Digital Signal Processing Report
On
"Audio Spectral Analysis using DFT in MATLAB "

Submitted in the fulfilment of the
requirements For the Degree of
Bachelor of
Technology In
Electronics & Telecommunication
Engineering By

Snehal Prusty 2014110968
Archit Saini 2014110980
Utkarsh Singh 2014111001

Under the guidance of
Prof. Mr. Sudhir Kadam

Department of Electronics & Telecommunication Engineering
Bharati Vidyapeeth (Deemed to Be University)
College of Engineering, Pune – 4110043

Academic Year 2022-23

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)

COLLEGE OF ENGINEERING, PUNE – 4110043
DEPARTMENT OF ELECTRONICS &
TELECOMMUNICATION ENGINEERING

CERTIFICATE

Certified that the project report entitled, " Audio Spectral Analysis using DFT in MATLAB" is a bonafide work done by **Utkarsh Singh , Archit Saini & Snehal Prusty** in fulfilment of the requirements for the award of degree of Bachelor of Technology in Electronics & Telecommunication Engineering.

Date:

Prof. Mr. Sudhir Kadam

Project Coordinator

Examiner1:

Examiner2:

ACKNOWLEDGEMENT

We would like to extend our sincere gratitude to the Head of Department Electronics & Telecommunication, **Prof. S. K. Oza**, for nurturing a congenial yet competitive environment, which motivates all the students not only to pursue goals but also to elevate the Humanitarian level.
Inspiration and guidance are invaluable in every aspect of life, which we have received from our respected project guide **Prof. Dr. Sudhir Kadam** who gave us his careful and ardent guidance because of
which we are able to complete this project. More words won't suffice to express our gratitude to his untiring devotion. He undoubtedly belongs to the members of the artistic gallery who are masters in all aspects.
We would also like to thank all the faculty members who directly or indirectly helped us from time to time with their valuable inputs.

UTKARSH SINGH	2014111001
ARCHIT SAINI	2014110980
SNEHAL PRUSTY	2014110968

TABLE OF CONTENTS

ABSTRACT

A theory of short-term spectral analysis, synthesis, and modification is presented with an attempt at pointing out certain practical and theoretical questions.

The objective of this project is to develop a DSP-based approach for recording & analyze the audio file using spectrum graph of the audio. The proposed approach involves recording the signal, processing the signal & saving the signal. The project will be implemented using MATLAB, a powerful tool for signal processing.

INTRODUCTION

In today's world, there is a continuous need for processing of signal. The objective of this project is to make a spectral analysis of the recorded audio which can be further processed to perform various advanced processes like DFT which can help in audio analysis using magnitude and frequency graphs.

Here we are using MATLAB and its various functions to record an audio sample and display it visually on MATLAB plots.

METHODOLOGY

Recording:

In this step, we use audiorecorder function to start recording an audio sample. We use an audiorecorder object to record audio data from an input device such as a microphone for processing in MATLAB®.

The audiorecorder object contains properties that enable additional flexibility during recording. For example, you can pause, resume, or define callbacks using the audiorecorder object functions.

Record Blocking:

recordblocking (recorderObj, length) records audio from an input device, such as a microphone connected to your system, for the number of seconds specified by length. The recordblocking method does not return control until recording completes. recorderObj is an audiorecorder object that defines the sample rate, bit depth, and other properties of the recording.

Retrieving Audio Data:

getaudiodata(recorder) returns recorded audio data associated with audiorecorder object recorder in a double array y.

Converting Audio signal into DFT:

fft(X) computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.

- If X is a vector, then fft(X) returns the Fourier transform of the vector.
- If X is a matrix, then fft(X) treats the columns of X as vectors and returns the Fourier transform of each column.
- If X is a multidimensional array, then fft(X) treats the values along the first array dimension whose size does not equal 1 as vectors and returns the Fourier transform of each vector.

fftshift:

fftshift(X) rearranges a Fourier transform X by shifting the zero-frequency component to the center of the array.

- If X is a vector, then fftshift swaps the left and right halves of X.
- If X is a matrix, then fftshift swaps the first quadrant of X with the third, and the second quadrant with the fourth.
- If X is a multidimensional array, then fftshift swaps half-spaces of X along each dimension.

