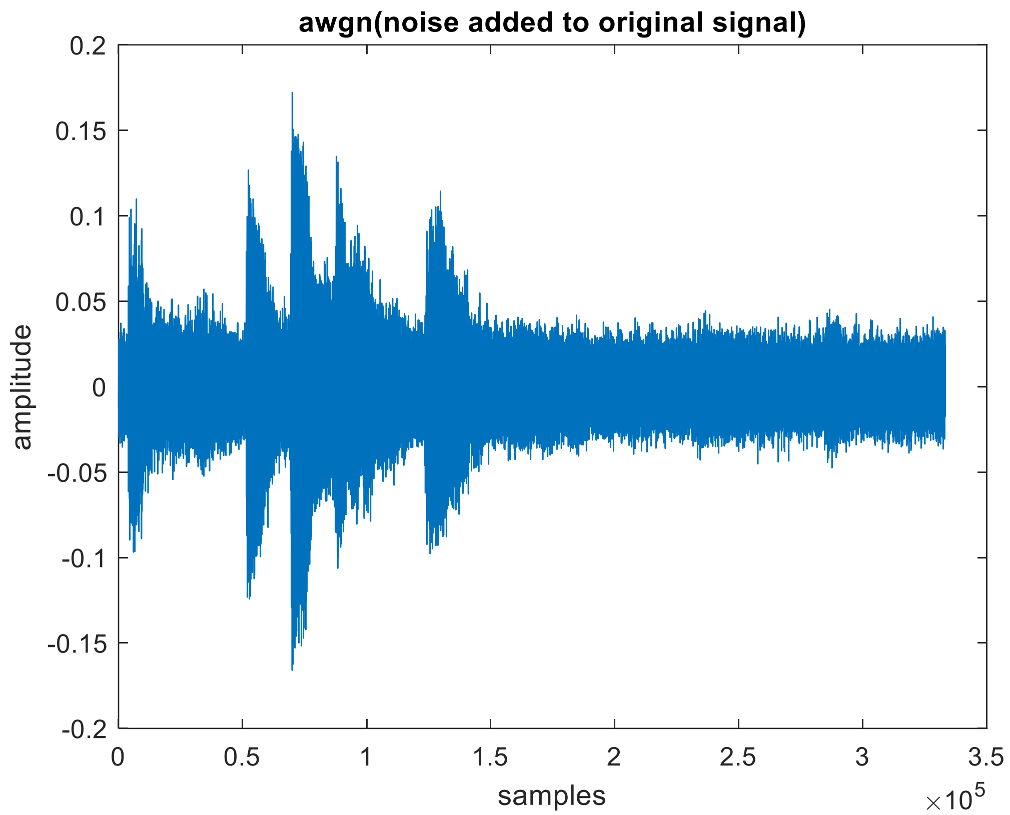
```
%%
%step-2
y = awgn(y,40);% here 40 is the Signal to noise ratio(SNR)
noisy = y;
figure,
plot(y);
xlabel('samples');
ylabel('amplitude');
title('awgn(noise added to original signal)');
sound(noisy,Fs)
pause(12)
```

```
%%
%step-3

%Fp = passband frequency
%Fst = stopband frequency
%Ap = passband ripple
%Ast = stopband attenuation

filteredsignal = fdesign.lowpass('Fp,Fst,Ap,Ast',1e-10,0.1,0.1,1e-5);%lowpass filter function
D = design(filteredsignal);
freqz(D);%Frequency response of digital filter
x = filter(D,noisy);
figure,plot(x);
title('denoise');
xlabel('samples');
ylabel('amplitude');
sound(x,Fs)
pause(12)
```

