Spectrally-Sound Portable Audio Spectrum Analyzer

Omkar Gaikwad[1], Siddhi Lokhande[2], Khushi Shingare[3], Aditya Patil[4], Prof. Sunita Jadhav[5], Dr. Jyoti Gangane[6]

[1,2,3,4,5,6]*Electronics & Telecommunications Engineering, Vishwaniketan Institute of Management Entrepreneurship and Engineering Technology (ViMEET), Raigad, India*

***Abstract*—The project "Spectrally Sound" presents the design and implementation of a compact, low-cost, and real-time audio spectrum analyzer using the ESP32 microcontroller. The primary objective of this system is to analyze sound signals and convert them into a visual representation of their frequency components, enabling users to understand how sound energy is distributed across different frequency ranges such as bass, mid, and treble. This project aims to provide an affordable and portable alternative to conventional spectrum analyzers, which are often expensive and bulky.**

The system uses an INMP441 digital MEMS microphone to capture ambient sound signals. Unlike traditional analog microphones, the digital interface (I2S) ensures better signal quality with reduced noise and distortion. The ESP32 receives this digital audio data and processes it in real time. The core processing technique used is the Fast Fourier Transform (FFT), which converts the captured audio from the time domain into the frequency domain. To improve the accuracy of the analysis and reduce errors such as spectral leakage, a windowing function (such as the Hann window) is applied before performing the FFT.

The processed frequency data is then divided into different frequency bands and displayed on an OLED display as a dynamic bar graph. This allows users to visually interpret sound intensity and frequency variations in real time. The ESP32's dual-core architecture and efficient peripheral support enable smooth execution of multiple tasks, including data acquisition, signal processing, and display control, ensuring continuous and real-time operation.

In addition to audio spectrum analysis, the system includes an electromagnetic (EM) signal detection module using a copper coil connected to the ESP32's ADC. This feature allows the system to detect the presence of nearby electromagnetic signals, adding extra functionality beyond basic audio analysis. The entire system is powered by a rechargeable battery using a TP4056 charging module, making it portable and suitable for field applications.

The project demonstrates the practical implementation of key Digital Signal Processing (DSP) concepts such as sampling, windowing, frequency transformation, and real-time visualization on an embedded platform. Although the system has some limitations in terms of precision and advanced features compared to professional analyzers, it successfully achieves its goal of providing a cost-effective, user-friendly, and educational tool for audio analysis. Furthermore, the system offers significant potential for future enhancements, including IoT integration, advanced visualization, and AI-based sound classification, making it a scalable solution for modern applications.

I. INTRODUCTION

Sound is an important part of human life, and its behavior can be better understood by studying how its energy is distributed across different frequencies. Whether it is music, speech, or environmental noise, every sound is made up of multiple frequency components. To observe and analyze these components, an audio spectrum analyzer is commonly used. Traditional spectrum analyzers, however, are often expensive and large in size, limiting their use in classrooms, small laboratories, and portable applications.

With advancements in microcontrollers and digital signal processing techniques, it has become possible to design compact and affordable spectrum analyzers that still offer accurate results. The project "Spectrally Sound" focuses on creating a portable, low-cost, real-time audio spectrum analyzer using an ESP32 microcontroller. The idea behind the project is to convert sound into a visual display that shows the strength of different frequency ranges such as bass, mid, and treble.

The system uses a microphone to capture ambient sound, which is then passed through a FET-based preamplifier to improve signal quality and reduce noise. This conditioned analog signal is fed into the Analog-to-Digital Converter (ADC) of the ESP32, which samples the sound at a frequency of 32 kHz. After digitization, the data is processed using the Fast Fourier Transform (FFT) algorithm. To ensure accuracy and minimize errors such as spectral leakage, a Hann window is applied before performing the FFT.

The resulting frequency information is displayed in real time on an LCD or LED bar graph, allowing the user to visually interpret the sound spectrum. This project demonstrates the practical use of Digital Signal Processing (DSP) concepts such as sampling, windowing, and frequency analysis on an embedded platform. It is designed using Arduino IDE or ESP-IDF and programmed in Embedded C, making the system efficient, reliable, and easy to customize.

Being battery-powered, lightweight, and portable, the device can be used in multiple fields including music visualization, sound engineering, audio equipment testing, noise monitoring, and educational demonstrations. Additionally, the ESP32 offers built-in Wi-Fi and Bluetooth capabilities, enabling future enhancements such as wireless data transmission and remote monitoring.