Saving the recorded audio using Audiowrite:

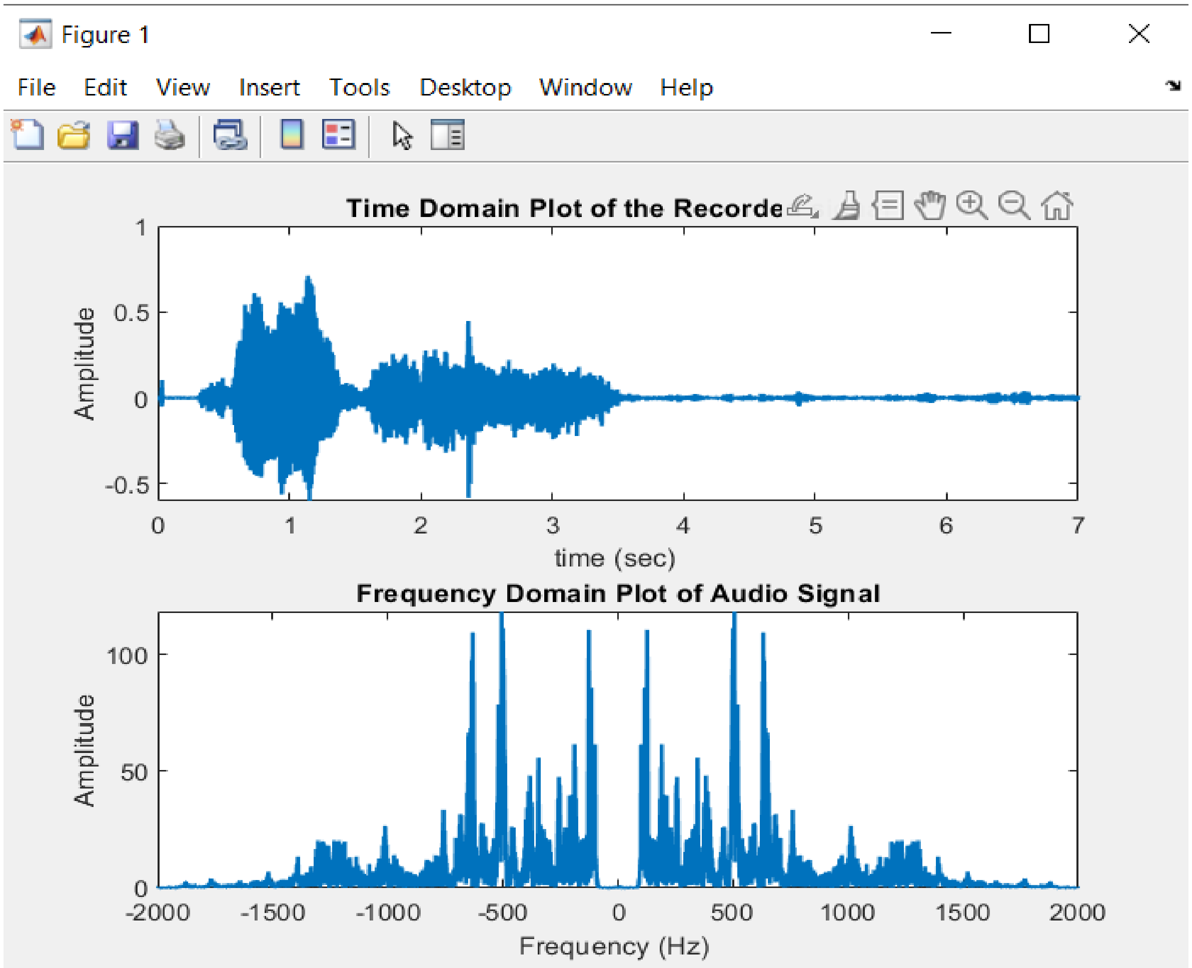
audiowrite(filename, y, Fs) writes a matrix of audio data, y, with sample rate Fs to a file called filename. The filename input also specifies the output file format. The output data type depends on the output file format and the data type of the audio data, y.

