MICROCONTROLLER-BASED SPECTRUM ANALYZER SYSTEM

KRITHARAN A/L RAJANDRAN

BACHELOR OF ELECTRICAL ENGINEERING (ELECTRONICS) WITH HONOURS

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : Kritharan A/L Rajandran

Date of Birth : 13th Dec 1999

Title : Microcontroller-Based Spectrum Analyzer System

Academic Session : 20/21 Semester II

I declare that this thesis is classified as:

- ☐ CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*
- ☐ RESTRICTED (Contains restricted information as specified by the organization where research was done)*
- ☑ OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

R. Kritharan.	
(Student's Signature)	(Supervisor's Signature)
991213055487	Izzeldin Ibrahim Mohamed
New IC/Passport Number	Name of Supervisor
Date: 15/06/2022	Date: 26/06/202

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

Universiti Malaysia PAHANG

Engineering • Technology • Creativity

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Electrical Engineering (Electronics) with Honours.

DR. IZZELDIN IBRAHIM MOHAMED ABDELAZIZ
SENIOR LECTURER
DEPARTMENT OF ELECTRICAL ENGINEERING
COLLEGE OF ENGINEERING
UNIVERSITI MALAYSIA PAHANG
LEBUHRAYA TUN RAZAK
26300 GAMBANG, KUANTAN, PAHANG
TEL: +609-424 6145 FAX: +09-549 2689

(Supervisor's Signature)

Full Name : Dr. Izzeldin Ibrahim Mohamed Abdelaziz
Position : Senior Lecturer
Date : 26/06/2022

Universiti Malaysia PAHANG

Engineering • Technology • Creativity

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

R. Kitharan.

____

(Student's Signature)

Full Name: Kritharan A/L Rajandran

ID Number: EA18098

Date : 15/06/2022

MICROCONTROLLER-BASED SPECTRUM ANALYZER SYSTEM

KRITHARAN A/L RAJANDRAN

Thesis submitted in fulfillment of the requirements
for the award of the
Bachelor of Electrical Engineering (Electronics) with
Honours

College of Engineering
UNIVERSITI MALAYSIA PAHANG

JUNE 2022

ACKNOWLEDGEMENTS

First and foremost, I would like to show my gratitude and appreciation to University Malaysia Pahang for giving me the opportunity to continue my studies in to enhance my knowledge, experience and working skills throughout the completion of my final year project.

In addition, I want to thank my supervisor, Dr. Izzeldin Ibrahim Mohamed Abdelaziz, for his guide from the beginning of this process and shared so many helpful information to us to complete this project. Many facets of this project have been investigated under supervision, and with his expertise, ideas and assistance, this thesis completed in the allocated time.

Throughout my project, my parents and friend have been unwavering in their support and encouragement. Lastly, I would like to thank everyone who contributed to my research and project completion, whether directly or indirectly. This opportunity represents a significant step forward in my professional development.

ABSTRAK

Peranan penganalisa spektrum menjadi sangat penting bagi editor lagu dan jurutera audio untuk menilai kerja mereka; namun, ternyata penganalisa spektrum untuk keperluan ini masih sukar ditemui, secara umumnya hanya wujud untuk keperluan dengan julat frekuensi terhad untuk keperluan lain selain keperluan pembetulan lagu yang memerlukan spektrum isyarat audio 32 jalur. Kertas kerja ini memperkenalkan reka bentuk penganalisa spektrum audio yang cekap dan kos rendah menggunakan Arduino untuk menganalisis spektrum isyarat dalam audio 32-jalur. Penganalisa spektrum untuk spektrum isyarat audio 32-jalur umumnya disepadukan dalam set peranti sistem audio yang agak mahal dan tidak dipisahkan, walaupun tidak semua ciri penganalisa audio digunakan. Fast Fourier Transform (FFT) digunakan dalam penganalisa spektrum ini. Sistem penganalisa spektrum audio telah direka bentuk di sekeliling mikrocontroller (Arduino MEGA Board) yang tersedia dengan ciri Analog to Digital Converter (ADC) yang digunakan untuk menukar isyarat input audio kepada sampel digital. ADC dikonfigurasikan untuk mencuba isyarat input dengan frekuensi jam 38.46 kHz. Ini dicapai dengan mengkonfigurasi pra-skala ADC kepada 32. Kekerapan pensampelan 38.64 kHz bermakna sampel digital boleh menghasilkan semula frekuensi input sehingga 19.32 kHz (teorem Nyquist) yang sudah cukup baik untuk isyarat audio. Sistem mengumpul isyarat audio dan analisis audio direka berdasarkan skema ini. Perisian ini telah dibangunkan menggunakan MATLAB di mana analisis keputusan akan dipaparkan. Analisis hasil daripada penganalisa spektrum audio boleh digunakan sebagai pembetulan audio atau muzik, bar grafik audio 32-jalur akan menunjukkan jika isyarat audio berjalan dengan betul.

ABSTRACT

The role of the spectrum analyzer becomes very important for song editors and audio engineers to assess their work; however, it turns out that spectrum analyzer for these needs is still difficult to find, which generally exists only for needs with a limited frequency range for other needs besides song correction needs that require a 32-band audio signal spectrum. This paper introduces a design of efficient and low-cost audio spectrum analyzer using Arduino board to analyze signal spectrum in 32-band audio. Spectrum analyzer for 32-band audio signal spectrum generally integrated in a set of audio system devices that are relatively expensive and not separated, even though not all features of the audio analyzer are used. Fast Fourier Transform (FFT) implemented in this spectrum analyzer. The audio spectrum analyzer system has been designed around the microcontroller (Arduino MEGA Board) which is available with an Analog to Digital Converter (ADC) feature that is used to convert audio input signals into digital samples. The ADC is configured to sample the input signal with clock frequency of 38.46 kHz. This achieved by configuring the ADC pre-scaler to 32. Sampling frequency of 38.64 kHz means that digital samples can reproduce input frequencies up to 19.32 kHz (Nyquist theorem) which is good enough for audio signals. The system collects audio signal and audio analysis is designed based on this scheme. The software has been developed using MATLAB where the result analysis will be displayed. The result analysis from the audio spectrum analyzer can be used as audio or music correction, the 32-band audio graphic bar will show if the audio signal run properly.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

LED	Light Emitting Diode
FFT	Fast Fourier Transform
IDE	Integrated Drive Electronics
ADC	Analog-to-Digital Converter
PWM	Pulse Width Modulation
USB	Universal Serial Bus
DFT	Discrete Fourier Transform
DTS	Discrete Time Signal
PSD	Power Spectrum Density
RDS	Radio Broadcast Data Service
GUI	Guide User Interface
RTA	Real-Time Analyzer
DAQ	Data Acquisition
CPU	Central Processing Unit
UART	Universal Asynchronous Receiver/Transmitter
GLCD	Graphical Liquid Crystal Display
IIR	Infinite Impulse Response

CHAPTER 1

INTRODUCTION

1.1 Project Background

A spectrum analyzer is an electronic device used to measure the magnitude of the amplitude of the input signal displayed in the form of a frequency spectrum that can be seen visually. Frequency spectrum is the arrangement of the frequency bands in units of vibration of electromagnetic waves that propagates in air space. The results of the frequency spectrum can be used to analyze signal patterns, so that the types of signals can be distinguished according to the frequency range that was successfully received.

At the present time, there have been various kinds of analytical apparatus in market available, however these instruments also have shortcomings. From the results of the previous research, the design of the spectrum analyzers has been hit with costs and only certain people can use it. Related to the research that has been done, this thesis made more specific as an alternative to spectrum analyzer design for audio signal analyzing. With the aim of spectrum analyzer can be utilized by everyone because it is easy to use and made with relatively very low cost. In addition, the result of the spectrum analyzer product design can be used as a basis for the development of similar products or other research.

The audio editors will analyze the spectrum produced by the audio signal which is displayed by the spectrum analzyer visually. Then, it can be seen which frequencies have low, medium and high amplitude, after that is corrected using an equalizer. Clearly above, the equalizer will not be much of a use in audio correction if it is not equipped with spectrum analyzer. So, the role of the spectrum analyzer becomes very important for editors and audio engineers, however it turn out that spectrum analyzer for these needs is still difficult to find. Therefore, it is necessary to design a spectrum analyzer in 32-

band audio and the software implementation to generate function from the audio spectrum analyzer.

1.2 Problem Statement

This project is to design an audio spectrum analyzer. An audio spectrum analyzer should be able to measure the magnitude of the amplitude of input signal displayed in the form of frequency spectrum that can be seen visually. The results can be further used to analyze signal patterns. However, it turns out that spectrum analyzer for the needs of real time audio analyzing and audio correction is still difficult to find which generally exist only for needs with a limited frequency range. Besides, at the present time the design of spectrum analyzers have been hit with costs and only certain people can use it.

Therefore, it is necessary to design a low-cost audio spectrum analyzer in 32-band audio and software implementation to generate function from the audio spectrum analyzer. Thus, one need to know which microcontroller to be used so that it allows the spectrum analyzer works well and can be utilized by everyone because it is easy to use and made with relatively low cost. In addition, for spectrum analyzer 32-band audio signal spectrum usually integrated with audio system devices that are relatively expensive and not available separately, even though not all features are useful.

1.3 Objective

The aim of this research is to design a microcontroller-based spectrum analyzer system. In order to do so, a few objectives are constructed to achieve its target. Among the objectives are:

i. To develop an efficient and low-cost 32-band Audio Spectrum Analyzer system.

ii. Multifunctional that is capable of real time speech signal acquisition, recording, playback of speech.

iii. Able to develop spectrum visualization for spectrum analysis on PC using MATLAB

1.4 Scope of the study

There are several parts that needed to be done for this project. First and foremost, is hardware set up of the audio spectrum analyzer. Next, software interfacing on PC for spectrum visualization. Lastly, on need to implement FFT, testing the system for noise filtering.

For hardware set up, on can refer the data sheet of the microcontrollers from arduino.cc which is an open-source electronic prototyping platform site. Arduino IDE platform is used throughout the project for audio spectrum system development. This audio spectrum analyzer design only uses Arduino Technology for hardware. This is needed to programme the microcontroller used which is the Arduino UNO board that act as the electronic control system.

Next, the capturing speech signal for spectrum visualization. For this part, one need to know how to create graphical interfaces for collecting and analyzing data and visualizing the spectrum results on PC using MATLAB which is also used throughout the project.

Finally, implementing FFT algorithm, testing it where the audio spectrum analyzer system collect data from real-time speech signal, recording and playback of speech while visualizing the spectrum on the PC and in the hardware. The audio signal input used in this study came from speech signal, not from other audio system devices. The 32-band audio spectrum results will display visually using LED display for hardware and MATLAB for software, not in other visualizer devices.

1.5 Project Outline

This paper consists of 5 chapters which are chapter 1 until chapter 5. The thesis focuses on the discussion and progress of development and design that be carried out. The construction of all chapters is as follow

Firstly, chapter 1 represent the Project Background, Problem Statement, Objectives, and Project Scope will be analysed in this chapter. In this chapter it will explain the introduction of this project.

Chapter 2 is viewing the literature reviews of this project based on some journals, research papers and various references. In this scope it will explain about the technique that has been use and the achievement by every research paper.

Chapter 3 reviews the techniques that have been used to run the project and all method used. This chapter includes two sections which are the development of hardware and software. In addition, the methods and techniques used will also be described in detail in this project, including the method and steps required.

Chapter 4 presents the results acquired after the experiment is done. The results are collected and recorded. The results and analysis are recorded by using graph, figure and table.

Chapter 5 will talk about the conclusion for the whole project, recommendation for future study and references are also included in this chapter.

CHAPTER 2

LITERATURE REVIEW

2.1 Spectrum analyzer and Audio spectrum

Various types of spectrum analyzers have been created during the last few decades for a variety of applications in various fields. A spectrum analyser, according to Thomas, S., and Haider, N.S. (2013), is a device that evaluates the magnitude of an input signal vs frequency within the instrument's whole frequency range [1]. The most common application is to determine the strength of a spectrum of known and unknown signals. Analysis of a signal can simply be defined as checking information signal in the frequency domain and time domain. Signal analysis in the time domain is done by means of an oscilloscope, while signal analysis in the frequency domain is done by using a spectrum analyzer. There are three types of spectrum analyser which are super heterodyne type, real time spectrum analyser and FFT spectrum analyzer.