By combining hardware and DSP algorithms in a single compact unit, the project offers a cost-effective alternative to commercial spectrum analyzers while maintaining real-time performance and user-friendly operation.

II. RELATEDWORK

1.ESP32-Based Real-Time Audio Spectrum Analyzer

Recent open-source and academic projects have widely used the ESP32 microcontroller to implement real-time audio spectrum analyzers. These systems typically use digital microphones (such as INMP441) connected via the I2S interface to capture highquality audio signals. The ESP32 processes the audio using FFT algorithms and displays the output on OLED or TFT screens. The dual-core architecture of ESP32 allows one core to handle signal processing while the other manages display or communication tasks, resulting in smooth real-time performance. These systems are cost-effective, portable, and easy to develop. However, they often face limitations such as ADC noise (if analog input is used), limited frequency resolution, and lack of calibration features. Despite these challenges, ESP32-based systems are highly suitable for educational and basic analysis purposes.

2.Arduino-Based Spectrum Analyzer

Earlier implementations of audio spectrum analyzers were based on Arduino microcontrollers such as Arduino Uno or Mega. These systems used analog microphones and basic FFT libraries to perform frequency analysis. Due to limited processing power and memory, Arduino-based systems could only handle small FFT sizes, resulting in low frequency resolution and slower processing speed. The output was often less stable and less accurate compared to modern systems. However, these designs were simple, easy to understand, and useful for beginners. They laid the foundation for later improvements using more powerful microcontrollers like ESP32.

3.STM32-Based Audio Spectrum Analyzer

Many research projects have used STM32 microcontrollers for audio spectrum analysis due to their higher processing capability and support for optimized DSP libraries like CMSISDSP. These systems provide faster FFT computation, better accuracy, and more stable output compared to Arduino-based designs. STM32-based analyzers often include features like DMA-based data transfer, higher sampling rates, and improved signal conditioning. However, they require more complex development tools (such as STM32CubeIDE) and do not have built-in wireless communication features, which limits their flexibility for IoT-based applications. They are more suitable for intermediate to advanced embedded system development.

4.FPGA-Based Spectrum Analyzer Systems

Advanced spectrum analyzers are implemented using FPGA (Field Programmable Gate Array) platforms. These systems offer extremely high-speed processing, parallel computation, and very high frequency resolution. FPGA-based designs can handle complex DSP operations, including large FFT sizes and advanced filtering techniques, making them ideal for industrial and research applications. However, they are expensive, powerconsuming, and require deep knowledge of hardware design and HDL programming (such as VHDL or Verilog). Due to these factors, they are not suitable for low-cost or portable applications.

5.Commercial/Professional Spectrum Analyzers

Professional spectrum analyzers used in laboratories and industries provide highly accurate and calibrated measurements. These devices include advanced features such as high dynamic range, precise frequency resolution, automated calibration, and detailed graphical interfaces. They are widely used in applications like telecommunications, audio engineering, and RF analysis. However, they are bulky, expensive, and not easily portable. Their high cost makes them unsuitable for students and small-scale applications.

III.SYSTEMARCHITECTURE

The system is designed using a layered architecture, where each layer performs a specific function, ensuring smooth data flow from input to output. The overall architecture is presentedinFigure1.

1.Presentation Layer (User Interface Layer)

The presentation layer is the topmost part of the system and displays the final output to the user. It uses an OLED display (SSD1306) to show the waveform and frequency spectrum of the sound. This helps in understanding how sound energy is distributed across different frequencies. The display also shows sound intensity levels in real time. It continuously updates, so changes in sound are instantly visible. An LED is used as an indicator to show signal detection or system activity. This layer converts complex processed data into a simple visual form. Overall, it acts as a bridge between the system and the user for easy understanding.

2.Processing Layer (Core Computation Layer) The processing layer acts as the core or brain of the system. It is built around the ESP32 microcontroller, which performs all the major operations required for audio analysis. When the sound data is received, the ESP32 processes it using the Fast Fourier Transform (FFT) to convert the signal from the time domain into the frequency domain. It also handles signal analysis and controls the communication between input and output components. This layer ensures that raw audio signals are transformed into meaningful information that can be displayed effectively.