MATLAB CODE

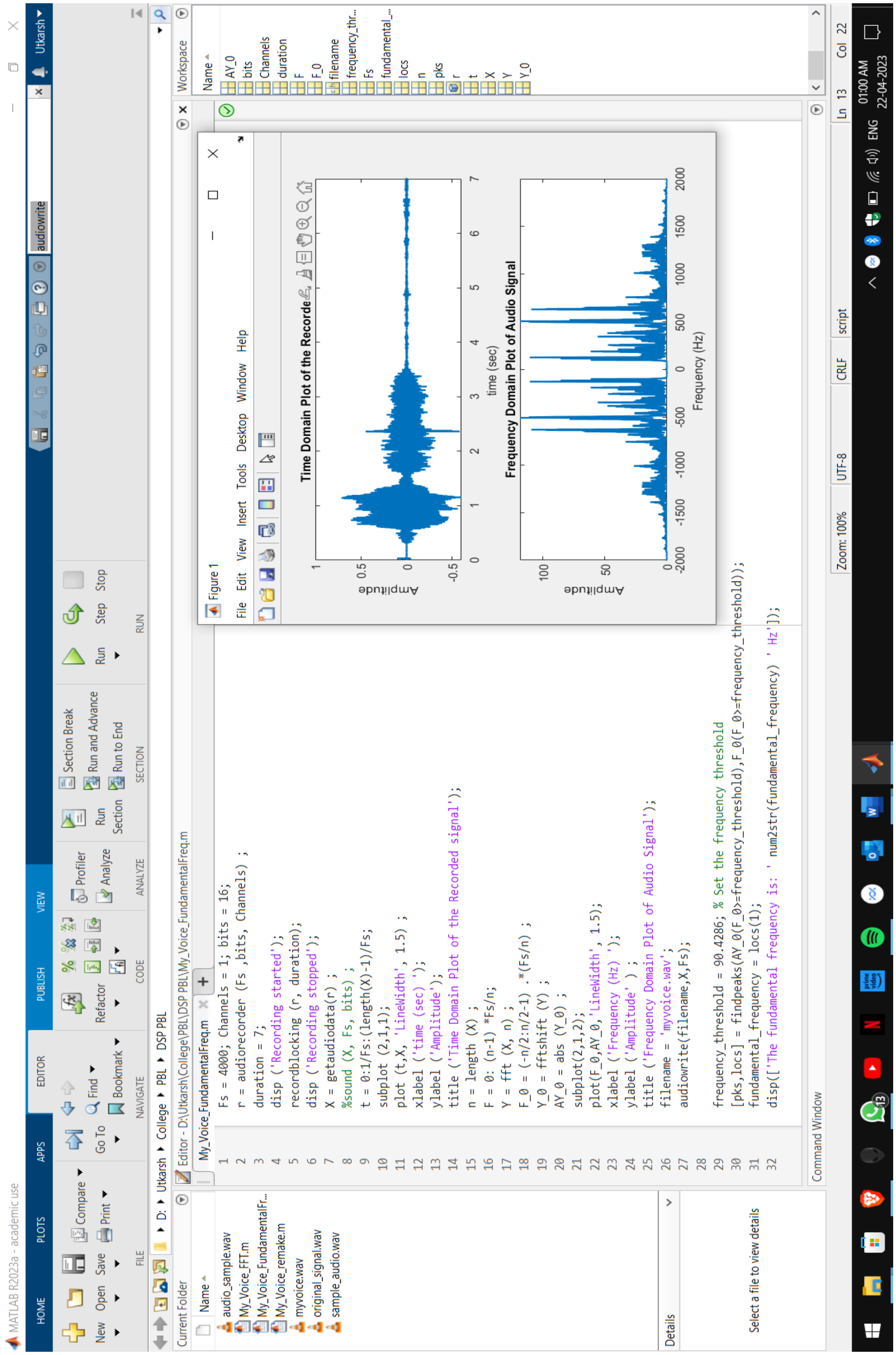
```
Fs = 4000; Channels = 1; bits = 16;
r = audiorecorder (Fs ,bits, Channels) ;
duration = 7;
disp ('Recording started');
recordblocking (r, duration);
disp ('Recording stopped');
X = getaudiodata(r) ;
%sound (X, Fs, bits) ;
t = 0:1/Fs:(length(X)-1)/Fs;
subplot (2,1,1);
plot (t,X, 'LineWidth', 1.5) ;
xlabel ('time (sec) ');
ylabel ('Amplitude');
title ('Time Domain Plot of the Recorded signal');
n = length (X) ;
F = 0: (n-1) *Fs/n;
Y = fft (X, n) ;
F_0 = (-n/2:n/2-1) .*(Fs/n) ;
Y_0 = fftshift (Y) ;
AY_0 = abs (Y_0) ;
subplot(2,1,2);
plot(F_0,AY_0,'LineWidth', 1.5);
xlabel ('Frequency (Hz) ');
ylabel ('Amplitude' ) ;
title ('Frequency Domain Plot of Audio Signal');
filename = 'myvoice.wav';
audiowrite(filename,X,Fs);


frequency_threshold = 90.4286; % Set the frequency threshold
[pks,locs] =
findpeaks(AY_0(F_0>=frequency_threshold),F_0(F_0>=frequency_thres
hold));
fundamental_frequency = locs(1);
disp(['The fundamental frequency is: '
num2str(fundamental_frequency) ' Hz']);
```