The range of audible frequencies that humans can hear is from 20 Hz to 20000 Hz. The audio spectrum ranges from 20 Hz to 20000 Hz, and it may be efficiently divided into seven frequency bands, each of which has a particular impact on the overall sound. In recent years, some spectrum analyzers have been developed, including hardware derived from widely used microcomputers and software based on user self-programming. [2] Despite the extensive study that has been conducted and published in the literature, as well as in design, architecture and control, there are still limitation or improvements to be made on the audio spectrum analyzer. It is necessary to design a spectrum analyzer product for signal spectrum analyzer in 32-band audio.

2.2 Microcontroller

The number of applications developed with microcontrollers has increased rapidly in recent years. A microcontroller is an integrated circuit that is housed within each component that is required to do specified operations and that can perform a certain task on a regular basis without the need for another boom. This microcontroller contains a microprocessor, analog-to-digital conversion (ADC), memory units and input-output interfaces, pulse width modulation (PWM) and communication modules and various control. [3] For beginners it may confuse them to decide which microcontroller to be used when developing or implement it in an application. While selecting the appropriate microcontroller aids the user in the later stages of implementation, selecting the incorrect microcontroller may result in the application being halted at some time.

It is necessary to consider several features when choosing a right microcontroller development card such as card interface, intrinsic hardware, program development interface, operating voltage, input and output numbers as well as processor power and capacity. Furthermore, to assure that the additional equipment supporting the development is both hardware and software compatible care must be taken just as what Yilmaz Guven (Yilmazlar, 2017) stated in their paper entitled “Understanding the concept of microcontroller-based systems to choose the best hardware for applications”.

2.2.1 Arduino microcontroller

Arduino is a free and open-source platform for developing and programming electronics. It can receive and transfer data to most devices, as well as deliver commands to specific electronic devices over the internet. In today's world, Arduino is widely utilised in microcontroller programming for a variety of reasons, including its user-friendly or easy-to-use configuration. An Arduino, like any other microcontroller, is a circuit board with a chip that can be programmed to perform a variety of activities. It transfers data from a computer programme to the Arduino microcontroller, which then sends it to a specific circuit or machine with many circuits to carry out the command. You may use an Arduino to read data from other input devices. In their paper, Badamasi, Y.A. (2014) stated that, unlike most previous programmable circuit boards, the Arduino does not require a separate piece of hardware to load new code onto the board; instead, you can simply use a USB cable to upload, and the Arduino software uses a simplified version of C++, making it easier to learn to code and providing you with a more accessible environment that bypasses the microcontroller's functions.

Many different types of built-in modules are available for Arduino boards. For wireless connection, boards like the Arduino BT have a built-in Bluetooth module. These built-in modules can also be purchased separately and interfaced (mounted) to the device. [6] These modules are known as Shield.

Figure 2.1 Types of Arduino boards

Arduino Type	Microcontroller	Clock Speed
Arduino Uno	ATmega328	16 MHz with auto-reset
Arduino Duemilanove / ATmega328	ATmega328	16 MHz with auto-reset
Arduino Nano	ATmega328	16 MHz with auto-reset
Arduino Mega 2560 or Mega ADK	ATmega2560	16 MHz with auto-reset
Arduino Leonardo	ATmega32u4	16 MHz with auto-reset
Arduino Mini w/ ATmega328	ATmega328	16 MHz with auto-reset
Arduino Ethernet	Equivalent to Arduino UNO with an Ethernet shield	
Arduino Fio.	ATmega328	8 MHz with auto-reset
Arduino BT w/ ATmega328	ATmega328	16 MHz with auto-reset
LilyPad Arduino w/ ATmega328	ATmega328	8 MHz (3.3V) with auto-reset
Arduino Pro or Pro Mini	ATmega328	16 MHz with auto-reset
Arduino NG	ATmega8	16 MHz with auto-reset

Table 2.1 List of different types of Arduino board with microcontroller type

2.3 Implementation of Fast Fourier Transform (FFT) Algorithm

In the study of audio and acoustics measurement, the Fast Fourier Transform (FFT) is a significant measurement method. It breaks down a signal into its constituent spectral components and hence offers frequency information. FFTs are employed in machine or system defect analysis, quality control, and condition monitoring. The Discrete Fourier Transform (DFT) is implemented using an improved technique called FFT (DFT). A signal is sampled and separated into its frequency components over a period of time. Single sinusoidal oscillations at different frequencies, each with its own amplitude and phase, make up these components.

The FFT method of computing the Discrete Fourier Transform is quick and efficient (DFT). [7] In the world of digital signal processing, FFT is among the most perfect and widely used operations. In their study, (Mehrotra, 2017) claims that the results of FFT are the same as those of DFT. The only difference is that the algorithm is optimised to eliminate redundant calculations, which means that the FFT reduces the number of calculations required and hence the computation time. The FFT decomposes the set of data to be changed into a sequence of smaller data sets to be altered in terms of functionality. The smaller sets are then decomposed into even smaller groups. Each stage of processing combines the results of the previous stage in a unique way.

2.4 Journal Review / Related works

An early low-cost spectrum analyzer designed to monitor FM and AM signals is presented in [8]. In this research from (E. Santos-Luna, 2019), clarified that the spectrum analyzer can be developed using low-cost microcontroller. The author shows how to make a low-cost spectrum analyzer using a Raspberry Pi3 B+ board and Python code to process the spectrum. The author proposes a system that includes three key features of wireless communications, including a general signal analyzer, as well as FM and AM signal analyzers. The associated application can display spectrograms, discrete time signals (DTS), power spectrum density (PSD) signals, and radio broadcast data service (RDS) signals, as well as record received signals. The resulting signals can be seen in the frequency or time domain, and they can also be recorded.

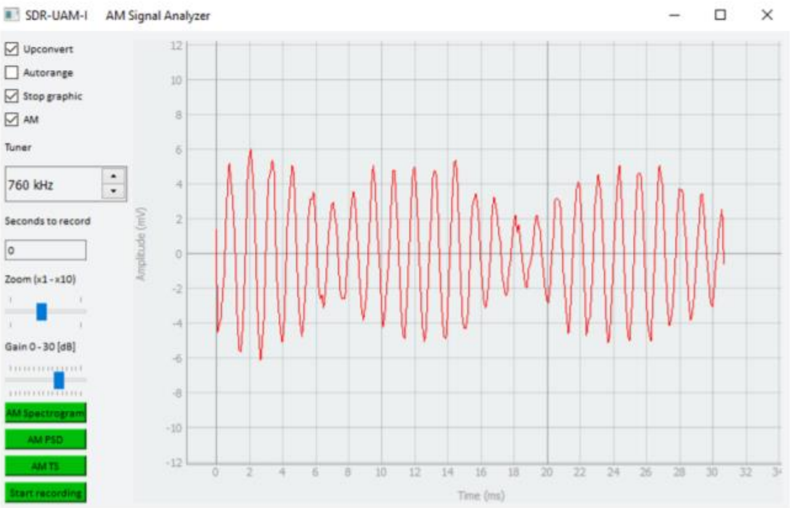

Figure 2.2 Spectrum result in time domain

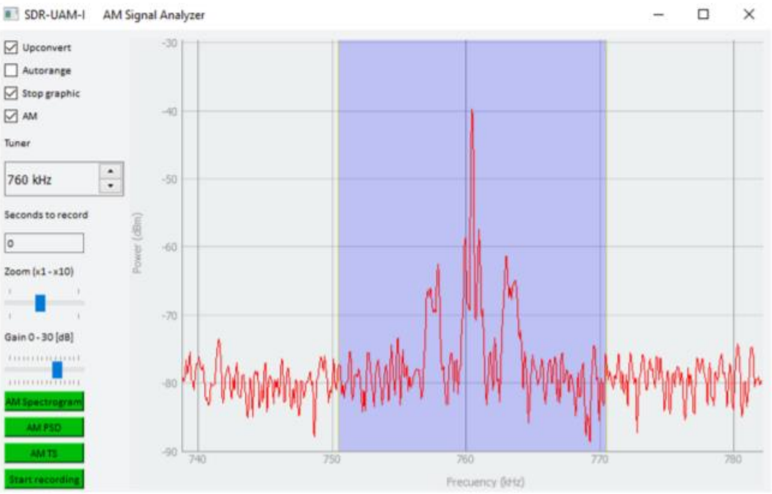

Figure 2.3 Spectrum result in frequency domain

Another spectrum analyzer option is introduced in [9], this time for real-time audio analysis of loudspeakers. Raharjo, B. J., and Zakaria, M. (2019) provide an Audio Spectrum Analyzer application that allows users to see the loudspeaker's frequency response in real time. The Fast Fourier Transform (FFT) function, which calculates the amplitude value of the acquired audio sound, is the first step in this application. The Fast Fourier Transform Algorithm approach will be utilised in the software to convert data efficiently to the frequency domain [7]. Additionally, the equalizer and normalization processes should be used to determine the highest amplitude value of each spectrum so that it may be inspected easily. MATLAB Guide User Interface (GUI) feature was used to create this spectrum analyzer application.

Figure 2.4 The implementation of the whole system with loudspeaker

The inputs are processed as sound signals, which are recorded by an RTA microphone, processed by an external soundcard, and then presented in the software. The audio spectrum analyzer took audio signals as input and produced a real-time frequency spectrum as output. This device's design included both hardware and software components. A Real-Time Analyzer (RTA) microphone, an external soundcard, and a loudspeaker as the audio source to be processed were included in the design of the hardware needed to integrate with the software. The software was created using Matlab's Guide User Interface (GUI) feature and the Fast Fourier Transform (FFT) technique to transform the signal into a spectrum.

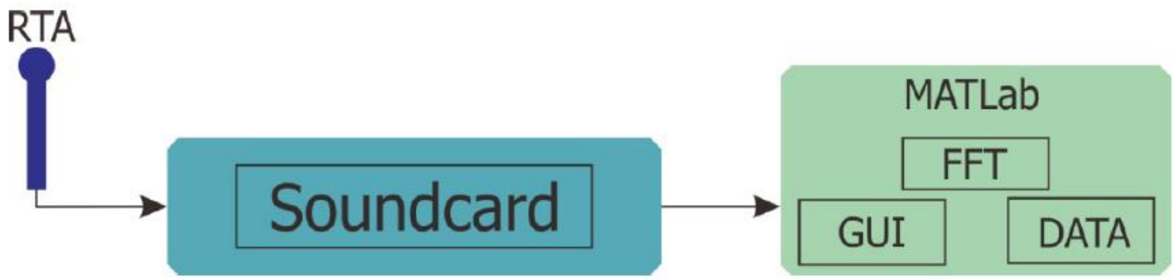

Figure 2.5 System Block Diagram

The main interface of the program interface designed would display several frequency spectrums with additional features. It may be deduced from the test findings in this study and a comparison of the Audio Spectrum Analyzer applications that this device can also indicate the precise position of the spectrum.

A proposal similar to the one presented here appears in [10]. In this paper, author present a design and development of a FFT spectrum analyzer on PC using Lab View and microcontroller-based data acquisition (DAQ) system is discussed. The FFT spectrum analyzer that implemented by interfacing a microcontroller based DAQ system to PC through parallel port. It has been designed around microcontroller AT89C51 and analog to digital converter ADC0804 and the software developed using Lab VIEW. The microcontroller is used to control the operation of amplifier and the ADC in the DAQ system. It is also used to read the converted data from the ADC and to transfer to PC. AT89C51 is and 8-bit microcontroller that operates at 11.0592 MHz clock frequency. In this design the author also includes a low pass filter at input stage using UA 741 op-amp to prevent aliasing.

Program for transfer of sampled data from microcontroller to computer memory has been developed in LabVIEW. FFT on the sampled data is computed using LabVIEW built-in function. The spectrum analyzer has been tested by applying square wave of known amplitude, duty cycle and frequency and sine wave known as frequency. The output which is the FFT spectrum plotted and computed using MATLAB. The limitation of this spectrum analyzer is the maximum frequency of signal that can be analyzed is limited by the conversion time of ADC in the system. Besides, it also restricts the maximum number of samples that can be collected and analyzed which is turn limits the resolution of the spectrum.

However, the designing for spectrum analyzer for use of audio signal analyzing need to consider and studied first. In that case the recent works has been gone through to find out its design and how it works. In recent years, some virtual signal analyzers based on PC have been developed, with hardware derived from widely used microcomputers and software based on user self-programming. Duan and Zhao (2013) introduce the construction of an audio analyzer employing the Fourier Transform (FFT) algorithm in line with the ARM microprocessor [2]. The authors proposed a strategy for audio analysis over an audio stream using the ARM7 and UDA1341TS processors, utilising double cache and DMA technologies to boost data processing performance. This audio analyzer uses the C programming language in conjunction with the assembly language. In audio device drivers, the method of Double Buffering is well used to process a large volume of audio data.