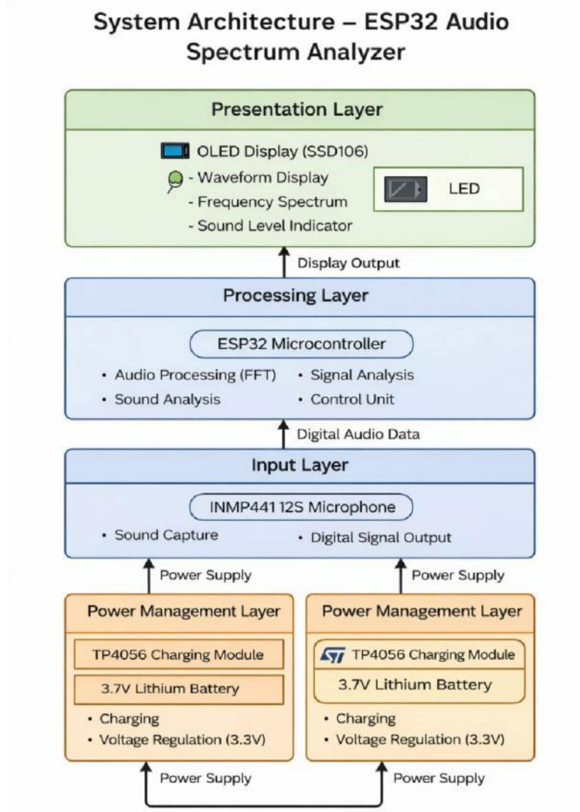

Fig. 1. ESP32-based spectrum analyzer architecture with four layers: presentation (OLED/LED output), processing (ESP32 with FFT), input (INMP441 microphone), and power (battery with TP4056).

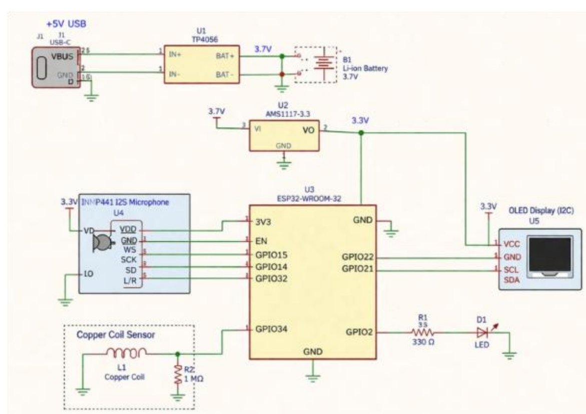

Fig. 2. Schematic of ESP32-based spectrum analyzer with I2S microphone input, EM coil sensing, OLED output, and battery-powered supply.zing

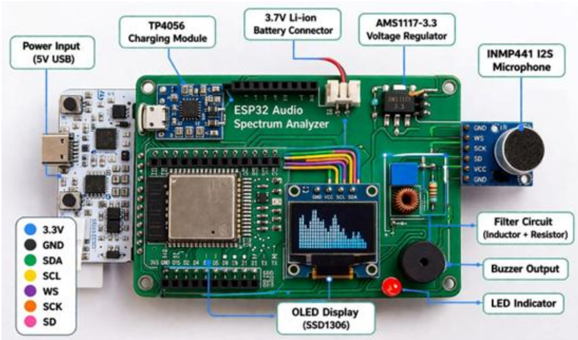

Fig. 3. Top view of the assembled ESP32 audio spectrum analyzer with microphone input, OLED display, and power module.

3. Input Layer (Data Acquisition Layer)

The input layer is responsible for capturing real-world sound signals. It uses the INMP441 I2S microphone, which directly provides digital audio output. The microphone captures sound from the environment and converts it into digital signals, which are then transmitted to the ESP32 through the I2S interface. This ensures accurate and efficient data acquisition without the need for additional analog-to-digital conversion. This layer acts as the entry point of the system where all external signals are collected.

4.Power Management Layer (Support Layer)

At the bottom, the power management layer ensures that the entire system operates reliably. It consists of a 3.7V lithium battery and a TP4056 charging module. The charging module manages battery charging and also regulates the voltage to a stable 3.3V, which is suitable for the ESP32 and other components. This layer continuously supplies power to all other layers, making the system portable and independent of external power sources.

IV. LITERATURE REVIEW

The field of audio signal analysis has evolved significantly with the advancement of digital signal processing (DSP) techniques and embedded system technologies. Traditionally, spectrum analysis was performed using high-end laboratory instruments such as real-time spectrum analyzers and signal analyzers, which offered high accuracy, wide bandwidth, and precise calibration features. However, these systems were expensive, bulky, and required specialized environments, limiting their accessibility for educational and portable applications. Early research in this domain focused on improving spectral estimation techniques, including Discrete Fourier Transform (DFT) and later the more computationally efficient Fast Fourier Transform (FFT), which became the foundation for modern spectrum analysis systems.

