Stereo Audio Spectrum Visualizer

Technical Documentation

Max Strack

Kaiserslautern, 24.08.2025

Table of Contents

1 Introduction and Project Overview

Audio spectrum visualization has evolved from simple VU meters to sophisticated real-time frequency domain displays that provide detailed insights into the spectral content of audio signals. This project presents the design and implementation of a stereo audio spectrum visualizer that combines analog frequency analysis with digital LED matrix visualization to create a comprehensive audio monitoring system.

The system processes stereo audio input through dual MSGEQ7 graphic equalizer integrated circuits, which perform real-time spectral analysis across seven discrete frequency bands ranging from 63Hz to 16kHz. The resulting amplitude data is processed by an Arduino Uno microcontroller, which implements custom algorithms to drive a 16×16 WS2812B LED matrix. This approach creates a dynamic visual representation that responds in real-time to the frequency content and amplitude characteristics of the input audio signal.

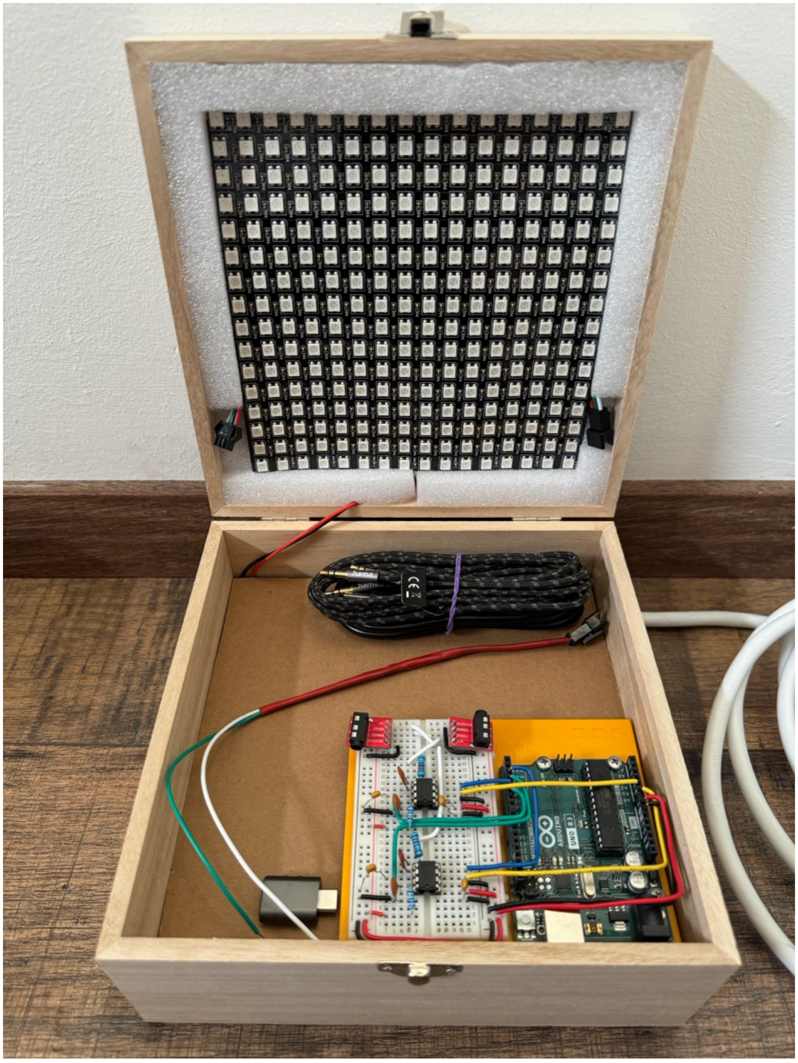

Figure 1: Complete Stereo Audio Spectrum Visualizer system

The motivation for developing this system arose from the need to create a practical demonstration of embedded systems integration that combines analog signal processing, digital control systems, and advanced power management techniques. Unlike software-based spectrum analyzers, this hardware implementation provides tactile learning experiences in circuit design, embedded programming, and electrical safety practices while delivering real-time performance that cannot be easily achieved through pure software solutions.

2 Design Philosophy and Technical Objectives

The development of this spectrum visualizer was guided by several fundamental design principles that prioritize both technical excellence and educational value. The system architecture emphasizes modularity, allowing individual subsystems to be analyzed, modified, and upgraded independently while maintaining overall system integrity.

Proper power management formed a key aspect of the design approach. Rather than compromising on LED brightness or system reliability through inadequate power distribution, the design incorporates a dedicated 90W Mean Well power supply unit. This decision was driven by the theoretical maximum power consumption of 76.8W when all 256 RGB LEDs operate at maximum brightness, requiring a current capacity of approximately 15.36A at 5V DC. The implementation includes electrical isolation between mains voltage components and low-voltage control circuits, demonstrating appropriate safety practices for embedded systems development.

The choice of the MSGEQ7 integrated circuit for frequency analysis represents a balanced approach between complexity and functionality. While digital signal processing techniques such as Fast Fourier Transform (FFT) could provide higher resolution spectral analysis, the MSGEQ7 offers several advantages for this application. The IC provides hardware-based peak detection with built-in decay characteristics, eliminating the computational overhead that would be required for equivalent software-based processing on the Arduino platform. Additionally, the seven-band frequency distribution aligns well with the 16-column LED matrix layout, providing optimal visual representation without requiring complex interpolation algorithms.