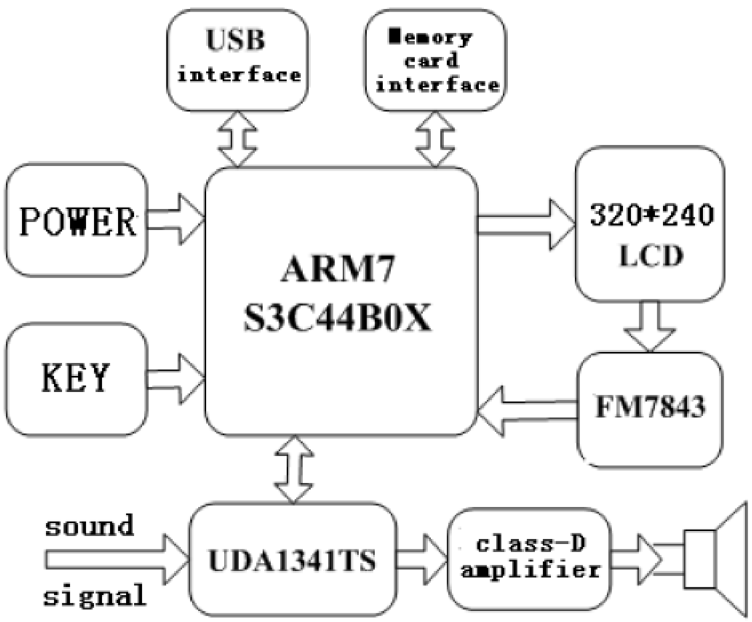

Figure 2.6 The structure of the hardware

The audio analyzer designed by the authors consist of signal acquiring, signal processing, result displaying and control segment. Power section, central processing unit (CPU), input, liquid crystal display, touch screen, USB interface, memory card interface, and audio acquiring processor form the system's hardware. The touch screen transforming interface chip is used to control touch screen, which is 12-bit sampling ADC with synchronous serial interface. The spectrum analyzer has been tested on 440 Hz International Standard Audio and 880 Hz signal.

In evaluating the frequency spectrum, this audio analyzer was shown to have high precision and little errors. The results were plotted using the signal's frequency structure to determine the components of amplitude and phase position, as well as to create various spectrums with frequency as the x-axis. With a friendly interface, the author mention that this audio analyzer is easy to operate and low-cost. However, if a smaller FFT sample is used for assessing changes, this design has a limitation: larger analysis errors will occur.

It is also necessary one need to consider the cost of the audio spectrum analyzer. Recent work appears in [11], explains the background essential for audio spectrum analyzer designing for relatively low cost. Chaudhari and Kulkarni (2016) have used the 32-bit ARM7TDMI microcontroller and developed the system by implementing FFT algorithm. The LPC2148 in this design will sample the audio signal at a constant rate and convert it to the frequency domain using a Fast Fourier Transform (FFT). The frequency domain data is then sent to the graphical LCD (GLCD), which converts the data into a real-time histogram representation. The software portion consists of FFT MCU code. The ADC sampling of the audio signal, FFT frequency conversion, and UART transmission of the frequency data are all performed by the FFT MCU code.

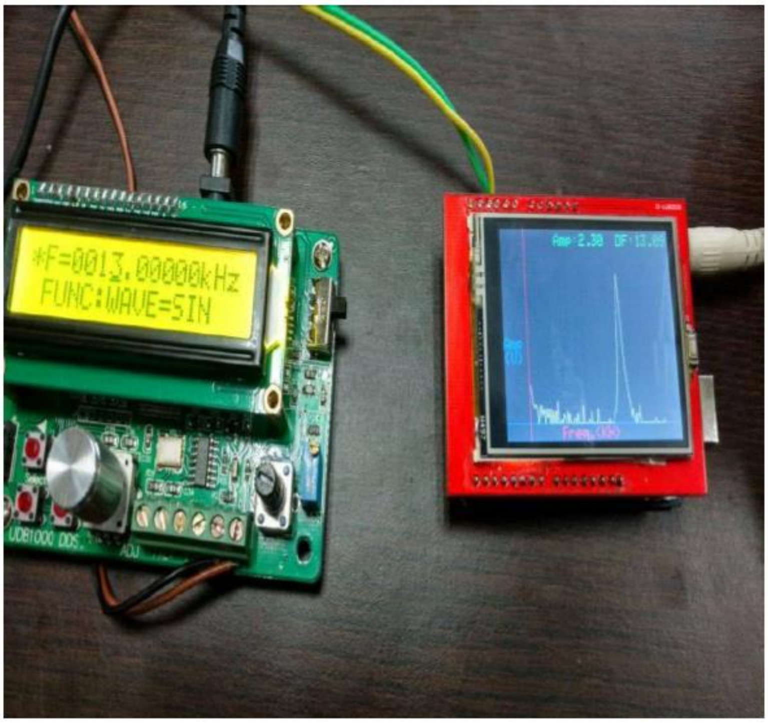

Figure 2.7 The FFT spectrum output display unit

The authors utilized Keil Vision to build and develop the code and program processor for the software configuration. It must ensure that the sample is sent to the ADC port at properly spaced intervals in order to obtain accurate results. For displaying the output for the computed spectrum, a GLCD was used in this design. More frequency bins and a more precise frequency representation of the audio signal are usually beneficial. By using standard 3.5mm audio jacks or small mice as analog to digital transducer along with very few components the author has built a low cost, very economical audio spectrum analyzer and fully functional system which operates in real time mode and accurately displayed the frequency content by visualizing it.

In the other hand, the audio signal information is also important to be able to be analyze. A proposal similar to the one presented here appears in [12]. In this paper, the system collects and processed speech signals through computer soundcard, LabView and MATLAB. This system combines LabView and MATLAB to complete the basic research and operates simply. This proposed system is a part of human-computer interaction to achieve the objectives which are recording, playback, infection play, time domain and frequency domain analysis of the audio signal. Analysis and processing by FFT, noise and filtering of speech signal can be done by MATLAB.

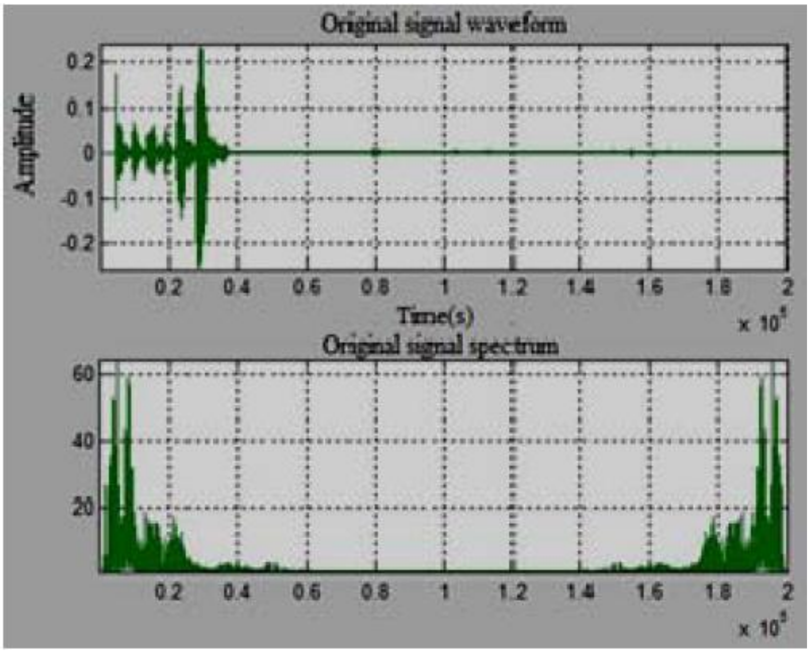

Figure 2.8 Signal waveform and spectrum of input audio signal

In this speech signal analysis work, it mainly includes time domain and frequency domain spectrum analysis for noise reduction and filtering. Using wavread function, the speech signal can be sampled, getting the sampling frequency and points of the signal in MATLAB software programming platform. It is very important for high fidelity signal processing such as speech processing, data processing and testing. At the time of collecting audio signal, inevitably there will be some noise signal input. Finite Impulse Response (FIR) filter is implemented and adopted to filter the audio signal in this paper.

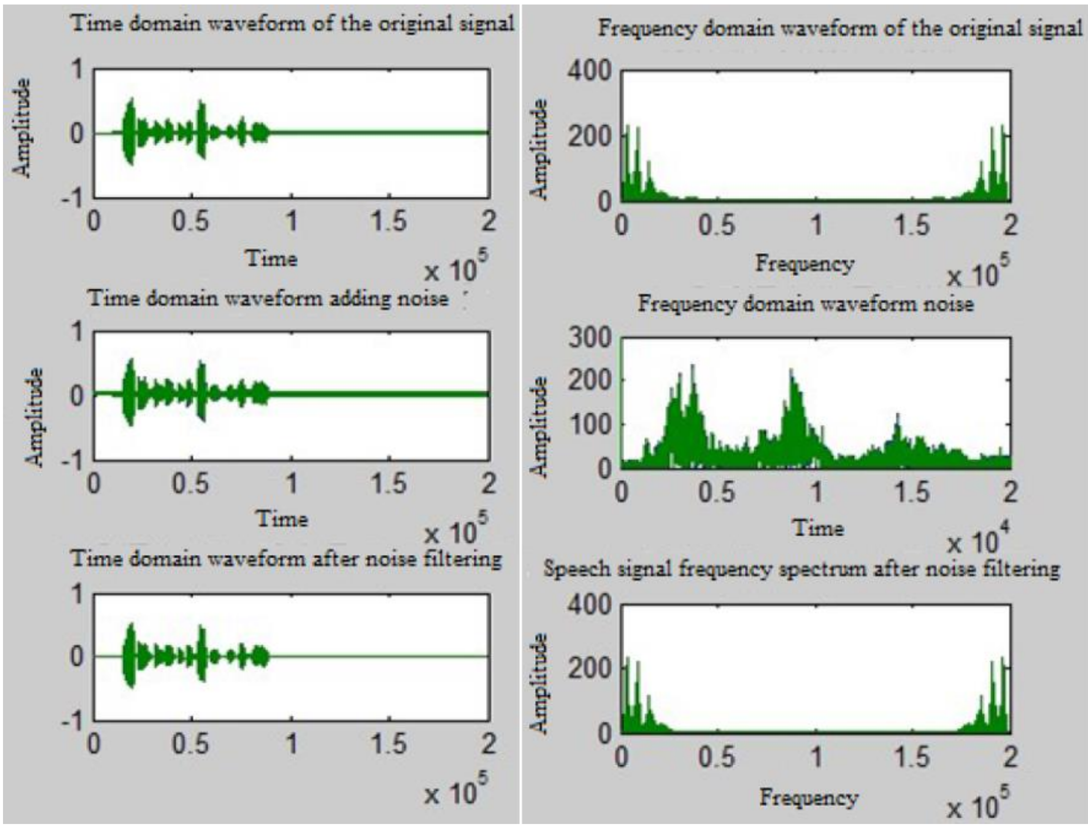

Figure 2.9 The comparison result for original and filtered signal

From this related works, it is possible that real time speech signal acquisition, recording and playback of speech as well as able to visualize the spectrum on PC either using LabView or MATLAB. This could be done in low-cost basis with a good efficient of the audio spectrum analyzer system. At last filter processing of noise signal can also be consider as mentioned in previous researches to filter the unwanted signal and reached good result.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, the method and technique that will be used to accomplish the project's objectives and the factors that were taken into consideration for this initiative will be discussed. It starts with a discussion of the project flow. Then, moves on the system design process, as well as the methods and software used in this project. An efficient, low-cost, and good material selection is critical to a project's progress and perfection. In order to create well-functioning audio spectrum analyzer system, it is critical to choose the most suitable components with the right components with the right requirements. This concept can be applied to both hardware and software development.

3.2 Development Process

The aim of this project is to develop an efficient and low-cost 32-band microcontroller-based audio spectrum analyzer system. To do so, the most critical disciplined need to look is the techniques and technological solutions suggested. As a result, a simpler phase by phase approach has been proposed. Problems can be observed early on with this approach, avoiding stressful failures. There is still an aim to achieve, whether it is short-term or long-term. It is more organized to complete the tasks one at a time, according to their stages. As a result, the audio spectrum analyzer system development includes 3 stages. Electronic system development, software development, implementing FFT algorithm, testing the system for noise filtering are among them.

3.3 Electronic System Development

For the audio spectrum analyzer, the hardware of the system involves electronic components. The hardware of the system is composed of power section, a microcontroller, voice recorder module, dot matrix LED display module, speaker, breadboard, solder cables and jumper wires. The design of the hardware used to be affordable and relatively low-cost. In that case, the electronics component decided in developing the hardware was a critical factor. At the same time, the efficiency of this audio spectrum analyzer need to achieve in order to fulfil the objectives in this research.