RESULT

Output 1 -



Output 2 –


MATLAB R2023a - academic use
My_Voice_FundamentalFreq.m
Fs = 4000; Channels = 1; bits = 16;
r = audiorecorder (Fs ,bits, Channels) ;
duration = 7;
disp ('Recording started');
recordblocking (r, duration);
disp ('Recording stopped');
X = getaudiodata(r) ;
%sound (X, Fs, bits) ;
t = 0:1/Fs:(length(X)-1)/Fs;
subplot (2,1,1);
plot (t,X, 'LineWidth', 1.5) ;
xlabel ('time (sec)');
ylabel ('Amplitude');
title ('Time Domain Plot of the Recorded signal');
n = length (X) ;
F = 0: (n-1) *Fs/n;
Y = fft (X, n) ;
F_0 = (-n/2:n/2-1) .*(Fs/n) ;
Y_0 = fftshift (Y) ;
AY_0 = abs (Y_0) ;
subplot(2,1,2);
plot(F_0,AY_0,'LineWidth', 1.5);
xlabel ('Frequency (Hz) ');
ylabel ('Amplitude') ;
title ('Frequency Domain Plot of Audio Signal');
filename = 'myvoice.wav';
audiowrite(filename,X,Fs);
frequency_threshold = 90.4286; % Set the frequency threshold
[pks,locs] = findpeaks(AY_0(F_0>=frequency_threshold),F_0(F_0>=frequency_threshold));
fundamental_frequency = locs(1);
disp(['The fundamental frequency is: ' num2str(fundamental_frequency) ' Hz']);
Figure 1
Time Domain Plot of the Recorded signal
time (sec)
Amplitude
Frequency Domain Plot of Audio Signal
Frequency (Hz)
Amplitude

Advantages -

1. *The Fourier Transform is a powerful tool for analyzing signals in the frequency domain.*

2. *It is easy to implement and can be used to analyze both linear and non-linear systems.*

3. *It is computationally efficient and can be used to decompose signa Is into their constituent frequencies.*

4. *It can be used to identify and isolate specific frequencies or frequency bands in a signal.*

5. *It is widely used in applications such as digital signal processing, image processing, and communications.*

Limitations -

1. *The Fourier Transform is limited to analyzing signals that are continuous in time, so it cannot be used to analyze signals that are discrete in time.*

2. *It is sensitive to noise, so signals with high levels of noise may not be accurately represented by the Fourier Transform.*

3. *It can be computationally expensive, depending on the length of the signal and the number of frequencies that need to be analyzed.*

4. *It is not suitable for analyzing non-periodic signals, such as speech or random signals.*

CONCLUSION

The Fourier Transform is an important mathematical tool that can be used to analyze and manipulate signals.

It can be used to study the frequency content of signals, filter out unwanted frequencies, or to reconstruct signals from their frequency components.

It is a powerful tool that is used in a variety of applications.

REFERENCES

Websites:

MATLAB Website : https://in.mathworks.com/help/matlab/help-and-support.html

Yourengineer.in : https://yourengineer.in/knowledge-hub/what-is-fourier-transform-definitions-advantages-disadvantages/