3 System Architecture and Component Selection

The system architecture follows a hierarchical approach with clearly defined interfaces between major functional blocks. The audio input stage implements proper impedance matching and signal conditioning to ensure optimal performance of the MSGEQ7 analysis circuits. High-pass filtering removes DC components and low-frequency noise that could interfere with the spectral analysis, while maintaining signal integrity across the intended frequency range.

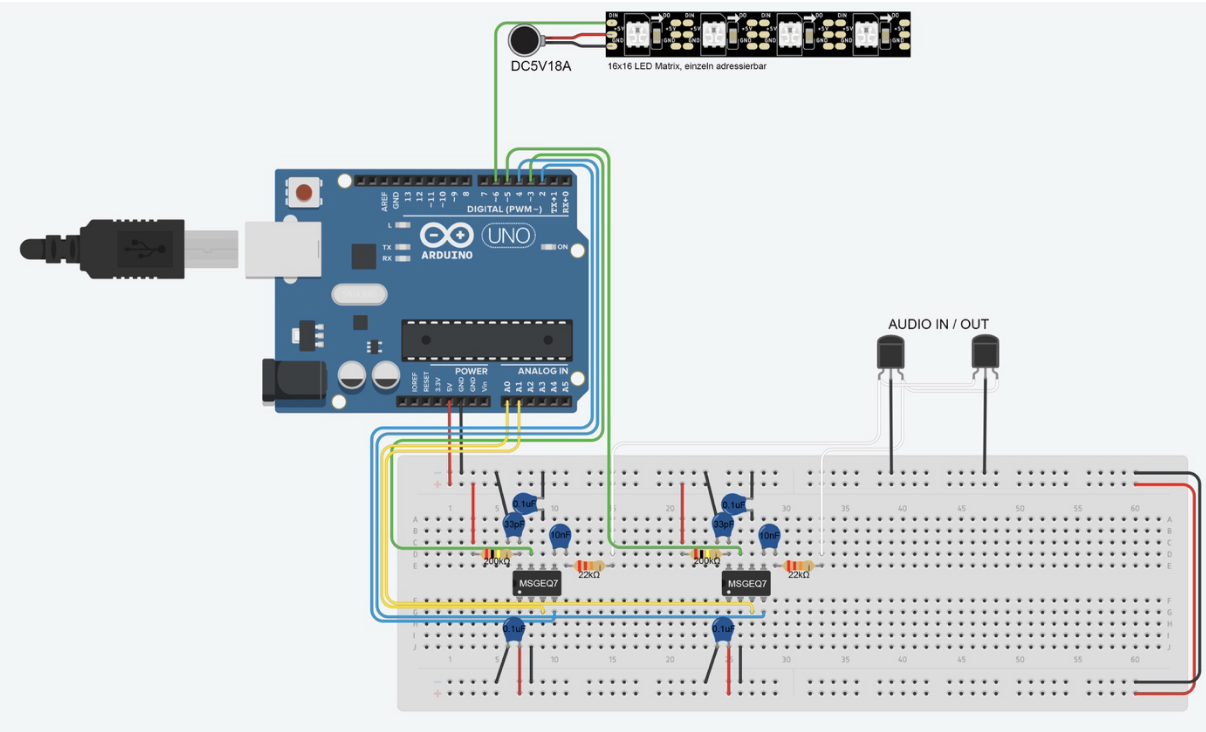

Figure 2: High-level system architecture

The dual MSGEQ7 configuration enables true stereo processing, with each integrated circuit dedicated to analyzing one channel of the stereo input signal. This approach preserves the spatial characteristics of the audio content and allows for visualization techniques that highlight stereo separation and channel-specific frequency content. The multiplexed output structure of the MSGEQ7 requires precise timing control from the Arduino microcontroller, with strobe and reset signals synchronized to ensure accurate data acquisition from each frequency band.

The WS2812B LED matrix was selected for its individually addressable pixel architecture and integrated control logic. Each LED incorporates a dedicated driver IC that enables precise color control and eliminates the need for external LED driver circuits. The serial data interface simplifies wiring complexity and reduces the number of required I/O pins on the microcontroller, while the integrated signal reshaping ensures reliable data transmission across the entire 256-pixel array.

4 Technical Specifications and Performance Characteristics

The system operates with a base clock frequency of 16MHz provided by the Arduino Uno's ATmega328P microcontroller. The MSGEQ7 integrated circuits utilize an internal oscillator configured with external components (200kΩ resistor and 33pF capacitor) to achieve the specified operating frequency of approximately 150kHz. This configuration ensures optimal filter performance and maintains the specified center frequencies for each of the seven frequency bands.

The frequency response characteristics of the MSGEQ7 provide overlapping bandpass filters with center frequencies at 63Hz, 160Hz, 400Hz, 1kHz, 2.5kHz, 6.25kHz, and 16kHz. Each filter exhibits a quality factor (Q) of approximately 6.0, providing sufficient selectivity to distinguish between adjacent frequency bands while maintaining adequate bandwidth to capture the energy content of complex audio signals.