Components	Price (RM)
Arduino UNO	47.90
ISD 1820 voice recorder module	12.39
Breadboard	3.90
Speaker 3 W	7.50
Jumper wires	5.20
Solder cable	16.86
MAX7219 Dot Matrix Module	16.50
	Total : RM 110.25

Table 3.1 The components used for this project and its cost

3.3.1 Components

For this project, Arduino UNO Board is used as microcontroller for programming and control the operation of the system. It is a microcontroller board which is based on the ATmega328P. It has 5V operating voltage, 14 digital inputs or outputs pins (6 ports can be used as PWM outputs) and 6 analog input pins, a 16 MHz ceramic resonator, a USB connection, power jack, an ICSP header and a reset button.

This Arduino UNO Board is available with an Analog-to-Digital Converter (ADC) feature that will be used to convert audio input signals into digital samples. The ADC is configured to sample the input signal with a clock frequency of 38.46 kHz. This is achieved by configuring the ADC pre-scalar to 32. Sampling frequency of 38.46 kHz means that digital samples can reproduce input frequencies up to 19.32 kHz (Nyquist theorem). It will be good enough for audio signals processing.

Figure 3.1 The Arduino UNO Board

A ISD 1820 voice recorder module is used for recording the human voice or to receive the audio wave signal emitted from real time speech signal. This module is a multiple-message record and playback device. It detects the intensity of the audio signal where the sound is detected via a microphone and the signals transferred to the on-chip preamplifier. This on-chip Automatic Gain Control (AGC) circuit controls the gain of the preamplifier. This microphone captures sound intensity at certain frequencies and is only limited to its magnitude at the microphone location point.

This ISD 1820 voice recorder module will operate by push-button interface. The module will starts recording whenever the REC button is HIGH and stops when it is LOW. It will play the recording whenever the playback button is HIGH.

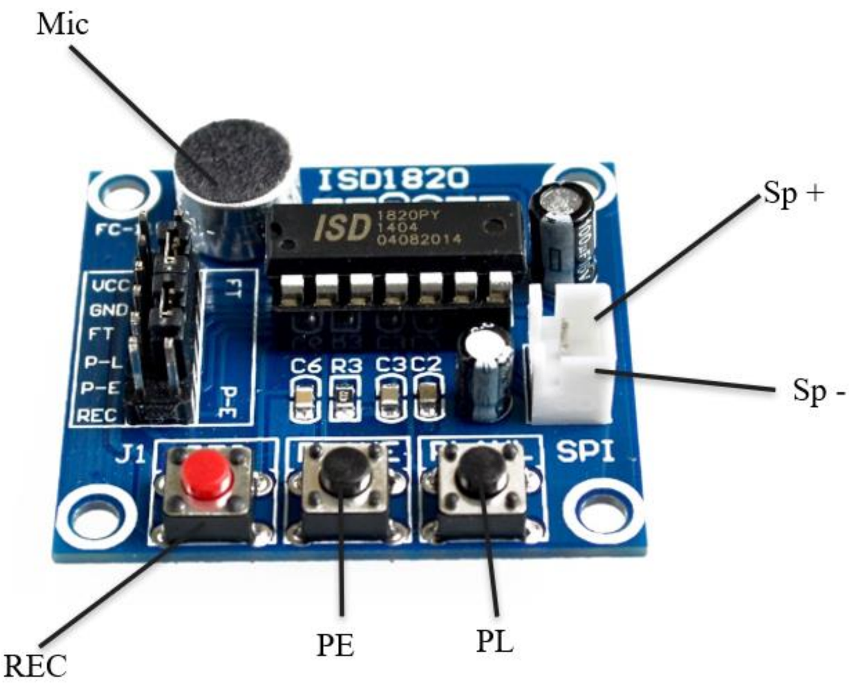

Figure 3.2 ISD 1820 voice recorder module

For the output of the system, a 32-band MAX7219 dot matrix module LED display and a speaker is needed. The 32-band dot matrix LED display will be used by the Arduino UNO as an output to display the mapped signal into 32-band spectrum results that have been processed by FFT and the results of the previous analog-to-digital converter outputs. Testing needs to be done so that it looks reliable from time to time giving the appropriate appearance. While the speaker is required to playback the speech that was recorded by the microphone. Breadboard is used to build the electronic circuit of the audio spectrum analyzer. Soldering cables is needed to connect speaker and the voice recorder module together. To connect the electronic components jumper wires is used.

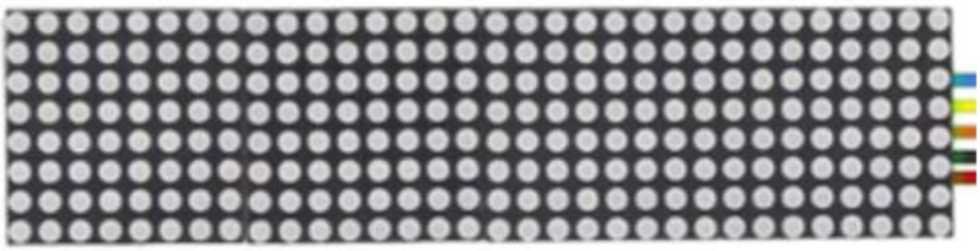

Figure 3.3 MAX7219 dot matrix module LED display

Figure 3.4 Speaker

3.4 Software Development

The code for the hardware can be developed using C programming language is employed in this audio spectrum analyzer. For this purpose, the code can be developed using the open-source Arduino IDE which make it easy to write and upload it to the Arduino UNO board. The most important part in this code development is the FFT library which ease the code development.

The Arduino FFT library is the main core of the code that translates analog input signals into the frequency spectrum. Arduino FFT can be easily used and produces the best and most accurate output for this research. The Arduino FFT library can do FFT samples between 16 to 128 samples. This can be configured in the Arduino program to this audio spectrum analyzer design.

Whereas the software design of the audio spectrum analyzer system was developed using MATLAB utilizing the programming platform by applying the Fast Fourier Transform (FFT) algorithm to convert the signal into a spectrum. This system is a part of human-computer interaction to achieve the objectives which are recording, playback of audio and spectrum visualization of the audio signal for further audio analysis.

3.5 FFT Algorithm and Filter Implementation

Fast Fourier Transform (FFT) algorithm was implemented to convert the audio signal recorded into the frequency spectrum which will be the output. This FFT algorithm was applied in the coding developed for hardware and software. For the hardware, to obtain the frequency spectrum FFT function was used. The ArduinoFFT library does the translation of input analog signal into frequency spectrum. While for the software part frequency domain plot can be obtained by using MATLAB FFT function. This method was used since it is efficient and fast way to convert signals to frequency domain by calculating the amplitude value of the recorded audio.

Speech signal recorded alongside with other unwanted signal which comes from the background noise while the recording was done. In this case, filter implementation will be the best way to rid of the background noise so that the audio quality gets better. Filters have the ability to transform the tone of the recorded audio completely and are most commonly used to remove unwanted frequency.

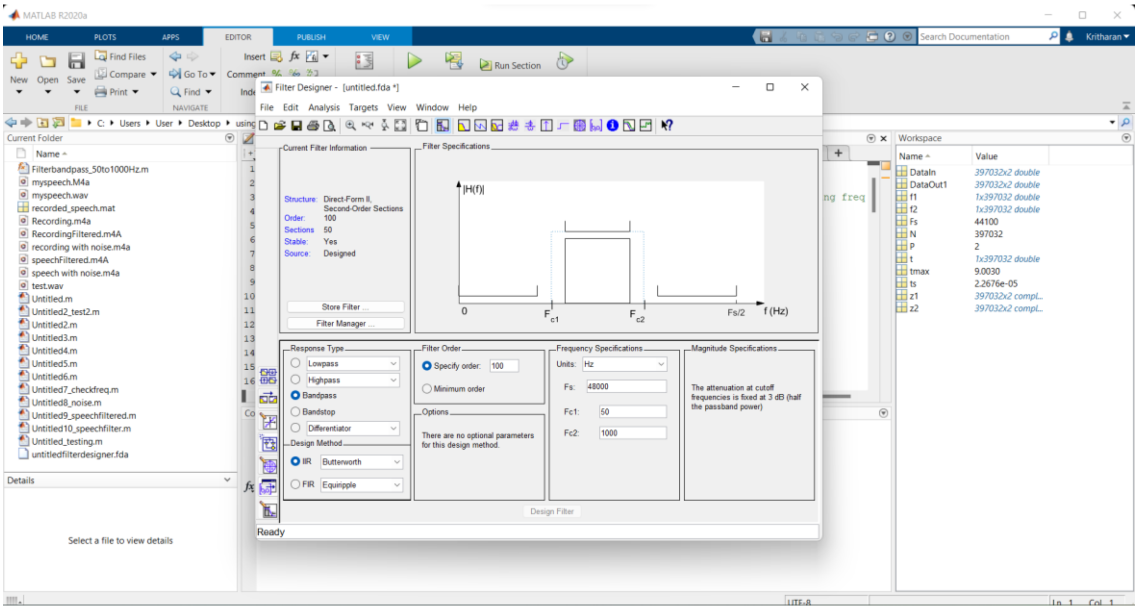

Figure 3.5 Filter specification and parameter

Filter designer command can be used to design filter that is suitable for the recorded speech signal. In the figure above, bandpass filter was designed to filter out the unwanted signal in the recorded speech. The specification and parameters of the filter such as sampling frequency, cut-off frequencies and filter order were specified. Bandpass filter type will be suitable since speech signal lies in a certain frequency range which can be specified easily from the frequency spectrum generated earlier. The sampling frequency of the filter will be same as the sampling frequency of recorded speech signal. Higher filter order can be used to get sharp transition edge to remove noise which lies within the transition edge.

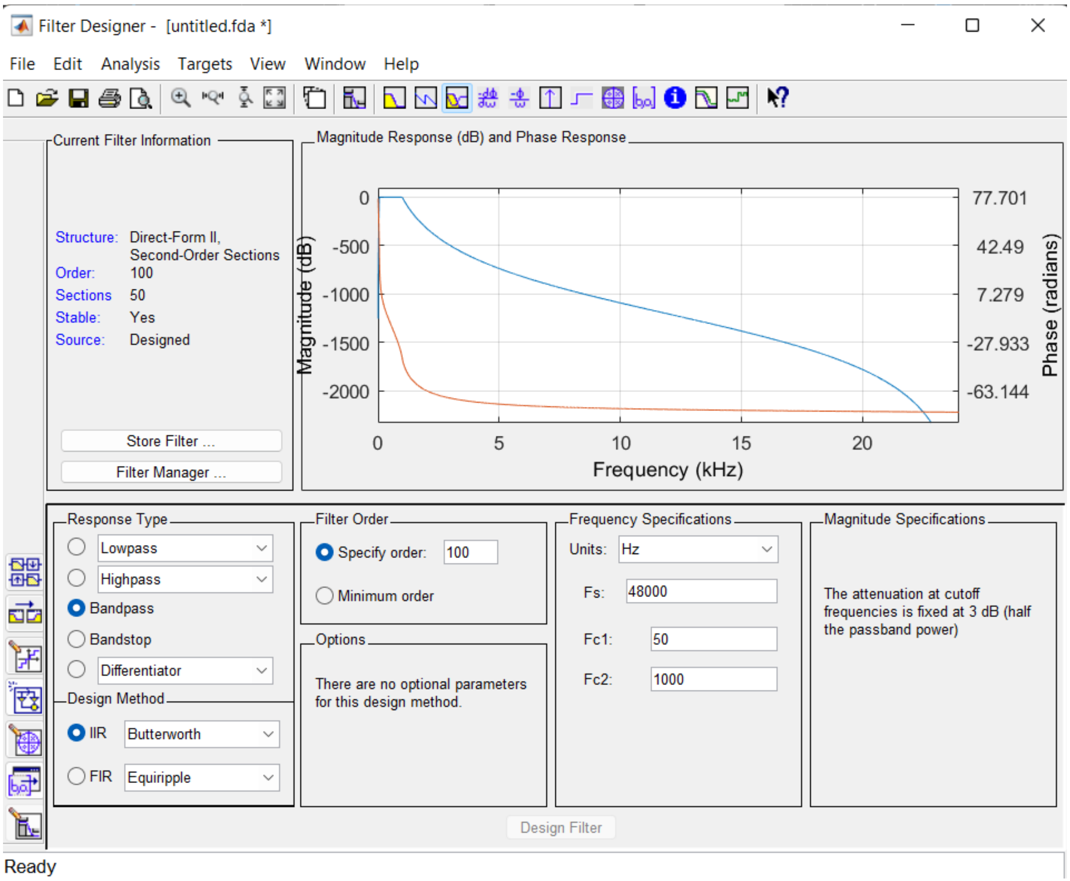

Figure 3.6 Magnitude Response and Phase Response of the designed filter

The MATLAB code of the filter can be generated after the specification and parameters were defined. This code can be combined together with the main coding so that it able to work as complete audio spectrum anlayzer system.