The introduction of FFT revolutionized real-time signal processing by significantly reducing computational complexity from O(n2) O(n^2) O(n2) to O (nlogn) O (n \log n) O(nlogn).

This advancement made it feasible to implement frequencydomain analysis on embedded platforms with limited processing capabilities. Numerous studies have demonstrated the effectiveness of FFT in decomposing complex audio signals into their constituent frequency components, enabling identification of dominant frequencies, harmonics, and noise characteristics. Researchers have also explored windowing techniques such as Hann, Hamming, and Blackman windows to reduce spectral leakage and improve frequency resolution. These techniques are now standard in modern audio spectrum analyzer designs.

With the rapid growth of microcontroller technology, recent research has shifted towards developing low-cost, portable spectrum analyzers using embedded systems. Microcontrollers such as Arduino, STM32, and ESP32 have been widely studied for their ability to perform real-time signal processing. Among these, the ESP32 stands out due to its dual-core architecture, higher clock speed, built-in Wi-Fi and Bluetooth capabilities, and support for advanced communication protocols such as I2S. Studies indicate that the ESP32 can efficiently handle real-time FFT operations while simultaneously managing peripheral interfaces and display updates. This parallel processing capability enhances system performance and responsiveness, making it suitable for applications requiring continuous realtime analysis.

In addition to processing capabilities, research has also emphasized the importance of efficient data acquisition methods. Traditional analog microphones required external ADCs, which introduced noise and reduced accuracy. The use of digital microphones such as the INMP441, which supports the I2S protocol, has addressed many of these limitations. I2Sbased microphones provide high-quality digital audio output directly to the microcontroller, minimizing signal degradation and simplifying system design. Several open-source and academic projects have successfully implemented I2S-based audio acquisition systems combined with FFT processing on ESP32 platforms, demonstrating reliable performance in realtime applications.

Another important area of research involves visualization and user interaction. Modern spectrum analyzers are expected to provide intuitive and real-time graphical outputs. Studies have explored the use of OLED and TFT displays for visualizing frequency spectra, waveform data, and sound intensity levels. The SSD1306 OLED display, in particular, is widely used due to its low power consumption, high contrast, and ease of interfacing via I2C. Researchers have shown that even resourceconstrained embedded systems can generate dynamic visualizations by mapping FFT output into frequency bands and displaying them as bar graphs or waveforms, enhancing user understanding of signal characteristics.

Furthermore, some recent works have integrated additional sensing capabilities, such as electromagnetic (EM) signal detection using coils and analog front-end circuits. These hybrid systems combine audio and EM analysis to expand the functionality of embedded analyzers. While such implementations are still in the experimental stage, they demonstrate the potential for multi-signal analysis using a single microcontroller platform. However, challenges such as noise interference, calibration complexity, and limited ADC resolution remain significant issues in these systems.

Despite the progress in low-cost implementations, several limitations have been identified in the literature. Compared to professional-grade analyzers, embedded systems often suffer from lower frequency resolution due to limited sampling rates and buffer sizes. The dynamic range is also restricted by hardware constraints, and the accuracy of results can be affected by environmental noise and imperfect signal conditioning. Additionally, real-time processing requires careful optimization of algorithms to balance speed and accuracy, especially when operating within the limited memory and computational resources of microcontrollers.

Overall, the literature clearly indicates that ESP32-based audio spectrum analyzers represent a strong balance between performance, cost, and portability. The integration of FFT-based signal processing, digital audio acquisition using I2S microphones, and real-time visualization through OLED displays has enabled the development of compact and efficient systems. While they may not match the precision of professional instruments, these systems are highly suitable for educational, experimental, and practical applications such as sound monitoring, music visualization, and embedded signal analysis. This body of research strongly supports the design and implementation of the proposed ESP32-based audio spectrum analyzer system.

V. RESEARCH METHODOLOGY

The research methodology for the ESP32-based audio spectrum analyzer is based on a systematic and structured approach that combines both hardware and software development along with experimental validation. The methodology begins with an extensive study of existing audio spectrum analyzers, digital signal processing techniques, and embedded system implementations. This includes understanding the working principles of FFT, I2S communication, and real-time data visualization. Based on this study, the system requirements and design objectives are clearly defined, focusing on achieving a low-cost, portable, and efficient real-time audio analysis system.