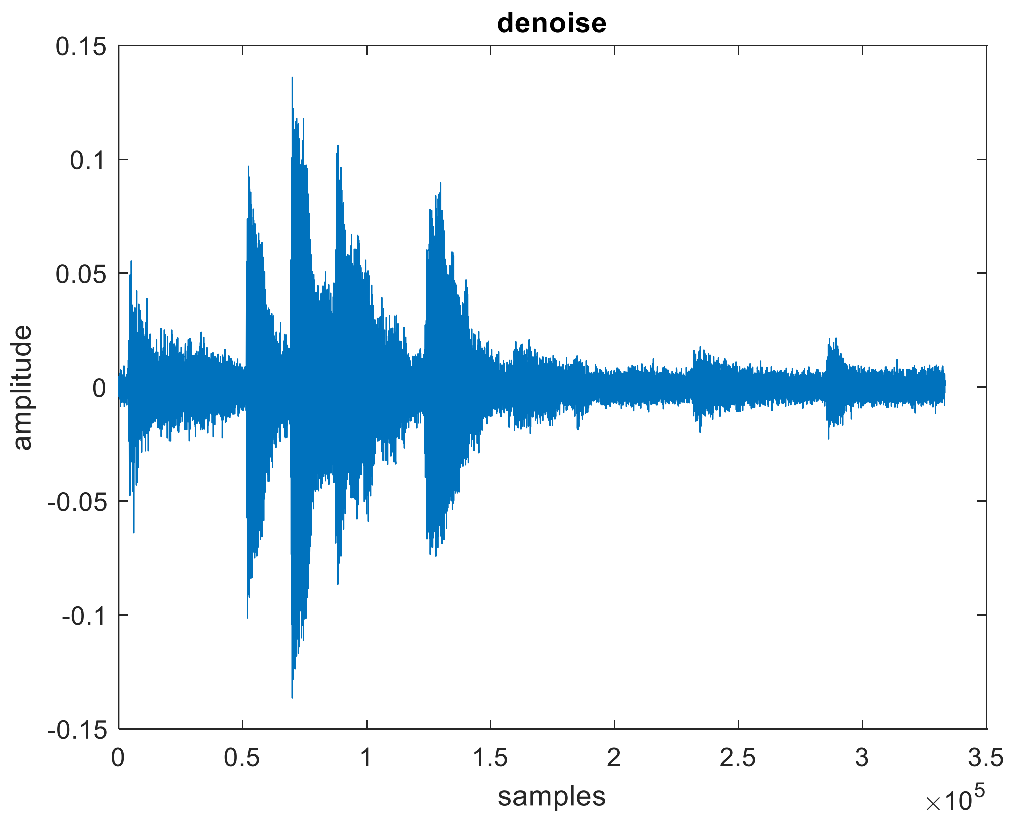
```
%%
%step-4
figure,
subplot(2,2,1)
plot(y);
xlabel('samples');
ylabel('amplitude');
title('original audio signal');
subplot(2,2,2)
plot(y);
xlabel('samples');
ylabel('amplitude');
title('awgn(noise added to original signal)');
subplot(2,2,3)
plot(x);
title('denoise');
xlabel('samples');
ylabel('amplitude');
```

Result:

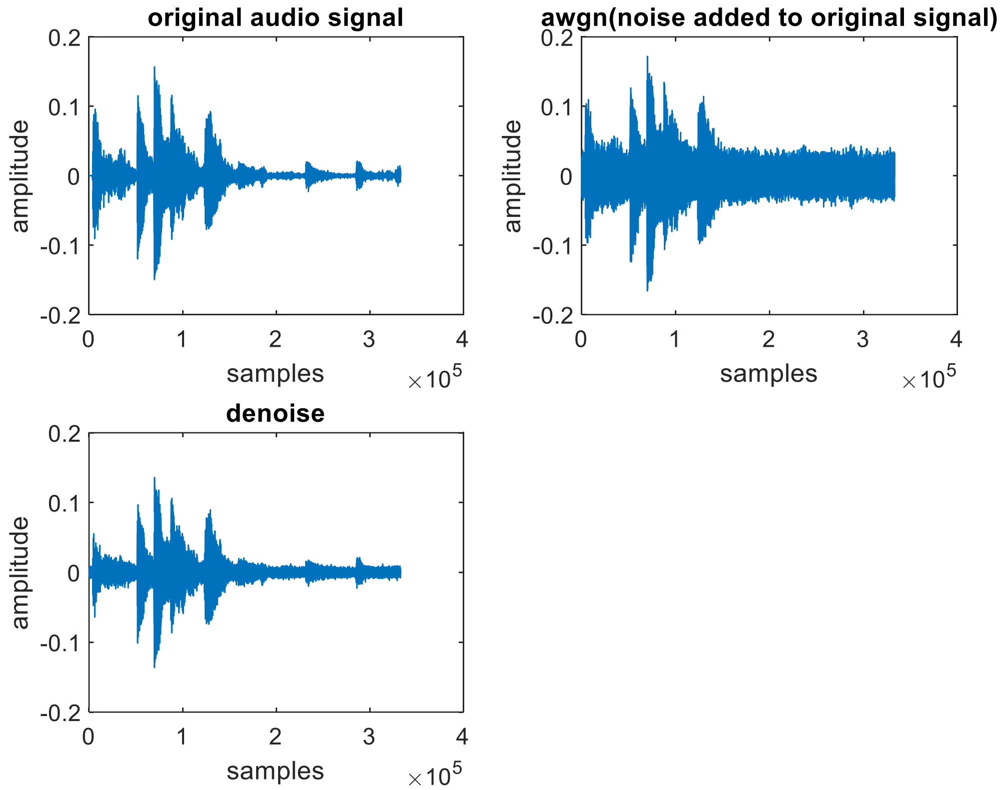




awgn(noise added to original signal)
amplitude
samples
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
0
0.5
1
1.5
2
2.5
3
3.5
×10^5



denoise
amplitude
samples
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
0.5
1
1.5
2
2.5
3
3.5
×10^5



Conclusion:

We were able to reconstruct the audio signal by removing the additive white Gaussian noise from the audio signal by passing it through a low pass filter.