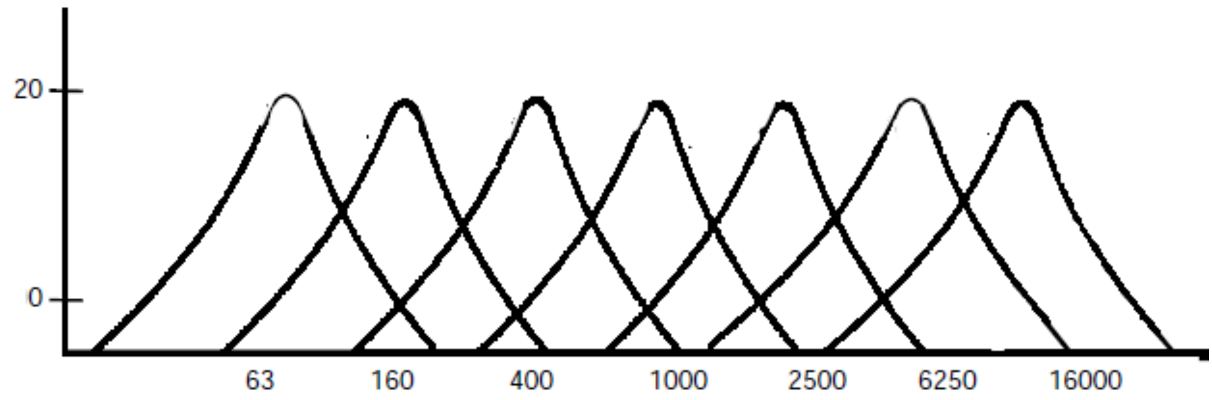

Figure 3: Frequency response characteristics
Source: MSGEQ7 Datasheet, Mixed Signal Integration Corp.

The LED matrix refresh rate provides smooth visual transitions that follow the dynamics of the audio input through optimized serial communication with the WS2812B arrays and efficient memory management within the Arduino firmware. The color mapping algorithm implements a continuous gradient from blue (representing low amplitudes) through green to red (representing high amplitudes), providing intuitive visual feedback that corresponds to conventional audio level representations.

5 Hardware Implementation and Circuit Design

The hardware implementation emphasizes both signal integrity and safety considerations through careful component selection and circuit topology design. The system is constructed on a breadboard platform that provides systematic signal routing while maintaining flexibility for development and testing.

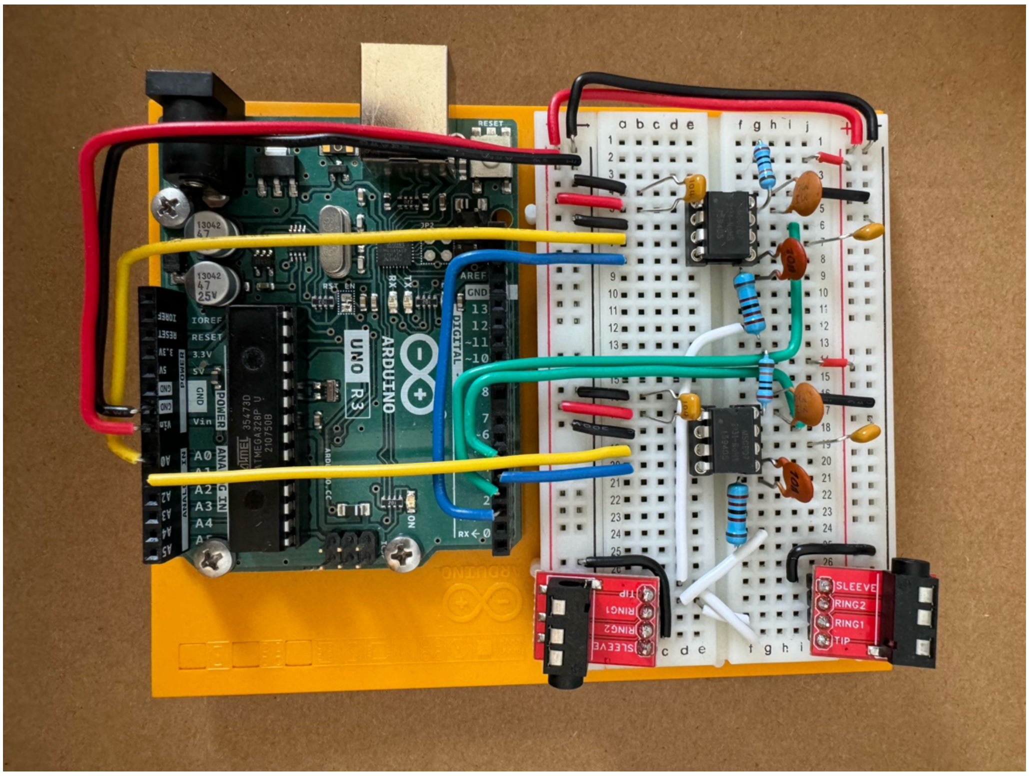

Figure 4: Hardware Design, Arduino and Breadboard

5.1 Audio Input Processing and Signal Conditioning

The audio input stage utilizes 3.5mm stereo audio jacks to separate the left and right channels for independent processing by the dual MSGEQ7 circuits. Each channel incorporates a high-pass filter consisting of a series 22kΩ resistor and 10nF capacitor. According to the project documentation, this filter configuration is designed to remove DC bias components and provide input conditioning for the MSGEQ7 integrated circuits. The 22kΩ input resistor provides current limiting protection for the CMOS inputs of the frequency analysis circuitry.