The next phase involves the design and selection of hardware components. The ESP32 microcontroller is chosen as the core processing unit due to its high processing capability and dualcore architecture. The INMP441 I2S microphone is selected for accurate digital audio acquisition, while the OLED display (SSD1306) is used for real-time visualization. A TP4056 charging module along with a 3.7V lithium battery is incorporated to ensure portability and stable power supply. Additionally, a copper coil sensor is integrated to detect electromagnetic signals, extending the functionality of the system. The hardware design is finalized through schematic development and proper interfacing of all components.

Following the hardware design, the software development phase is carried out. The system is programmed using the Arduino IDE, where code is developed to initialize peripherals, acquire audio data through the I2S interface, and perform signal processing using FFT algorithms.

Windowing techniques such as the Hann function are applied to improve frequency accuracy. The processed data is then mapped into frequency bands and

displayed on the OLED screen in real time. The software also includes control logic for LED indicators and optional EM signal detection through ADC inputs.
After implementation, the system undergoes testing and validation. Different types of sound inputs, such as music tones and environmental noise, are applied to evaluate system performance. The accuracy of frequency detection, responsiveness of real-time display, and stability of operation are analyzed. The output is observed and compared with expected results to verify correctness. Any noise issues, delays, or inaccuracies are identified and minimized through code optimization and hardware adjustments.
Finally, performance evaluation and analysis are conducted to assess the overall effectiveness of the system. Parameters such as processing speed, frequency resolution, portability, and power consumption are considered. The results are compared with existing systems to highlight improvements and limitations. Based on the findings, conclusions are drawn, and possible future enhancements such as IoT integration or AI-based sound classification are suggested. This methodology ensures a comprehensive development process from concept to implementation and evaluation.

VI. CONCLUSION

The “Spectrally Sound” project successfully achieves its objective of designing and implementing a portable, low-cost, and real-time audio spectrum analyzer using the ESP32 microcontroller. The system is capable of capturing sound signals using the INMP441 digital microphone, processing them efficiently using Digital Signal Processing techniques such as the Fast Fourier Transform (FFT), and displaying the frequency components visually on an OLED display. This enables users to clearly observe how sound energy is distributed across different frequency ranges such as bass, mid, and treble.
One of the major strengths of this project is the effective utilization of the ESP32 microcontroller, which offers high processing speed, dual-core architecture, and built-in peripherals like I2S and ADC. These features allow smooth realtime signal acquisition, fast FFT computation, and continuous display updates without delay. The use of digital audio input through I2S improves signal quality and reduces noise compared to traditional analog systems, enhancing overall system accuracy and performance.
The integration of additional features such as electromagnetic (EM) signal detection using a copper coil further expands the functionality of the system beyond basic audio analysis. The inclusion of a battery-powered design with a TP4056 charging module ensures portability and ease of use, making the device suitable for field applications, educational demonstrations, and laboratory experiments.
Moreover, the project demonstrates the practical application of important Digital Signal Processing (DSP) concepts, including sampling, windowing, frequency transformation, and real-time data visualization. It serves as an effective learning tool for students and beginners to understand complex signal processing techniques in a simple and interactive manner in conclusion, the “Spectrally Sound” system provides a cost-effective, compact, and efficient alternative to traditional spectrum analyzers. While it may not match the precision of high-end professional instruments, it successfully balances performance, affordability, and usability. The project also offers great potential for future enhancements such as wireless monitoring, mobile application integration, and improved signal processing techniques, making it a scalable and innovative solution in the field of embedded systems and audio analysis.

VII. LIMITATIONS OF THE STUDY

The “Spectrally Sound” project, while effective and low-cost, has certain limitations that affect its performance compared to professional systems. One of the main limitations is the accuracy of the ESP32’s ADC and microphone input, which may introduce noise and reduce precision. As a result, the system may not provide highly accurate measurements required for advanced or industrial-level applications.
Another limitation is the restricted frequency resolution, which depends on the number of FFT points and sampling rate. Since the system uses limited memory and processing resources, it cannot achieve very high resolution like professional spectrum analyzers. This may lead to less detailed frequency analysis, especially for closely spaced signals.
The project also faces challenges related to signal noise and interference, particularly in real-world environments. External noise, improper grounding, or

weak signals can affect the quality of the input, leading to inaccurate or unstable output on the display. Additionally, the EM signal detection using a copper coil is basic and does not provide precise measurement or identification of specific electromagnetic sources.

