International Journal of Scientific Research in Multidisciplinary Studies
Vol.**10**, Issue.**8**, pp.**09-15**, August **2024**
E-ISSN: 2454-9312 P-ISSN: 2454-6143
Available online at: www.isroset.org

Research Article

Music and Electronics: Audio Players outside the Black Box

Marcelo Moreira[1*]

[1]KPIT Technologias LTDA. R. James Watt, 84, Andar 8, Sala 2, São Paulo, Brazil, 04576-050

Corresponding Author: marcelo.mmoreira2@gmail.com

Received: 18/Jun/2024; **Accepted**: 20/Jul/2024; **Published**: 31/Aug/2024

Abstract— Electronic equipment can capture music in nature and convert it to a digital audio format. Music in the digital audio format can be listened to using audio players with headphones or speakers. This paper will present the design and development of an embedded system for audio reproduction that works as an audio player. The device was developed using an Arduino Micro and an audio decoder with an SD card in C language with local and remote controls. It is physically built with PCB fabrication. The paper may be used to understand players' inside behavior and develop new audio player architectures and similar devices with new functionality and user cases.

Keywords— music; digital audio format; audio players; Arduino micro.

1. Introduction

Music is a universal language beyond culture, country, life philosophy, gender, or race. It can connect people from anywhere because it relies on the same principles: notes and physics. Music can generate mood and promote emotions by activating different brain neurons and changing brain chemistry by stimulating electro signals. Therefore, it is used for psychological therapy or dementia treatment. [1], [2], [3], [4], [5]

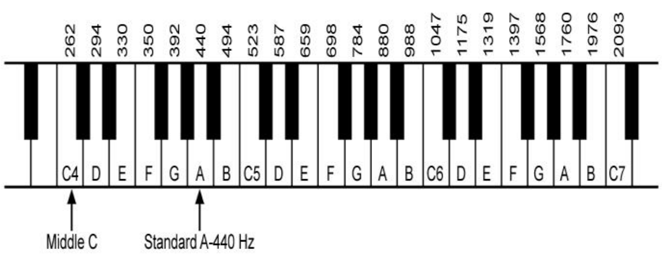
Music can be perceived as notes or as electronics and physics. In a conservatory, music is perceived as notes and chords. [6] The most used musical notation in the world divides all possible sounds into 12 notes. It uses seven notes called natural and 5 Sharps or 5 Flats. Each key in the piano is a single note. In the same way, each fretboard in the guitar is a single note.

A set of notes played or sung apart is called melody, and a set of notes played simultaneously is called harmony. The notes score from Figure 1 (top) shows the notes and their duration. [7], [8] The chord score of Figure 1 (bottom) shows the chords or harmony of a song.


```
    Eb
Overtime
          Cm7(9)                Fm7(9)  Ab7M/Bb
I've been building my castle of love
         Eb7M
Just for two
          Cm7M(9)
Though you never knew you
        F/A  G/B
Were my rea..son
```

Figure 1. Notes Score and Chord Score.
Source: Own authorship.

On the other hand, in a studio, music can be perceived as physics and electronics. The sound is the variation of air pressure. The sound is acquired in a studio and passes through an extensive electronic process to output a digital audio format. Each note is a sinusoidal wave with a different frequency. [1], [6], [9], [10], [11], [12] Figure 2 (top) shows different frequencies for different notes.



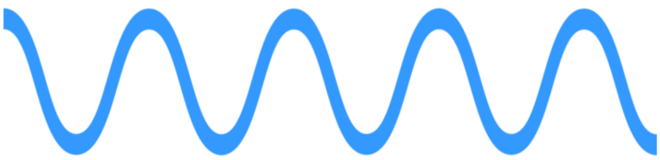


Figure 2. Sinusoidal shape and frequency.
Source: Own authorship.

Figure 2 (bottom) shows the shape of the note sinusoidal waves. The amplitude of the signal is the sound intensity. Voice singing or acoustic instruments produce analog signals with sound energy, as shown in Figure 3.