5.2 MSGEQ7 Frequency Analysis Implementation

The dual MSGEQ7 configuration represents the core signal processing element of the system. Each integrated circuit is configured according to the manufacturer's reference design, with critical attention paid to the oscillator components that determine the internal clock frequency and, consequently, the center frequencies of the seven bandpass filters. The external oscillator network consists of a 200kΩ resistor and 33pF capacitor connected between the CKIN pin and the positive supply rail.

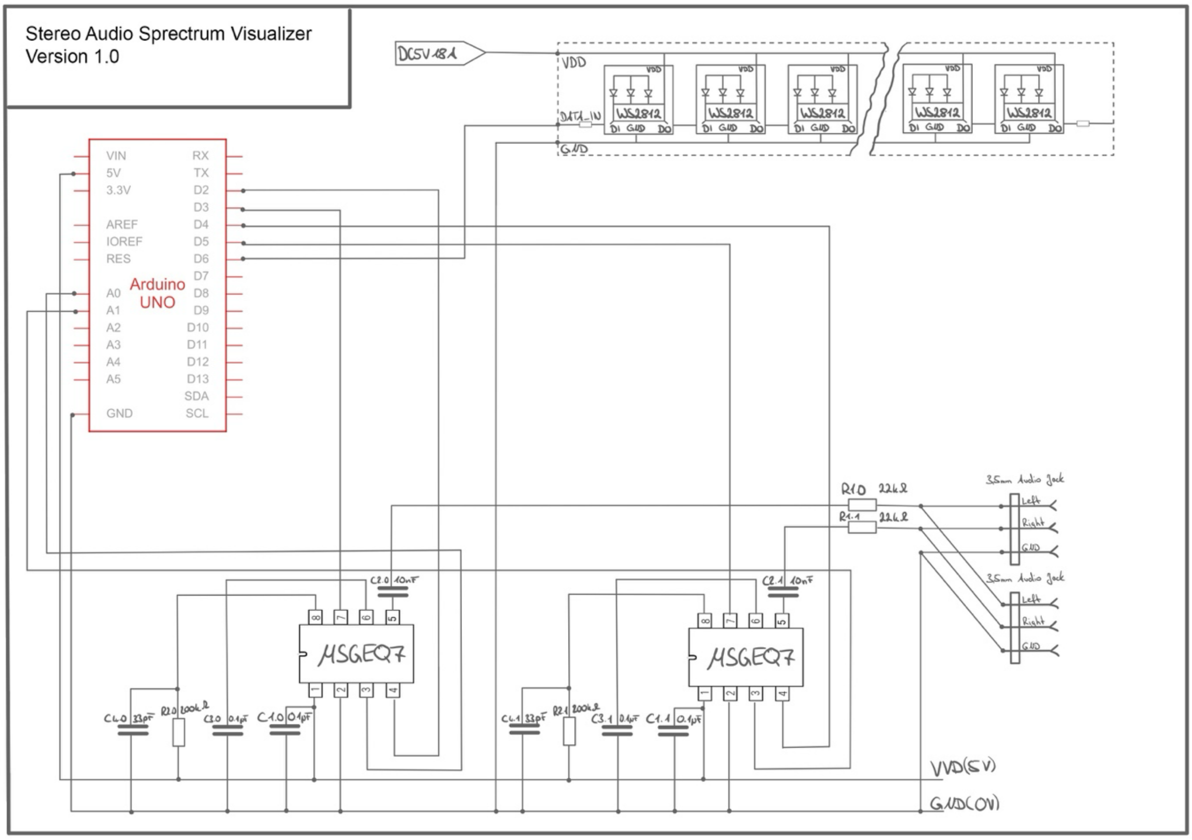

Figure 5: Complete system schematic showing dual MSGEQ7 configuration

The power supply decoupling strategy implements multiple ceramic capacitors to ensure stable operation under dynamic load conditions. Each MSGEQ7 receives a 0.1µF decoupling capacitor directly at the VDD pin, with an additional 0.1µF capacitor connected between the GND reference pin and system ground. This dual-decoupling approach addresses both high-frequency switching noise and provides a clean reference for the internal voltage generators.

The control interface between the Arduino and the MSGEQ7 circuits utilizes dedicated digital I/O pins for the RESET and STROBE signals. According to the firmware implementation, the left channel MSGEQ7 connects to digital pins 3 (RESET) and 2 (STROBE), while the right channel utilizes pins 5 (RESET) and 4 (STROBE). The analog output signals from each MSGEQ7 are routed to the Arduino's analog-to-digital converter inputs A0 and A1, providing 10-bit resolution for amplitude measurements.