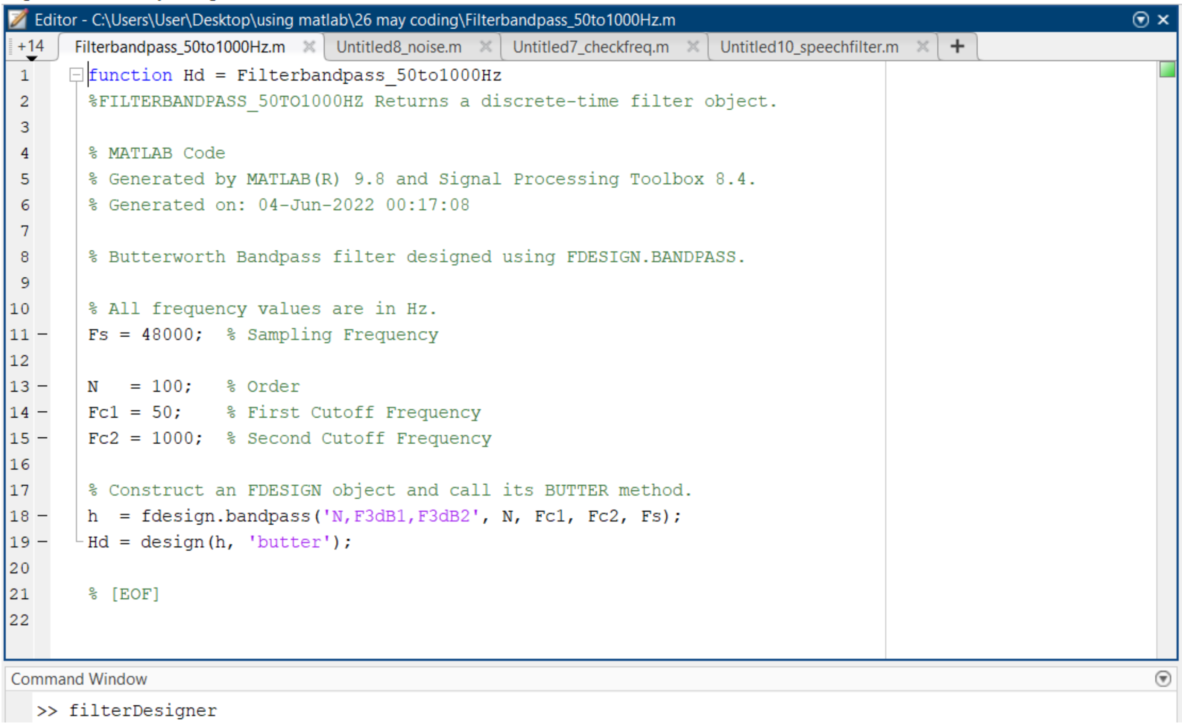

Figure 3.7 Generated MATLAB code for the designed filter

3.6 Flowchart

The flowchart presents the overall operation of the audio spectrum analyzer system for both hardware and the software developed.

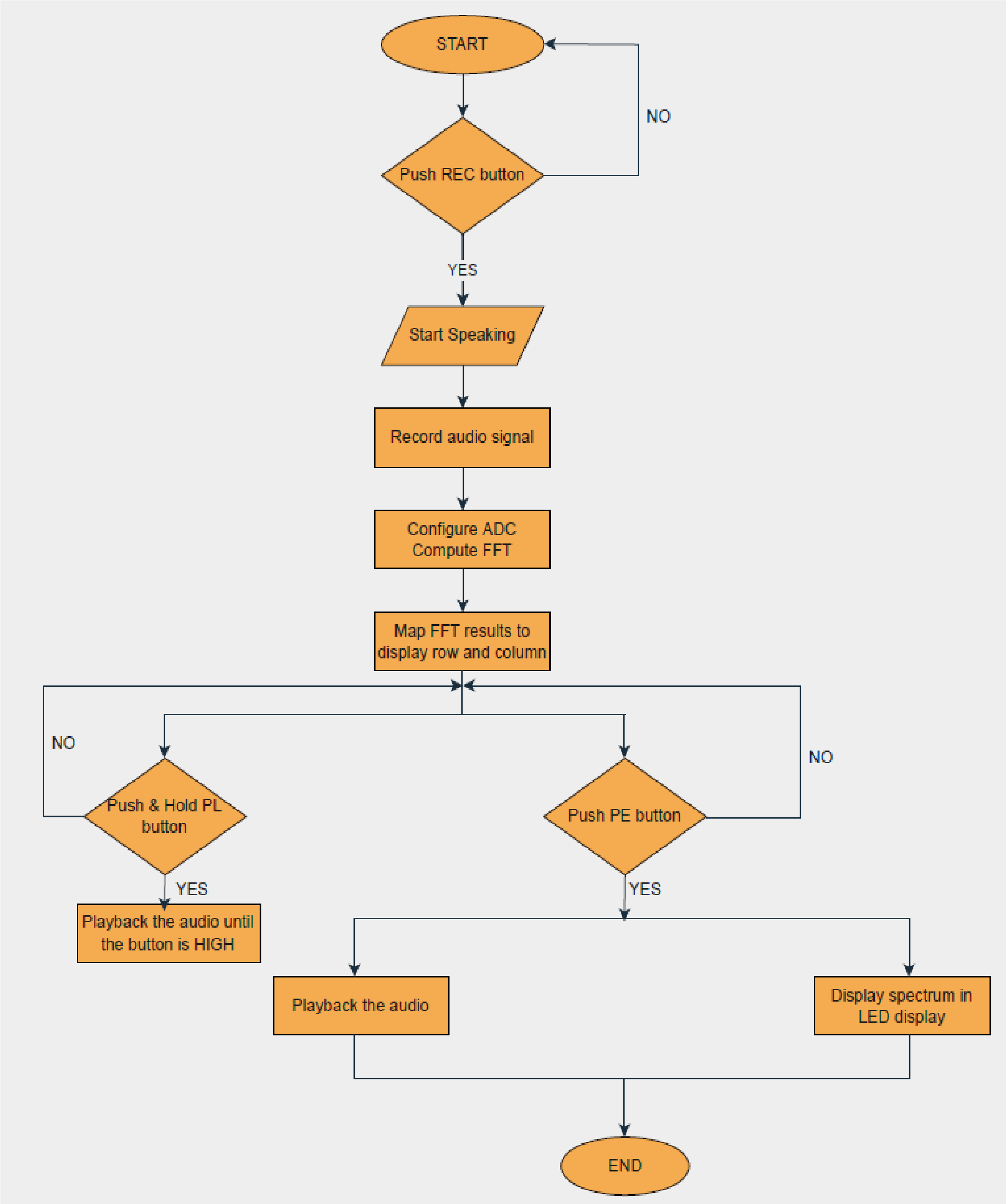

Figure 3.8 Flowchart of the audio spectrum analyzer (hardware)

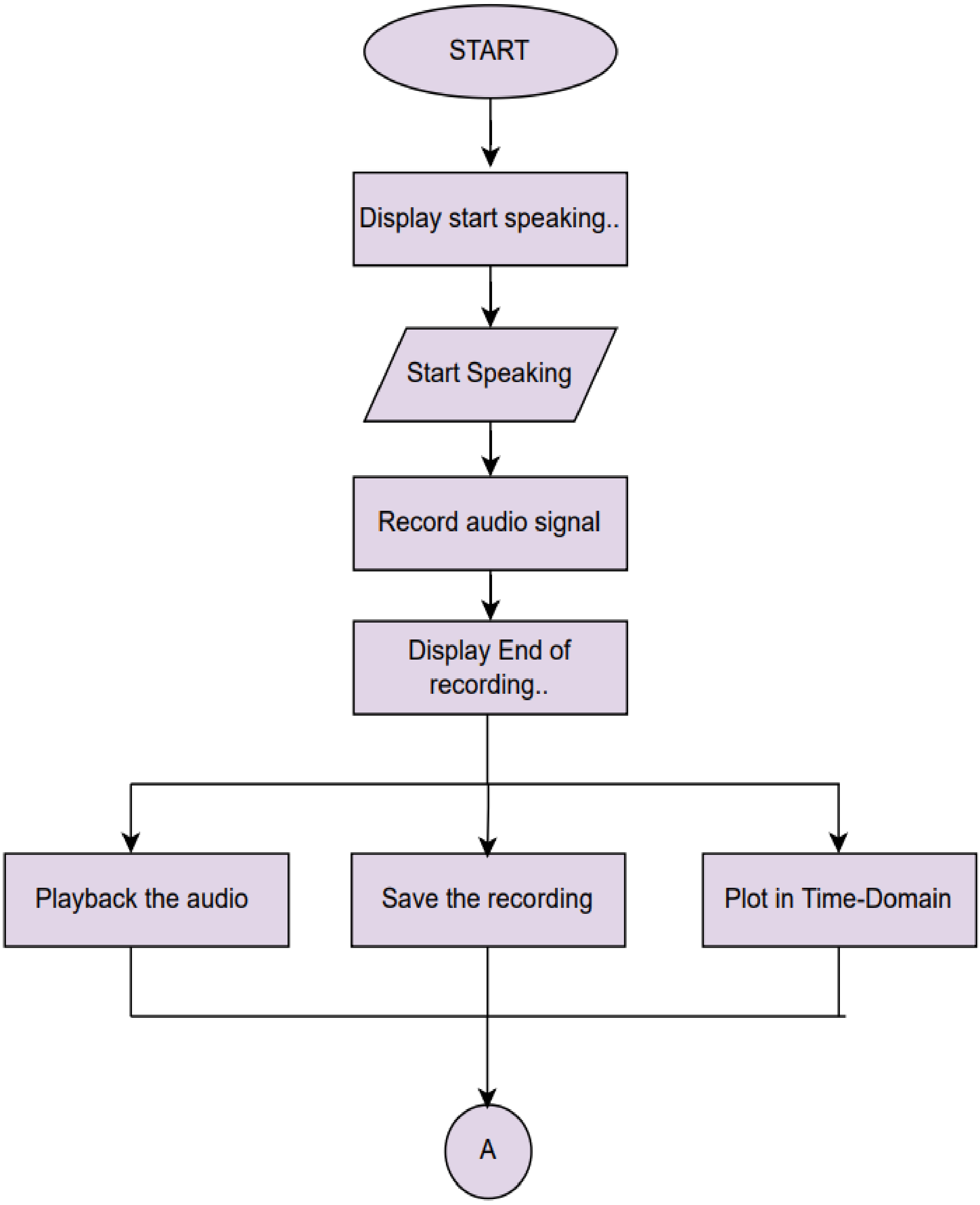

Figure 3.9 Flowchart of audio spectrum analyzer system (software)