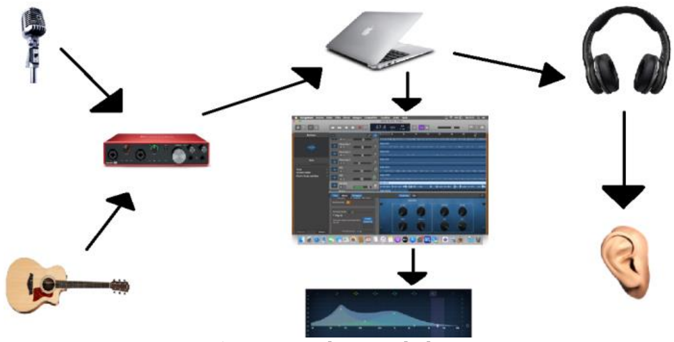


Figure 3. Studio Signal Flow.
Source: Own authorship.

The microphone is a transducer that transforms sound energy into electrical energy. The digital interface digitalizes the analog electrical signal and amplifies the generated digital signal. Digital software mixes the digital signal with equalizers and plug-ins. An equalizer is a set of digital filters that selects and adjusts the intensity of sound of each frequency. [1]

The Fourier Transform transforms the signal representation at the time domain into the frequency domain. The human ear hears the sound as bands of frequency. Low frequencies are listened to with lower intensity. Plug-ins add effects to the digital signal by changing frequency and timbre. Some examples of plug-ins are Chorus, Reverb, and Echo. The digital signal is compressed and encoded as MP3, AIFF, WAVE, or others. [1], [9], [13], [14]

Digital audio can be listened to using audio players with headphones or speakers. Different devices such as smartphones, tablets, and computers have built-in audio players inside of them nowadays. The audio player's functionality is designed according to their application.

This paper will present the design and development of an embedded system for audio reproduction with local and remote control using an Arduino micro and the C language. The device is physically built with PCB fabrication. Section 2 will present the material and methods, and Section 3 will present the digital audio encoding techniques. Section 4 will present the design and development of the embedded system for audio reproduction. Section 5 will show the results and discussion, and Section 6 will drive the conclusions.

The importance of the paper is that it makes the relationship between music and electronics using signal flow flux to present in detail the steps of the development of audio players. Therefore, the study of this paper makes it possible to understand players' inside behavior and data exchange. The paper may be used as a reference to design different players´ architectures.

2. Materials and Methods

Initially, the author studied the concepts of signal digitalization, compression rate, and digital audio formats. Secondly, the author developed an essential audio player using different digital and analog electronic components connected to an Arduino Microcontroller. The firmware was written in C and uploaded using the Dev C compiler.

The essential player can read a set of songs from an SD card and reproduce them in an audio output such as a headphone or speaker. The order in which the songs are reproduced follows the order in which they are displayed inside the SD.

Then, the author developed local and remote controls for the player. The local control of the device was developed by integrating switches and resistors into the system and changing a few lines of code. In this way, the user can control the player by interacting with the device's hardware.

The remote control was developed integrating a Bluetooth module into the system and making new software changes. In this way, the user can control the device from a MacBook Pro by Bluetooth.

Later, the author developed a Mac Book Pro Interface using the software Xcode. In this way, the user controls the player from the MAC Book Pro intuitively.

After the audio player was working on protoboards, the circuit layout was designed on Printed Circuit Boards using the software P-CAD and assembled using the iron removal technique.

After the PCB Design, the author designed a 3D box using the software Sketch Up. The 3D box makes the device more resistant to mechanical collision and can be used to hold the device and make it easier to transport.

Finally, the author corrected software errors using hardware and software tests to make the system more robust and improve its behavior.

3. Digital Audio

The way the sound appears in nature, it is an analog signal with continuous time because its amplitude can assume continuous values over time, and the signal is defined for any value in time. After being captured, the analog signal is generally converted to a digital signal in discrete time. Digital signals are less sensitive to noise, can be processed in a computer, waste less power, and can be compressed. Beyond that, digital systems are more robust to errors and have cheaper data storage when compared to analog systems. [6], [15]