5.3 Power System Design and Safety Implementation

The power distribution system addresses both the high current requirements of the LED matrix and the safety considerations associated with mains-powered equipment. The theoretical maximum power consumption occurs when all 256 RGB LEDs operate at full brightness, requiring approximately 0.3W per pixel and totaling

76.8W. To provide adequate current capacity, a Mean Well 90W power supply unit (5V/18A) was selected.

According to the project documentation, the power supply installation incorporates several safety measures. The mains voltage components are housed in a dedicated wooden enclosure with cable strain relief implemented using three cable ties to prevent accidental disconnection. Physical separation between high-voltage (230V AC) and low-voltage (5V DC) sections is maintained using cardboard barriers to reduce the risk of accidental contact during operation.

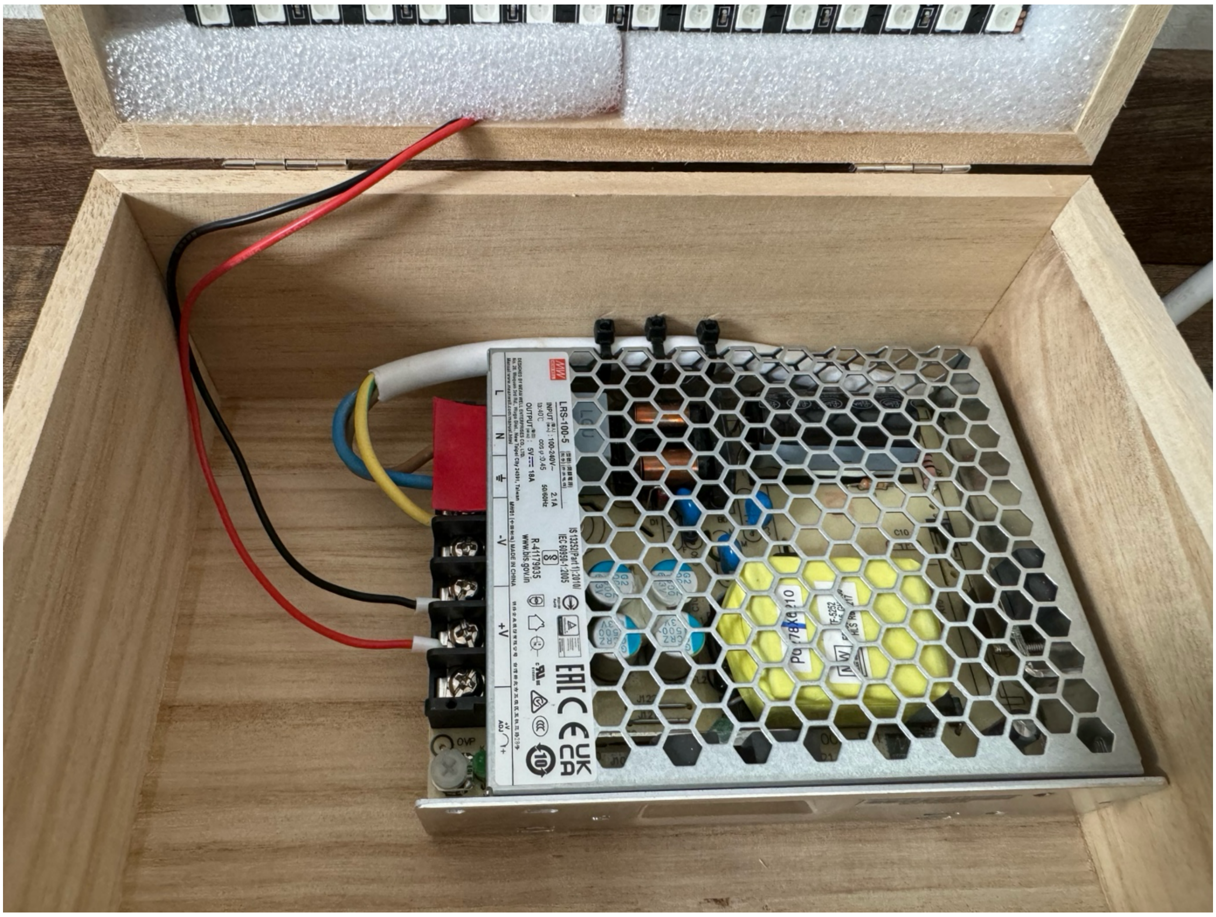

Figure 6: Professional power supply installation with safety isolation

The connection to the LED matrix is implemented through a dedicated cable assembly with appropriate connectors, reducing wiring complexity while maintaining signal integrity for the WS2812B control protocol.

5.4 LED Matrix Interface and Signal Integrity

The WS2812B LED matrix interface utilizes individually addressable RGB LEDs, each incorporating an integrated controller. The data connection from the Arduino utilizes digital pin 6, as specified in the firmware implementation. The LED matrix power distribution is implemented independently from the Arduino power supply to prevent interference between the high-current LED drivers and the audio processing circuitry.

6 Software Architecture and Firmware Implementation

The firmware architecture implements a streamlined real-time processing pipeline that coordinates MSGEQ7 data acquisition, amplitude processing, and LED matrix visualization within a continuous control loop. The implementation prioritizes timing accuracy and signal fidelity while maintaining code clarity and maintainability.