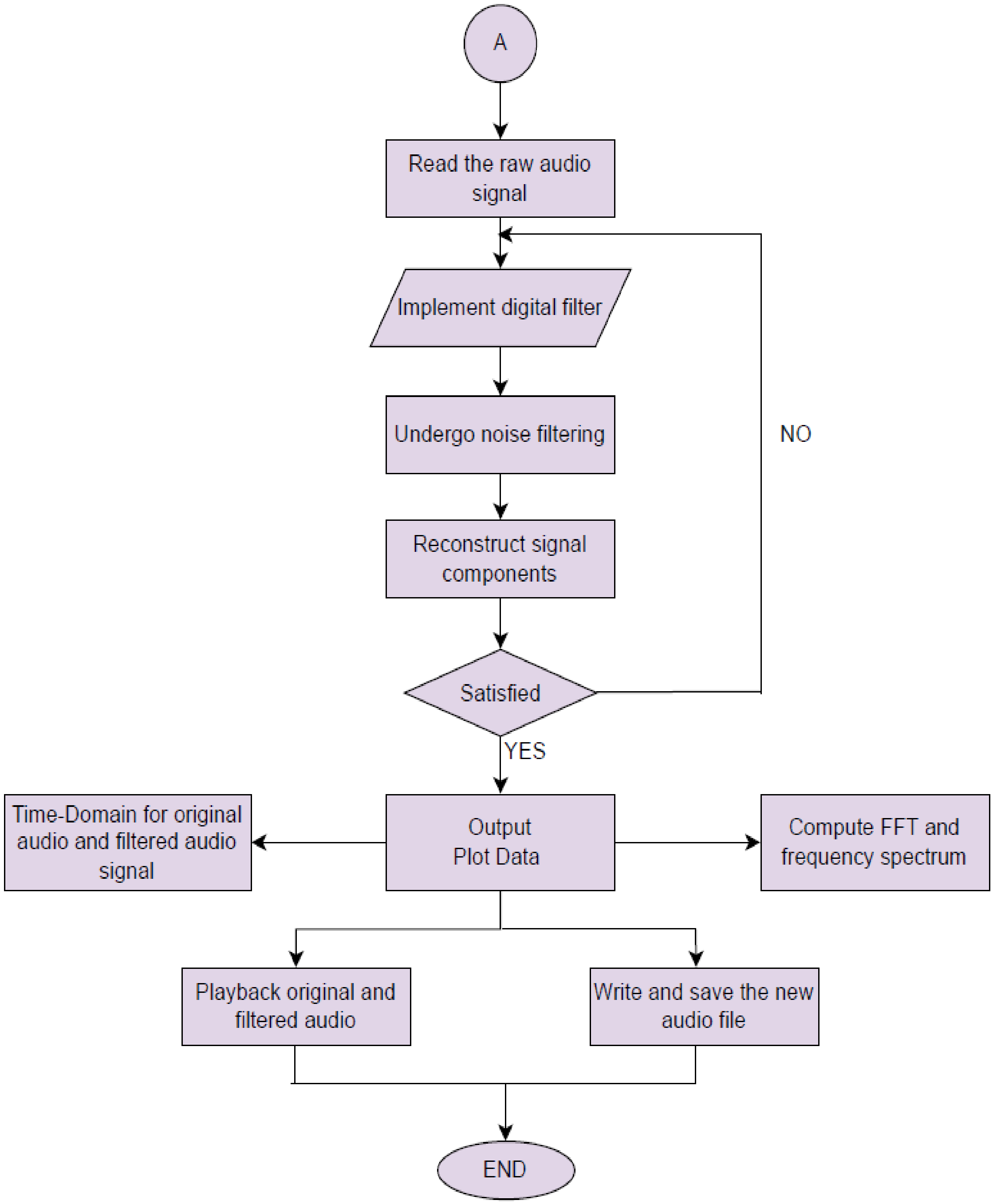

Figure 3.10 Flowchart of audio spectrum analyzer system (software)

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

In this chapter, the result from testing or demonstration and analysis of the system will be shown. The software stage of the audio spectrum analyzer was developed using MATLAB. The criteria that were chosen and tested is the real-time speech signal acquisition. Another scope of this research is to playback the audio and visualize the output spectrum in the PC using MATLAB. The implementation of filter to the recorded signal and how it effects the result was also discussed. So, the code was created and tested so its output.

4.2 Result From Audio Spectrum Analyzer (Hardware)

Figure 4.1 The fully functioning audio spectrum analyzer

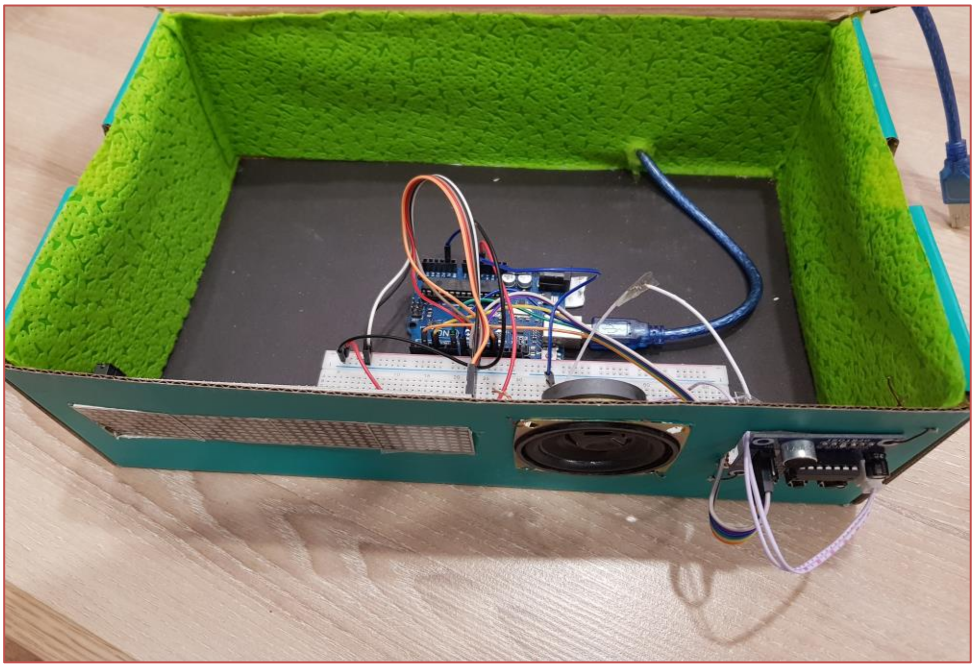

Figure 4.2 The electronic system of the hardware

Figure 4.1 and 4.2 shows the fully working audio spectrum analyzer system that could perform real-time speech acquisition, recording, playback of the speech and its spectrum visualization in the display. The hardware was tested with input and all the outputs was monitored and discussed. This full demonstration can be watched in this youtube link, https://youtu.be/WmPjJKwXS0s

4.2.1 Real-time Speech Signal and Recording

The audio system was tested by the input signal which was the speech signal of the user. In this case, the speech was recorded by the built-in microphone for about 12 seconds. This was done by pressing the REC button on the ISD 1820 voice recorder module. So, pulling the REC pin HIGH initiates a record cycle forms the beginning of the message space. The system automatically powers down after the button was released.

However, this hardware system of the audio spectrum analyzer can only record the speech signal up 20 seconds only. The input voice signals were stored directly in non-volatile cell of the module and reproduced without synthetic effect often heard with digital solid-state speech. Hence, a complete sample of the recorded speech signal was stored in a single cell where it minimizes the memory to store a single message.

4.2.2 Playback of Speech

This playback of speech can be done in this developed system when the user chooses this mode. There are two modes to play the speech that was recorded earlier. The first mode is when the PL button was pressed and hold. Thus, the system detects the LOW to HIGH signal on this pin. This initiates the playback cycle of the speech. The playback continues until PL button was pulled LOW or the end of memory space is reached, in other word full length of recording. The second mode of playback is when the PE button was pressed. Thus, the system detects the HIGH signal on the pin. This allow the device to start to playback the whole speech that was recorded. The system automatically powers down to standby mode upon the completion of the playback cycle of the recording.

This playback of the speech was played through the 3 W speaker. The speaker used allow the playback of speech in a higher output power where it can be heard loudly. This is due to the two opposite-polarity outputs provide an improvement to the output audio power of up to four times compare to the single-ended connection. So, when Sp+ and Sp- were used in this hardware, a speaker coupling capacitor is not required. A 50 kΩ resistance internally connected with the Sp+ and Sp- pins. This allow the pins to float when the system is not in the playback mode. As the system results in playback mode, it will also display the output spectrum in the MAX7219 dot matrix LED display.

4.2.3 32-band Spectrum Visualization

Another output of the developed hardware for the audio spectrum analyzer is the frequency spectrum visualization. To compute the frequency spectrum results, the system configures the ADC and will compute the FFT calculation of the speech signal that has been recorded by the system. The resolution of the built-in ADC of the Arduino board used was 10-bit. FFT algorithm was implemented to convert the input signal into frequency spectrum which is the output to display on the display.

The FFT algorithm used because it is a fast and efficient way of computing Discrete Fourier Transform (DFT). The results from the FFT are same as DFT. The only difference here is that the FFT algorithm is optimised to remove redundant calculations. In other words, the FFT greatly reduces the number of calculations required in the process. Hence it reduces the required time for the FFT spectrum computation where it provides information about frequency content, phase and other properties of the signal.

One of the important parameters that affect the result obtained as the number of FFT samples. As this hardware was programmed using Arduino IDE and involve Arduino board the FFT samples controlled after testing based on few considerations. ArduinoFFT library can do FFT samples between 16 to 128 where this can be configured in the programming code. But arduinoFFT library was slow to deal with calculation with 128 samples when the system was tested earlier. So, this research stick to the next best highest of 64-samples which is good enough for the speech signal.

The result obtained from the FFT then displayed in the 32 columns x 8 rows MAX 7219 dot matrix LED display. The library of the display provides function to turn on and off any number of the LEDs which is being used in this system. The output amplitude of every frequency band was mapped between 0 to 8 rows where it depends upon the amplitude corresponding not number of LEDs in each column get turned ON which finally allow the 32-band frequency spectrum visualized as the result in the display.

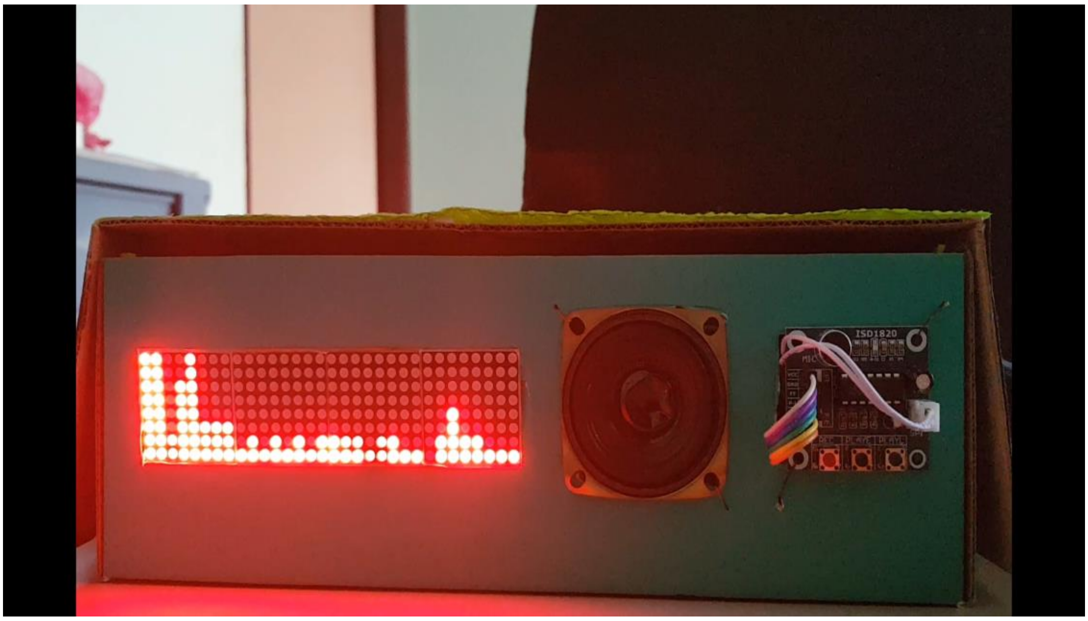

Figure 4.3 The output result obtained

For the recorded speech signal, the frequency spectrum obtained using FFT algorithm which then mapped to the display visualized when the playback of the speech through the speaker played as shown in the figure 4.3. So, all the outputs obtained proves that the hardware achieve the objectives and functioning well.

4.3 Result From the Audio Spectrum Analyzer System (Software)

The simulations of the programmed audio spectrum analyzer system are accomplished by using MATLAB. The result obtained form this system is the playback of recording, time-domain plot and frequency domain plot before and after filter implementation and finally write a new audio file which is the filtered audio signal. Thus, the discussion on the results and analysis mainly includes time-domain and frequency domain analysis of the recorded speech signal, noise reduction and filtering.