Digital signals with discrete time are signals whose amplitude assumes discrete values, and the signal is only defined for discrete values in time. The general digitalization of an analog signal involves three steps: Sampling, Quantification, and Encoding. In the Sampling step, the continuous-time signal is converted to discrete; in the Quantification step, the signal's amplitude is conditioned to assume only discrete values; and in the Encoding step, the signal takes another representation to reduce its storage space. The main coding techniques are Source Coding, Channel Coding, Errors Resilience, and Errors Occultation. [6], [16]

The most straightforward signal representation is the Pulse Code Modulation (PCM), a binary signal generated by Sampling, Quantification, and Encoding that always uses the same number of bits on each sample. [16] The compression rate is defined as the ratio between the number of bits of the PCM notation signal representation and the number of bits of a more efficient coding technique signal representation:

$$Compression\ Rate = \frac{bits\ in\ PCM}{bits\ in\ another\ technique} \quad (1)$$

Table 1 represents examples of audio formats and their compression rate and applications.

Table 1. Digital Audio Formats characteristics. [15]

Audio Format	Compression Rate	Application
NICAM	<2	Stereo Digital Sound in Analog TV
MPEG 1 layer 1	4:1	DCC
Apt, X100 APT	4:1	DTS, News Radio
ATRAC Sony	5:1	SDDS, Mini DISC
MPEG 1 layer 2	6:1	CAB
MPEG 2 BC	6:1	DVB , DVD
AC3 Dolby	6:1	HDTV, DVD, Dolby Digital
MPEG 2 layer 3	≥7:1	MP3, Internet
PAC AT &T/Lucent	≥8:1	Satellite
MPEG 1 AAC	≥8:1	Satellite, Internet
MPEG 4 T/F Audio	≥9:1	DVB

The MP3 encoding is made using a hybrid analysis structure combining a 32-band filter bank of pseudo-QMF (Quadrature Mirror Filter) with a Modified Cossin Discrete Transform (MDCT) [2]. The MP3 file structure or frame comprises a Header, a CRC, a Side Information, Primary and Auxiliary Data. [16]

The header is composed of 32 bits and has information about synchronization, MPEG version and layer number, CRC mode, transmission rate and sampling frequency, overshoot bit, privacy bit, stereo modes, legal authorship, and manufacturers. The CRC can stay in activated or deactivated modes. If activated, 16 bytes are added to the frame to detect the data most prone to errors. [16]

The side information has information about the last data bit, transmission scale factors, grans localization and auxiliary information, frequency spectrum coefficients, Huffman table tables, and amplification. The primary data are scale factors and decoded Huffman bits using side information. [17], [14] The auxiliary data are optional and have extra information like song title, artist, and gender. [18]

4. Digital Audio Embedded System Development for Audio Reproduction

Initially, the basic audio player configuration was developed in a breadboard using the C language, an Arduino Micro microcontroller, MP3 and Midi Breakout Board for VS1053, a 5 to 3.3V voltage divider (HEF4050-BP), an audio connector (TRS 3.5mm), an 8 GB Micro SD Card and a default USB cable. The firmware executes the following steps in Figure 4.