6.1 Main Control Loop Architecture

The system operates through a simple yet effective three-stage control loop executed continuously at maximum speed. The loop() function coordinates sequential execution of data acquisition (readData()), LED pattern calculation (setLEDs()), and display update (matrix.show()). This architecture ensures minimal latency between audio input and visual response while maintaining deterministic timing characteristics.

The absence of explicit timing control in the main loop allows the system to operate at the maximum refresh rate achievable by the hardware constraints, primarily limited by the MSGEQ7 multiplexer timing requirements and the WS2812B serial communication protocol.

6.2 MSGEQ7 Data Acquisition Protocol

The readData() function implements the precise timing protocol required by the MSGEQ7 integrated circuits. The data acquisition sequence begins with a synchronous reset of both MSGEQ7 multiplexers, ensuring that subsequent strobe pulses access identical frequency bands on both channels.

The frequency band reading process follows the MSGEQ7 datasheet specifications closely. Each band requires a strobe signal transition from HIGH to LOW, followed by a mandatory 36-microsecond settling time before ADC conversion. This settling time ensures that the MSGEQ7's internal peak detector circuits have stabilized and provide accurate amplitude measurements.

```
delayMicroseconds(36); // Wait for MSGEQ7 output to settle (datasheet requirement)

// Read ADC values from both channels
leftMSGEQ[band] = analogRead(analogPins[0]);
rightMSGEQ[band] = analogRead(analogPins[1]);
```

The ADC values undergo immediate processing through constrain() and map() functions to remove noise below the threshold and scale the amplitudes to the appropriate LED column height range. This processing converts the raw 10-bit ADC values (0-1023) into LED column heights (0-16), with values below the noise filter threshold of 90 effectively eliminated.

6.3 LED Matrix Addressing and Visualization

The setLEDs() function translates the processed frequency data into individual pixel commands for the WS2812B matrix. The visualization scheme dedicates columns 0-6 to the left audio channel and columns 9-15 to the right channel, with columns 7-8 remaining unlit to provide clear visual separation between stereo channels.

The calcPixelIndex() function addresses the physical wiring pattern of the LED matrix, which implements a zigzag configuration to minimize wiring complexity. Even-numbered columns are addressed from bottom to top (pixel 15 at bottom, pixel 0 at top), while odd-numbered columns are addressed from top to bottom (pixel 0 at top, pixel 15 at bottom). This addressing scheme ensures that amplitude visualization appears consistent across all columns, with higher amplitudes always appearing toward the top of the display.

6.4 Color Mapping and Visual Feedback

The colorMap() function implements a perceptually intuitive color gradient that transitions from blue (low amplitude) through green (medium amplitude) to red (high amplitude). This color scheme aligns with conventional audio level indicators and provides immediate visual feedback regarding signal strength and frequency content.

The color calculation utilizes a two-stage linear interpolation process. Values in the lower half of the amplitude range (0-127) transition from pure blue to pure green, while values in the upper half (128-255) transition from pure green to pure red. The smoothingOffset parameter prevents abrupt color transitions at the maximum amplitude level, ensuring smooth visual continuity.

6.5 Performance Characteristics and Timing Analysis

The system's refresh rate is determined by the cumulative timing requirements of the MSGEQ7 reading sequence and the WS2812B update protocol. Reading all seven frequency bands from both channels requires approximately 1.26 milliseconds (7 bands × 2 channels × 90µs per reading), while updating the complete 256-LED matrix requires approximately 7.68 milliseconds.

The total system latency from audio input to visual output is therefore dominated by the LED matrix update time, resulting in a theoretical maximum refresh rate of approximately 100-120 Hz. In practice, the system operates somewhat slower due to the additional processing overhead in the setLEDs() function and the Arduino's execution speed limitations.

7 System Performance and Operational Characteristics

The system performance is determined by the timing characteristics of the two main components: the MSGEQ7 frequency analysis and the WS2812B LED matrix communication. According to the respective datasheets, these components define the overall system refresh capabilities.

7.1 MSGEQ7 Data Acquisition Timing

The MSGEQ7 datasheet specifies the following timing requirements for reading all seven frequency bands:

- Reset pulse width: 100ns minimum
- Reset to strobe delay: 72µs minimum
- Strobe pulse width: 18µs minimum
- Strobe to strobe delay: 72µs minimum
- Output settling time: 36µs minimum

Reading all seven frequency bands from one MSGEQ7 requires approximately 630µs (7 × 90µs per band). Since the system processes two channels sequentially, the complete audio data acquisition cycle requires approximately 1.26ms.

7.2 LED Matrix Communication Performance

The WS2812B protocol requires specific timing for data transmission:

- Data transfer time per bit: 1.25µs ±600ns
- Each LED requires 24 bits (8 bits × 3 colors)
- Total time per LED: approximately 30µs

Updating the complete 256-LED matrix requires approximately 7.68ms (256 × 30µs), which represents the dominant factor in determining the overall system refresh rate.

7.3 Power Consumption Analysis

Based on the WS2812B datasheet specifications, each LED consumes approximately 0.3W at maximum brightness (full white). The theoretical maximum system power consumption is:

- LED Matrix: 256 × 0.3W = 76.8W
- Arduino and MSGEQ7 circuits: approximately 1-2W
- Total maximum: 79W

The selected 90W Mean Well power supply provides adequate headroom for this worst-case scenario, though typical music content results in significantly lower average power consumption due to the dynamic nature of the frequency content.