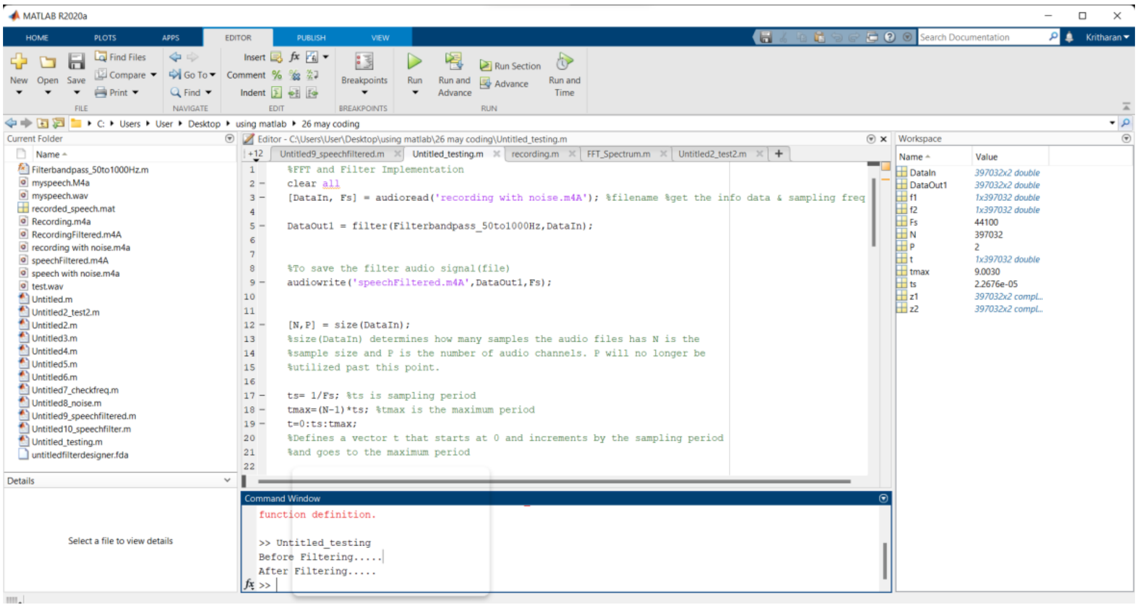

Figure 4.4 Audio spectrum analyzer developed using MATLAB

4.3.1 Speech Signal and Recording in MATLAB

This audio spectrum analyzer system developed is a part of human-computer interaction to achieve the objectives which are recording, playback of speech and spectrum visualization of the audio signal for before and after noise filtering. Input of the system which is the speech signal was recorded by the audiorecorder function in MATLAB programming. As a result, the system starts with displaying “Start Speaking” where the user’s speech signal will be started to record by the system. Then, it will display “End of recording” as the system stop the recording. The duration for the recording to be capture audio signal can be configured in the programmed. For these obtained results it was configured to be 10 second of recording.

As result at this stage, the recorded audio will be played, and the overall plot of the signal will be displayed as shown in figure 4.5 and the recorded speech will be saved as an audio file for further analysis.

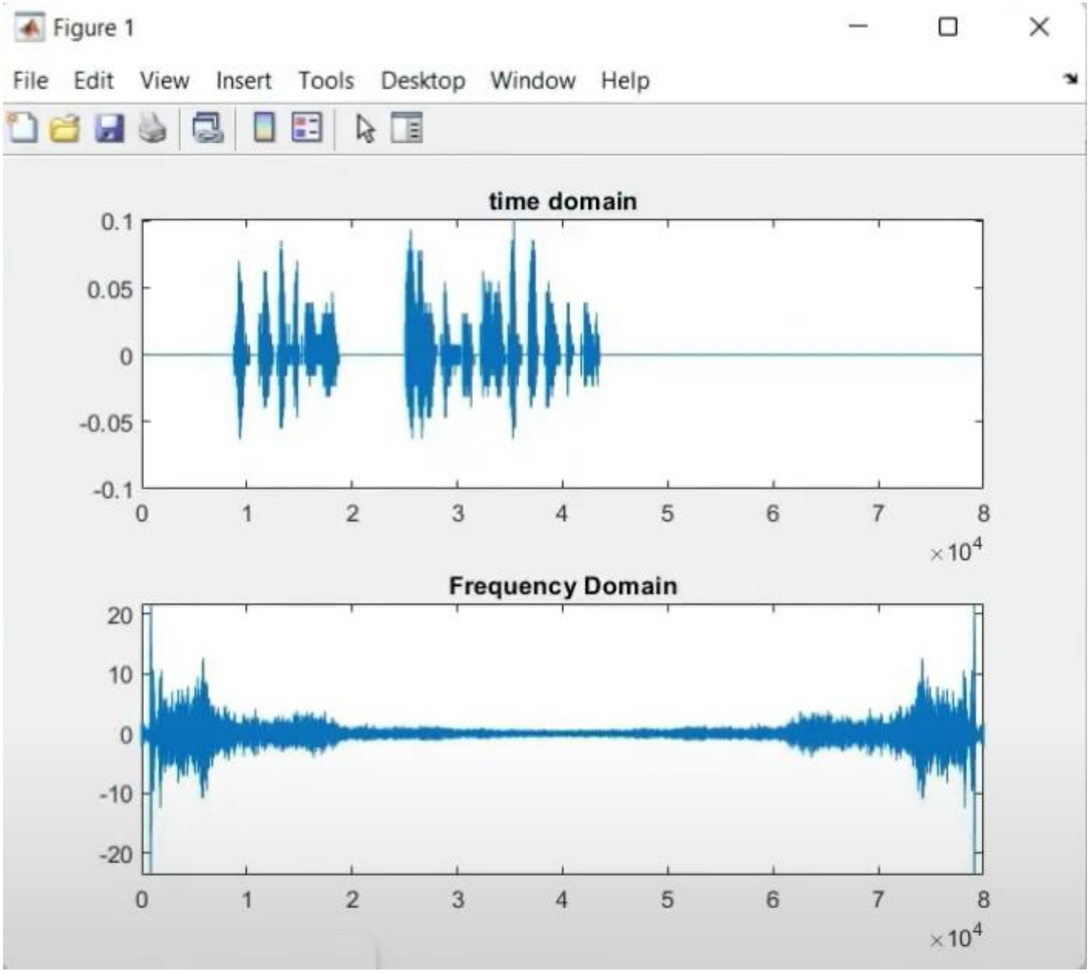

Figure 4.5 Time-domain and frequency domain plot

4.3.2 Playback of Recorded Speech

The system playback the recorded speech as the sound command was coded in the script. The recorded audio will be played in the PC speaker. It has been observed that the speech can be heard together with the noise which comes from the surrounding which affecting the speech signal and the quality of the audio. In that case, the noise which is the unwanted signal need to be filtered and gives the original information of the speech signal as the output.

4.3.3 Sampled Audio Signal

The saved recorded file will be read by the system using audioread function and the signal will be sampled. Thus, the audio signal will be sampled to yield its component and information. The parameter values obtained after the audio signal was sampled by the system as shown in table 4.1. These parameters values and the data points obtained will be used for FFT calculation and filter designing.

Parameters	Value
Sampling Frequency,Fs	44100 Hz
Samples/data length	397032
No of audio channel	2
Sampling period,Ts	$2.2676e^{-5}$ s
Time vector,t	9.0030 s

Table 4.1 Parameters result obtained

From the table 4.1 shows time vector, t shows that the total audio length that was recorded where for the tested audio file was 9.0030 seconds length. The audio data or samples was stored in double array. The channel type of this audio is stereo. One of the important parameters that need to take into consideration is the sampling frequency for the audio signal. For the sampling frequency it has been make sure that the value is high so that more samples that can be taken. Hence, the sound wave will be more accurate.

Reducing the sampling frequency will decrease the signal separation. If the sampling frequency used is below the Nyquist rate, the audio frequencies will crossover and cause aliasing to occur. Thus, increasing the sampling frequency will the best choice where more samples that will be taken, results in producing more accurate shape for the sound wave that captured by the system. In other words, the higher the quality of audio will be and ease the noise filtering process.

4.3.4 Implementation of Filter and Noise Reduction

Filters can be used in speech signal processing where it removes the unwanted signals and gives desired smooth signal. These filters may be analog or digital. So, in this research the digital type of filter was used as filters are the type of device which removes the noise from the speech signal and gives us the original information of the audio signal as the output. The fundamental voice frequency is 100 Hz to 900 Hz. So, bandpass filter was implemented where it allows to select the range of frequencies that desired, while it will prevent signals at unwanted frequencies from getting through. Infinite Impulse Response (IIR) filter is adopted to filter the audio signal. The range of cut of frequencies used is 50 Hz to 1 kHz.

In implementing filter to get the desired output an important parameter to take in consideration will be the filter order. For the filter designed in this research, higher filter order which is 100 was used. This is because higher filter order can be used to get sharp transition edge to remove noise which lies within the transition edge. The output audio after filtering has a big difference were almost all the noise were removed.

4.3.5 Time Domain and FFT Spectrum Plot

For testing and demonstration of the audio spectrum analyzer system, blender noise was used at the back as the background noise while recording the speech signal to prove that using this system the process of noise filtering can be done easily. The frequency spectrum allows us to get the information on the audio signal, the unwanted signal and where they lies in the spectrum. So, to compute it FFT algorithm was implemented using the FFT function. The number of FFT points used in MATLAB is 1024 points which later produce 512 output spectral lines. The computation of FFT will give us the spectrum in a complex function which means it can contain real and imaginary part of the signal. So, it is necessary to use absolute function when plotting the output spectrum in MATLAB code because the input signal is a real valued function.