Another constraint is the limited display size and visualization capability. The OLED display can only show a basic bar graph, which may not be sufficient for detailed analysis or advanced graphical representation. Furthermore, the system lacks proper calibration mechanisms, which are important for ensuring consistent and reliable results over time.

Finally, while the ESP32 is powerful, it still has limitations in terms of processing capacity compared to high-end DSP hardware, which may affect performance when handling complex signal processing tasks or higher sampling rates. Despite these limitations, the system remains highly useful for educational and basic analysis purposes.

VIII. DISCUSSION

The “Spectrally Sound” project demonstrates that a low-cost embedded system can effectively perform real-time audio spectrum analysis. The ESP32 microcontroller proved to be efficient in handling multiple tasks such as audio acquisition, FFT processing, and display control simultaneously. The implementation of FFT successfully converted time-domain signals into frequency-domain data, allowing clear visualization of sound components like bass, mid, and treble. The OLED display showed smooth and responsive output, confirming the system’s real-time capability.

The use of the INMP441 I2S microphone significantly improved signal quality compared to analog microphones by reducing noise and distortion. Additionally, applying a Hann window before FFT helped minimize spectral leakage and improved frequency accuracy. However, minor fluctuations were still observed due to environmental noise and hardware limitations. The system performance is also dependent on factors like sampling rate and FFT size, which create a trade-off between resolution and processing speed.

The integration of electromagnetic (EM) signal detection using a copper coil adds extra functionality to the system. While it successfully detects EM variations, the feature is basic and lacks precision and calibration, making it more suitable for demonstration purposes. The overall system is user-friendly, portable, and suitable for educational and practical applications, though it lacks advanced features like data logging and wireless connectivity.

In comparison to existing systems, the proposed design offers a good balance between cost, performance, and portability. Although it cannot match the accuracy of professional analyzers, it effectively demonstrates real-time DSP implementation on an embedded platform. The results indicate that the system meets its objectives while also highlighting areas for future improvement such as enhanced resolution, noise reduction, and feature expansion.

IX. FUTURE WORK

Enhanced FFT and Signal Processing: Increasing the FFT size (e.g., 1024 or 2048 points) can improve frequency resolution. Advanced DSP techniques like adaptive filtering and smoothing can also be implemented for better analysis.

Advanced Visualization: The system can be upgraded with larger displays such as TFT screens or integrated with mobile applications for interactive and detailed visualization. Features like peak hold, waveform display, and dynamic scaling can enhance user experience.

Multi-Channel Audio Support.: The system can be expanded to process multiple audio inputs simultaneously, allowing comparison and analysis of different sound sources, which is useful in advanced applications.

Data Storage and Logging: Integration with SD cards or cloud storage can allow recording and playback of audio data and frequency analysis results for future reference and detailed study.

Improved EM Signal Detection: The EM detection module can be enhanced using better sensors and calibration techniques to provide more accurate and meaningful electromagnetic signal analysis.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to their project guide for the continuous support, valuable guidance, and constructive suggestions provided throughout the development of this project titled “Spectrally Sound – ESP32-Based Audio Spectrum Analyzer.” The guidance received was instrumental in successfully completing this work.

The authors are also thankful to the faculty members of the department for their encouragement and for providing the necessary academic environment and resources required for this project. Their insights into digital signal processing and embedded systems greatly contributed to the successful implementation of the system.

The authors would like to acknowledge the support and cooperation of their peers for their helpful discussions and Integration of IoT Features: The system can be enhanced using the ESP32’s built-in Wi-Fi and Bluetooth to enable real-time data transmission to cloud platforms or mobile apps. This allows remote monitoring, data logging, and analysis from anywhere, making the system more flexible and useful for smart suggestions during the course of this project applications.

Finally, the authors express their heartfelt gratitude to their families for their constant encouragement and support.

Implementation of AI/ML Algorithms: Machine Learning models can be added to classify different types of sounds such as speech, music, or noise. This will help in intelligent applications like noise pollution monitoring, voice recognition, and smart surveillance systems.

Improved Accuracy with Advanced Sensors: Using higherquality microphones or external ADC modules can significantly improve signal accuracy. Better signal conditioning circuits and filtering techniques can reduce noise and distortion, leading to more reliable results.

REFERENCES

[1] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2007.

[2] V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2010.

[3] S. Monk, *Programming the ESP32 in Arduino*. New York, NY, USA: McGraw-Hill Education, 2018.

[4] Espressif Systems, *ESP32 Technical Reference Manual*, 2023.