8 Testing and Validation

8.1 Power System Validation and Design Evolution

Initial testing of the system revealed critical insights regarding power distribution requirements. The original design attempted to power the LED matrix directly from the Arduino's power supply. During functional testing, systematic activation of all LEDs revealed significant voltage drops measurable with the Benning MM7-1 multimeter from the project's instrumentation. These voltage drops caused system instability and unexpected resets, clearly demonstrating the inadequacy of the Arduino's power distribution for high-current LED applications.

This practical experience led to the implementation of the dedicated 90W Mean Well power supply, which eliminated the voltage stability issues completely. Subsequent measurements during normal operation showed that actual power consumption remains well below the theoretical maximum, as typical audio content produces dynamic frequency distributions rather than sustained full-brightness conditions across all LEDs.

8.2 Thermal Management and Safety Considerations

Critical Safety Note: The Mean Well power supply unit is designed with an open architecture to facilitate natural convection cooling. During the development phase, the power supply was temporarily covered with a cardboard separator for electrical isolation. The system must not be operated with the cardboard cover in place, as this prevents adequate cooling and poses serious thermal safety risks.

Practical measurements during extended operation confirmed that normal audio visualization maintains low average power consumption, and the power supply exhibits no thermal stress under typical operating conditions. However, the open cooling design must be maintained for safe operation, particularly during any testing that approaches the theoretical maximum power consumption.

8.3 Circuit Validation and Performance Verification

The dual MSGEQ7 configuration was validated through systematic testing of each frequency band using the Benning MM7-1 multimeter to verify DC output levels across the specified frequency range. The Arduino's analog-to-digital conversion provides adequate resolution for the amplitude measurements, with consistent readings across both stereo channels.

The WS2812B LED matrix communication proved robust during testing, with the timing-critical serial protocol maintaining reliable data transmission across all 256 pixels without observable data corruption or color artifacts.

8.4 Practical Performance Limitations and Signal Integrity Issues

During extended testing, two specific performance limitations became apparent that illustrate the challenges of real-world analog signal processing in embedded systems.

Limited Dynamic Range Utilization: The uppermost two pixels in each frequency column remain consistently unlit during normal operation, regardless of input signal strength. This limitation stems from the practical operating characteristics of the MSGEQ7 integrated circuits, which typically provide output voltages that correspond to ADC values in the range of 900-1000 counts rather than the theoretical maximum of 1023. Consequently, the linear mapping function rarely produces values exceeding 14-15, leaving the topmost pixels unused. This behavior represents a common discrepancy between theoretical component specifications and practical operating characteristics in analog signal processing applications.

Signal Artifacts at Low Input Levels: Occasional flickering of individual pixels in the lower regions of the display occurs intermittently, particularly during periods of low audio input. Analysis of this phenomenon indicates that electromagnetic interference from the audio input cables and open 3.5mm connector contacts introduces noise that can momentarily exceed the 90-count noise filter threshold. This interference is particularly noticeable when the audio cables are positioned near the LED matrix or power supply components, demonstrating the importance of proper signal routing and electromagnetic compatibility considerations in mixed-signal systems.

These limitations do not significantly impair the system's primary functionality as an audio visualization tool, but they illustrate important considerations for signal integrity and analog circuit design in practical embedded systems applications.

9 Results and Demonstration

9.1 Visual Representation and Display Characteristics

The completed system successfully demonstrates real-time stereo audio spectrum visualization through dynamic LED matrix display. The 16×16 pixel arrangement provides clear visual separation between the left and right audio channels, with each channel utilizing seven columns to represent the respective frequency bands from 63Hz to 16kHz.