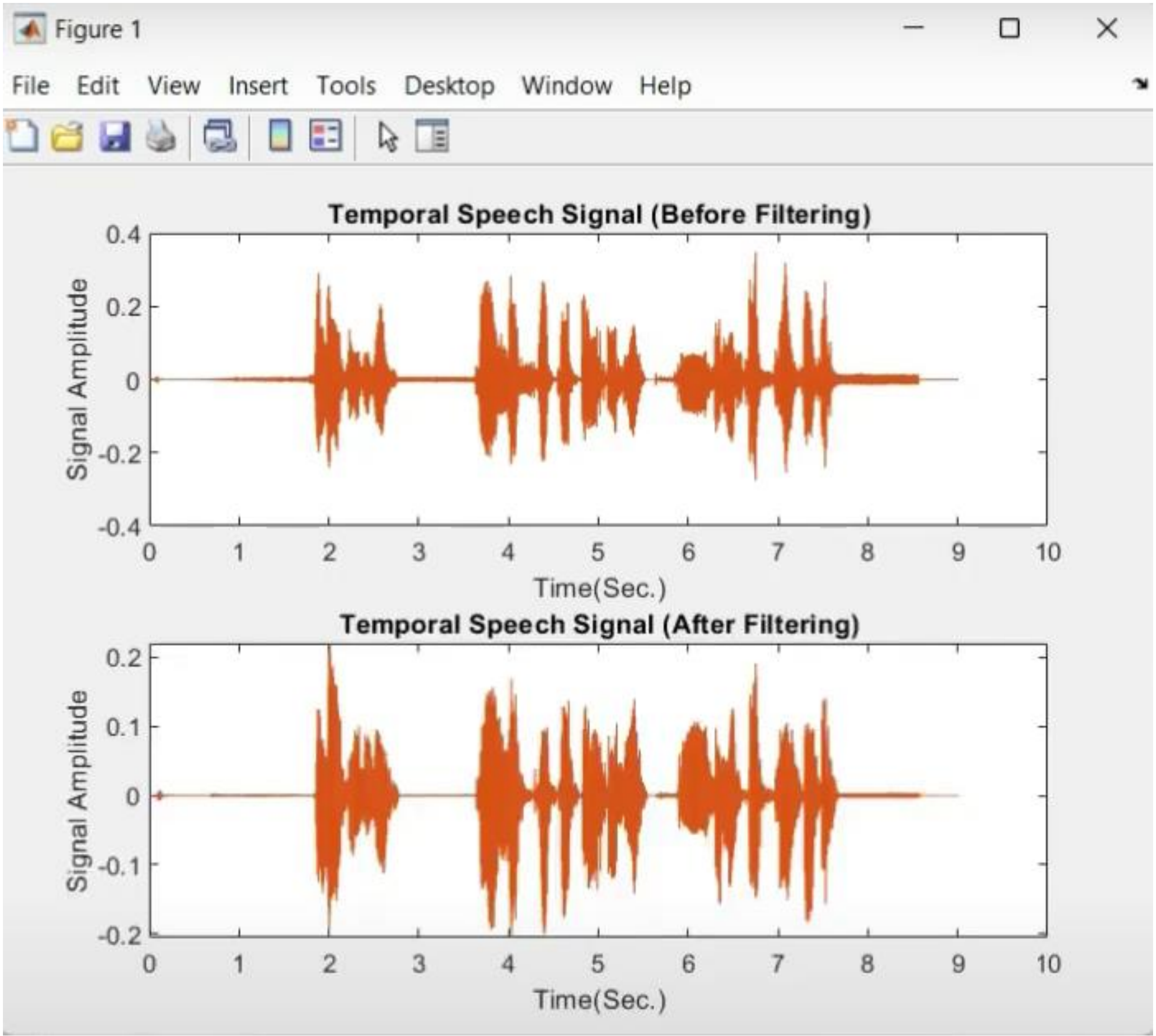

Figure 4.6 Time-Domain Plot

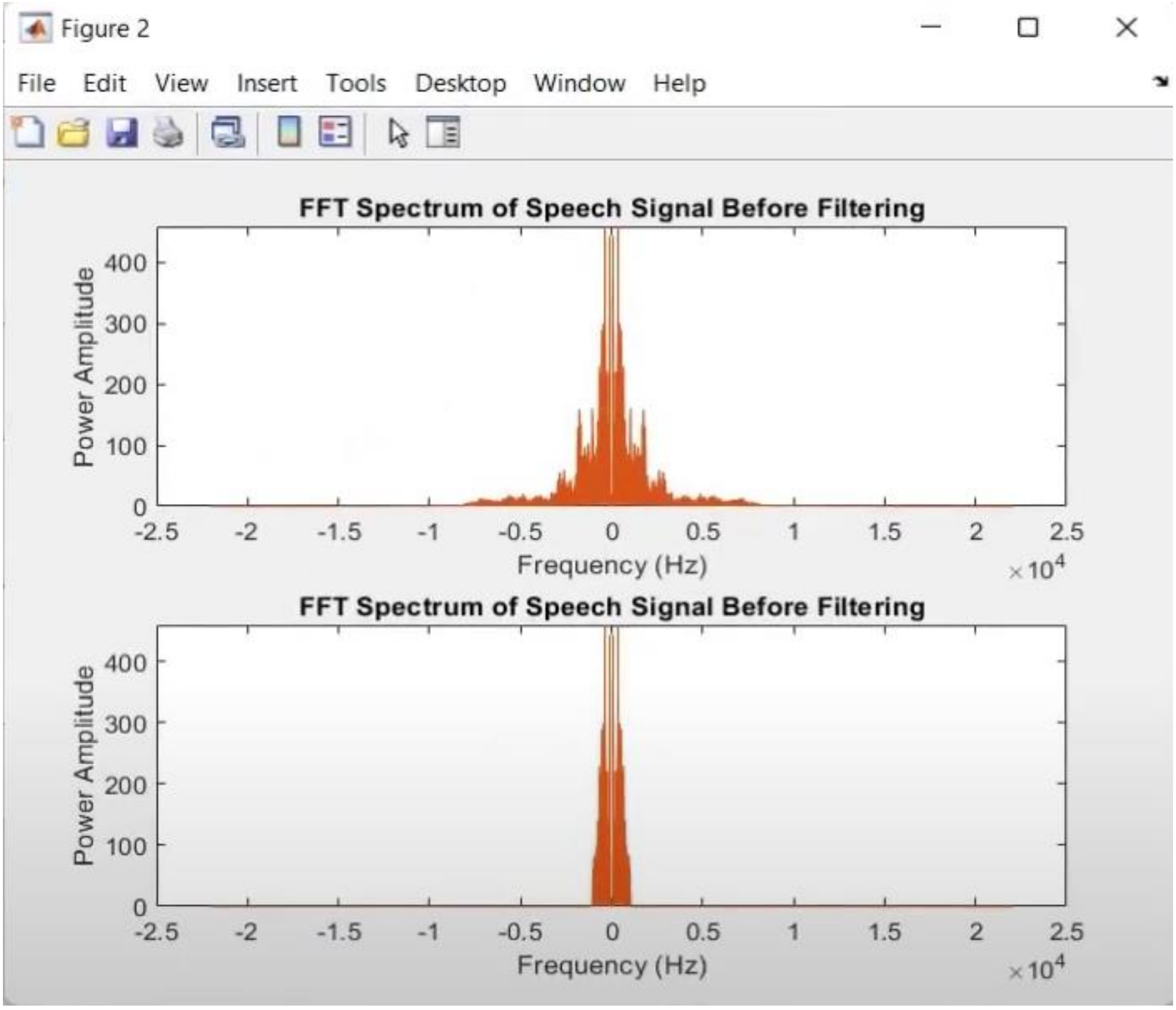

Figure 4.7 Frequency Domain Plot

Figure 4.6 shows the result obtained for the time-domain plot for the recording speech signal that was tested. The system displayed the time domain plot of the before and after filtering for the input speech signal where at the same time the audio before and after noise filtering will be playback corresponding to the signal plot as in the demonstration video.

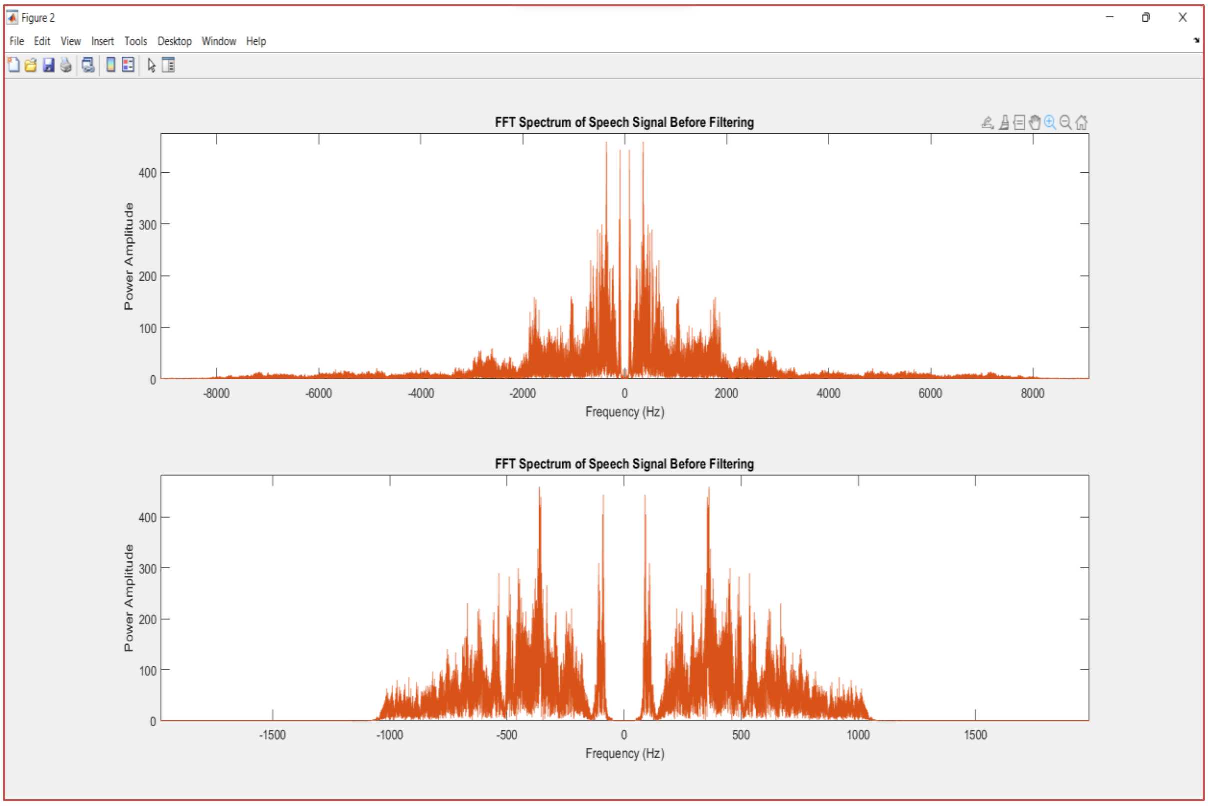

Figure 4.8 Frequency Domain Plot (zoomed)

The system also displayed the frequency spectrum that has been computed for both before and after noise filtering of the speech signal as shown in figure 4.7 and 4.8. The frequency spectrum visualized allow us to get the information on the speech signal, unwanted signal and where they lie in the spectrum. The high-power spikes occur in high frequencies is from the noise signal, so it has been filtered when the filter was used. From the results and the playback of speech both before and after noise filtering, the difference is obvious after filtering where almost all the noise from the blender sound was filtered and the better version of the audio file was produced and saved by the system.

CHAPTER 5

CONCLUSION & RECOMMENDATION

5.1 Conclusion

As conclusion, audio spectrum analyzer system using FFT algorithm is an efficient and quick way to analyze audio signal by computing frequency spectrum of the signal where it reduces the redundant calculations. In this speech signal analysis work, it mainly includes time domain and frequency domain spectrum analysis for noise reduction and filtering. The audio signal information is also important to visualize its spectrum, so that it could be analysed to get better audio quality. From this project, a low-cost a low-cost audio spectrum analyzer system developed which operates in real-time, able to playback, display the frequency spectrum by visualizing it and noise reduction using MATLAB. Thus, the objectives in this project were achieved. This design was also proven to be used as an alternative to other spectrum analyzer product design for spectrum visualizing and speech or audio correction needs which can be utilized by more people.

5.2 Future Recommendation

Based on the results and the analysis that has been done with the audio spectrum analyzer, then further research can develop the hardware by perfecting the design by using higher quality components. It will also be better if one can developed with various innovation with higher and upcoming technologies so that the hardware can handle and process higher number of samples up to 128 or more. Finally, the system design is recommended to be used in combination with othwe devices such as equalizers, since there is still a gap to perfect the audio spectrum analyzer system design.

REFERENCES

[1] Thomas, S., & Haider, N.S. (2013). A Study on BASICS OF A SPECTRUM ANALYZER.

[2] Q. Li, Y. Liu, Y. Liu and Y. Mei, "The design and implementation of Audio analyzer based on ARM," 2011 IEEE International Conference on Mechatronics and Automation, 2011, pp. 2238-2242, doi: 10.1109/ICMA.2011.5986287.

[3] Guven, Y. ı., Cosgun, E., Kocaoglu, S., Gezici, H. G. E. Z. İ. C. İ., & Yilmazlar, E. (2017). Understanding the Concept of Microcontroller Based Systems To Choose The Best Hardware For Applications. *International Journal of Engineering And Science*, *6*(9), 38–44.

[4] R. Bannatyne and G. Viot, "Introduction to microcontrollers," WESCON/97 Conference Proceedings, 1997, pp. 564-574, doi: 10.1109/WESCON.1997.632384.

[5] Badamasi, Y.A. (2014). The working principle of an Arduino. 2014 11th International Conference on Electronics, Computer and Computation (ICECCO), 1-4.

[6] Louis, L., 2016. Working Principle of Arduino and Using it as a Tool for Study and Research. International Journal of Control, Automation, Communication and Systems, 1(2), pp.21-29.

[7] Mehrotra, M., Pandey, G., & Singh Narula, M. (2017). IMPLEMENTATION OF FFT ALGORITHM. INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, doi:10.5281/zenodo.573508.

[8] E. Santos-Luna, A. Prieto-Guerrero, R. Aguilar-Gonzalez, V. Ramos, M. López-Benítez and M. Cardenas-Juarez, "A Spectrum Analyzer Based on a Low-Cost Hardware-Software Integration," 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2019, pp. 0607-0612, doi: 10.1109/IEMCON.2019.8936239.

[9] Raharjo, B. J., & Zakaria, M. (2019). Real-time audio spectrum analyzer to analyze loudspeaker. Journal of Physics: Conference Series, 1413(1), 012005. https://doi.org/10.1088/1742-6596/1413/1/012005

[10] Arun Venkatesh, K., Chauhan, S., Bhatta, S. G. H., & Mathivanan, N. (2006). Design of FFT spectrum analyzer on PC platform using LabVIEW and microcontroller-based data acquisition system.

[11] Chaudhari, M., & Kulkarni, V. (2016). Design and Implementation of Spectrum Analyzer. Journal of Advanced Research in Computer, doi: 10.17148/IJARCCE.2016.5907

[12] Editor, S. I. (2016). Research on Acquisition and Process of Speech Signals. *International Journal of Simulation: Systems, Science & Technology*. https://doi.org/10.5013/ijssst.a.17.25.18

[13] Puder, H., & Schmidt, G. (2006). Applied speech and audio processing. *Signal Processing*, *86*(6), 1121–1123. https://doi.org/10.1016/j.sigpro.2005.07.034

[14] Win, O. (2019). IIR Filter Design for De-Nosing Speech Signal using Matlab. *International Journal of Trend in Scientific Research and Development*, *Volume-3*(Issue-3), 4–8. https://doi.org/10.31142/ijtsrd21576

[15] Lee, H., & Bien, Z. (2004). Bandpass Variable-Bandwidth Filter for Reconstruction of Signals With Known Boundary in Time–Frequency Domain. *IEEE Signal Processing Letters*, *11*(2), 160–163. https://doi.org/10.1109/lsp.2003.819872

[16] Marchon, N. J. S. (2021). Sharp transition FIR bandpass filter for processing bioelectric signals. *Journal of Physics: Conference Series*, *1921*(1), 012019. https://doi.org/10.1088/1742-6596/1921/1/012019