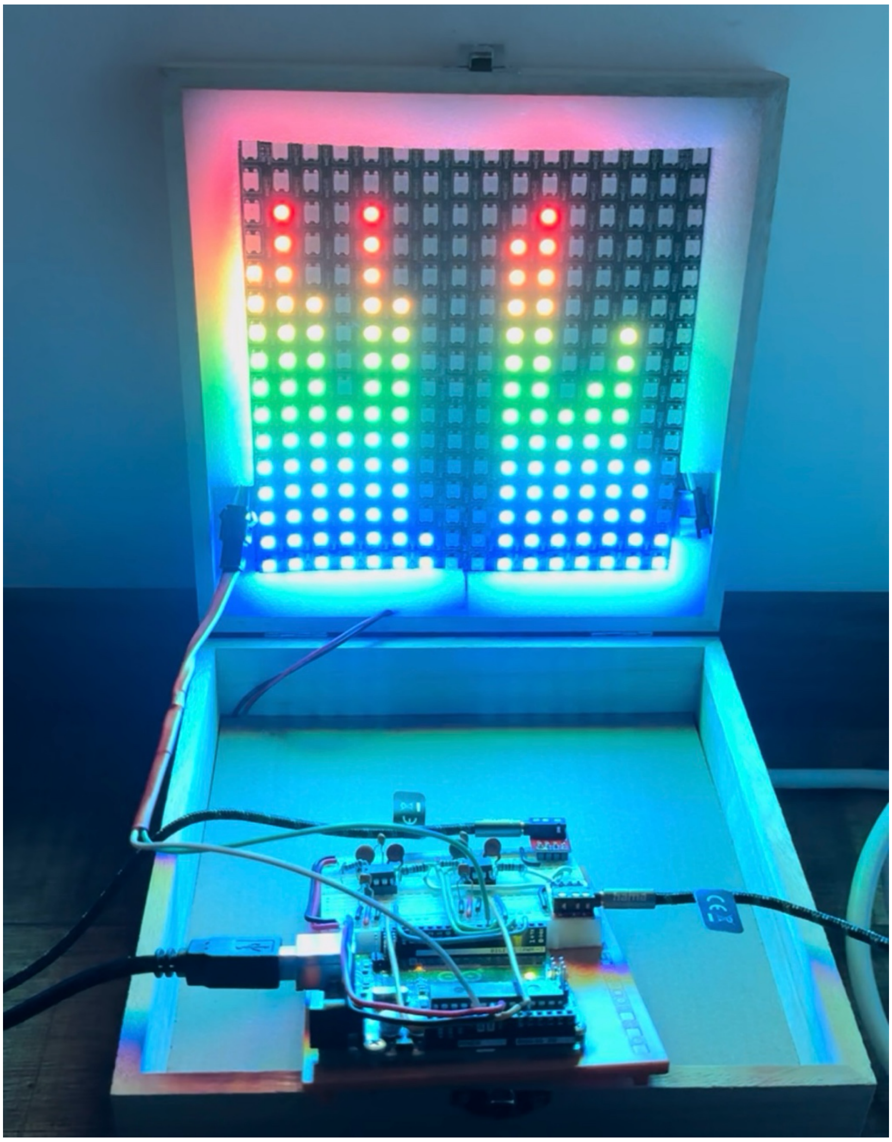

Figure 7: LED matrix displaying active frequency visualization

The visual layout implements a symmetrical design where the left channel occupies columns 0-6 and the right channel utilizes columns 9-15, leaving the center columns (7-8) unlit to provide clear stereo separation. This arrangement allows immediate visual identification of channel-specific frequency content and stereo imaging characteristics of the audio material.

9.2 Color Mapping and Amplitude Representation

The color mapping algorithm successfully translates the MSGEQ7 amplitude measurements into intuitive visual feedback through a continuous gradient progression. Low amplitude signals appear in blue, transitioning through green for moderate levels, and culminating in red for high amplitude content. This color scheme aligns with conventional audio level indicators and provides immediate visual feedback regarding the dynamic range of each frequency band.

The amplitude scaling utilizes the constrain() and map() functions to convert the 10-bit ADC values into the appropriate LED brightness levels. A noise filter threshold of 90 (out of 1023) effectively eliminates background noise while preserving sensitivity to genuine audio content. The mapping range of 0-16 provides sufficient resolution for visual distinction while preventing LED saturation that could reduce color differentiation.

9.3 System Integration and Performance Demonstration

The complete system demonstrates stable operation with various audio sources, successfully visualizing the frequency content of different musical genres and audio types. The real-time response characteristics provide immediate visual feedback that closely follows the temporal characteristics of the input audio, creating an engaging audiovisual experience.

The modular construction approach facilitates maintenance and potential modifications, while the wooden enclosure provides both aesthetic appeal and functional protection for the power supply components. The external audio connections enable easy integration with various audio sources including portable devices, audio interfaces, and sound systems.

10 Conclusion

This project successfully demonstrates the integration of analog frequency analysis, embedded microcontroller programming, and advanced LED matrix control to create a functional stereo audio spectrum visualizer. The implementation combines the MSGEQ7's hardware-based frequency analysis capabilities with the Arduino platform's programming flexibility and the WS2812B's individual pixel control to achieve real-time audiovisual conversion.

The development process revealed important practical considerations regarding power system design, thermal management, and electrical safety that extend beyond the theoretical aspects of the individual components. The initial power distribution challenges and subsequent implementation of the dedicated 90W power supply illustrate the importance of comprehensive system-level analysis in embedded systems design. The critical thermal management requirements of the power supply demonstrate how safety considerations must be integrated into both the design and operational procedures of such systems.

Additionally, the project highlighted common discrepancies between theoretical component specifications and practical performance characteristics. The limited dynamic range utilization of the MSGEQ7 outputs and the susceptibility to electromagnetic interference from audio input cables represent typical challenges encountered in real-world analog signal processing applications. These observations underscore the value of thorough testing and validation in embedded systems development.

From an engineering perspective, the project successfully balances technical complexity with practical implementation constraints. The dual MSGEQ7 architecture provides true stereo processing while maintaining manageable complexity for the Arduino microcontroller. The modular design approach enables future enhancements such as additional visualization modes, wireless connectivity, or expanded frequency analysis capabilities. The system serves as an effective demonstration platform for embedded systems concepts while providing practical utility as an audio visualization tool.

The completed visualizer represents a comprehensive example of modern embedded systems development, incorporating analog signal processing, digital control algorithms, high-current power management, and safety engineering practices within a single integrated system.