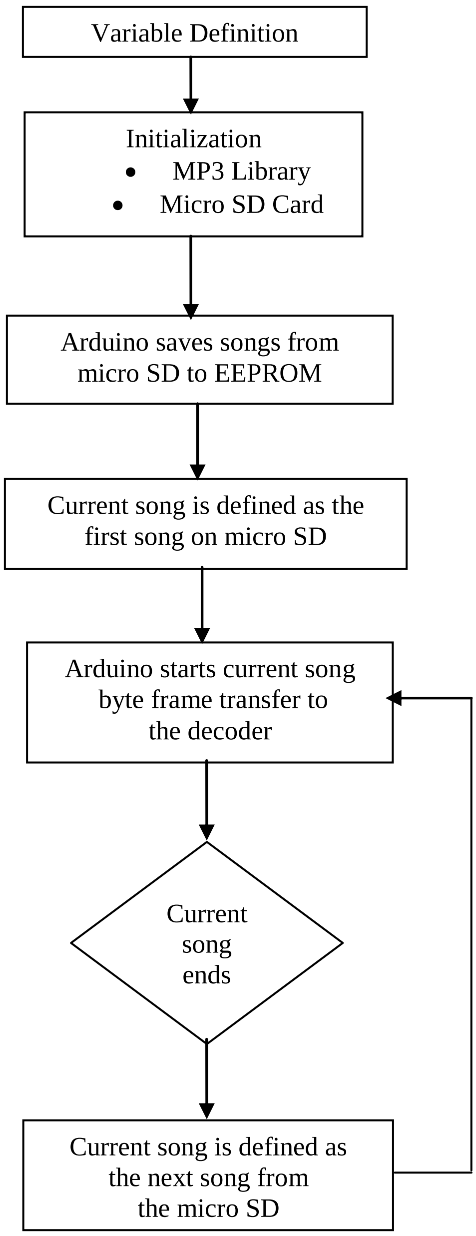


Figure 4. Basic Audio Player Code Flux.
Source: Own authorship.

First, the Arduino defines its variables and initializes the MP3 library and micro-SD card. Then, the Arduino reads the SD card and saves the songs from the micro-SD card to its EEPROM memory. Later, the Arduino current song is defined as the first song of the micro-SD card. Then, the Arduino starts to transfer the current song byte frame data to the decoder. When the current song ends, the current song is defined as the next song from the Micro SD card. After that, the Arduino starts the current song byte frame transfer to the decoder again.

At this stage, the player can read a set of songs inside the SD Card and play them all following the order they are displayed inside the SD successively and uninterruptedly until the Arduino is disconnected from power.

The local control was developed adding five switches and 5 10K Ohm's resistors. Each resistor is connected in series with the switch and powered by 5V from Arduino. Then, an Arduino input pin is placed between the resistor and the switch. Using this configuration, the Arduino Input receives 5V when the switch is open. Conversely, when the switch is closed, there is a voltage drop at the resistor, and the Arduino Input receives 0V. The Arduino is triggered after a voltage change is detected. The first switch plays or pauses songs; the second goes back to the previous song; the third goes to the next song; the fourth decreases volume, and the fifth increases volume.

The remote control was developed by integrating the RS232 TTL HC-05 Module Serial Bluetooth RF Transceiver into the main device circuit to perform wireless communication between the MacBook and the Arduino. The software was implemented using the Arduino Software Serial function. This function replicates the UART serial communication. The circuit assembled in the breadboard is shown in Figure 5.

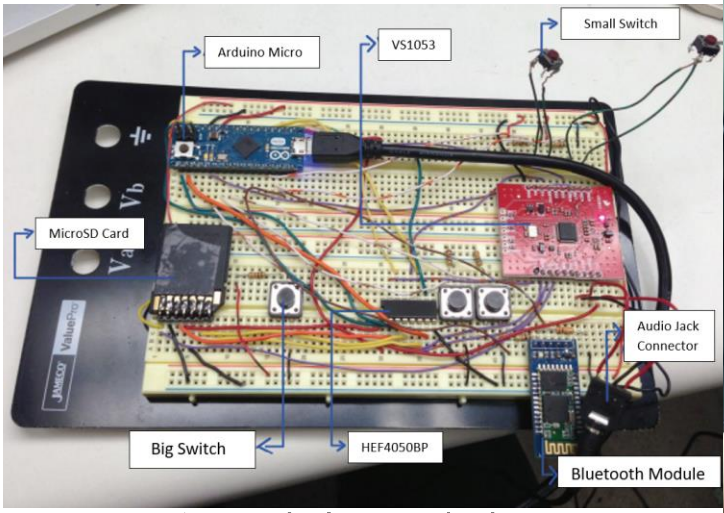


Figure 5. Audio Player on Protoboard.
Source: Own authorship.

Figure 5 shows all the components of the audio player and their names mounted on a breadboard and connected by wire. The Arduino micro is being powered by a MAC Book Pro.

The interface of communication between the user and the audio player was developed using the Xcode software in the Objective C language. It was designed to facilitate the user experience. The digital interface sends control characters to the Arduino through Bluetooth.

The circuit was assembled physically through the connection of two PCBs. The PCB design was done from the layout design and routing model using the software's P-CAD Pattern Editor and P-CAD PCB. The space criteria was making the audio player fit inside one average pant jeans pocket (18x12x10cm). The first PCB is shown in Figure 6.

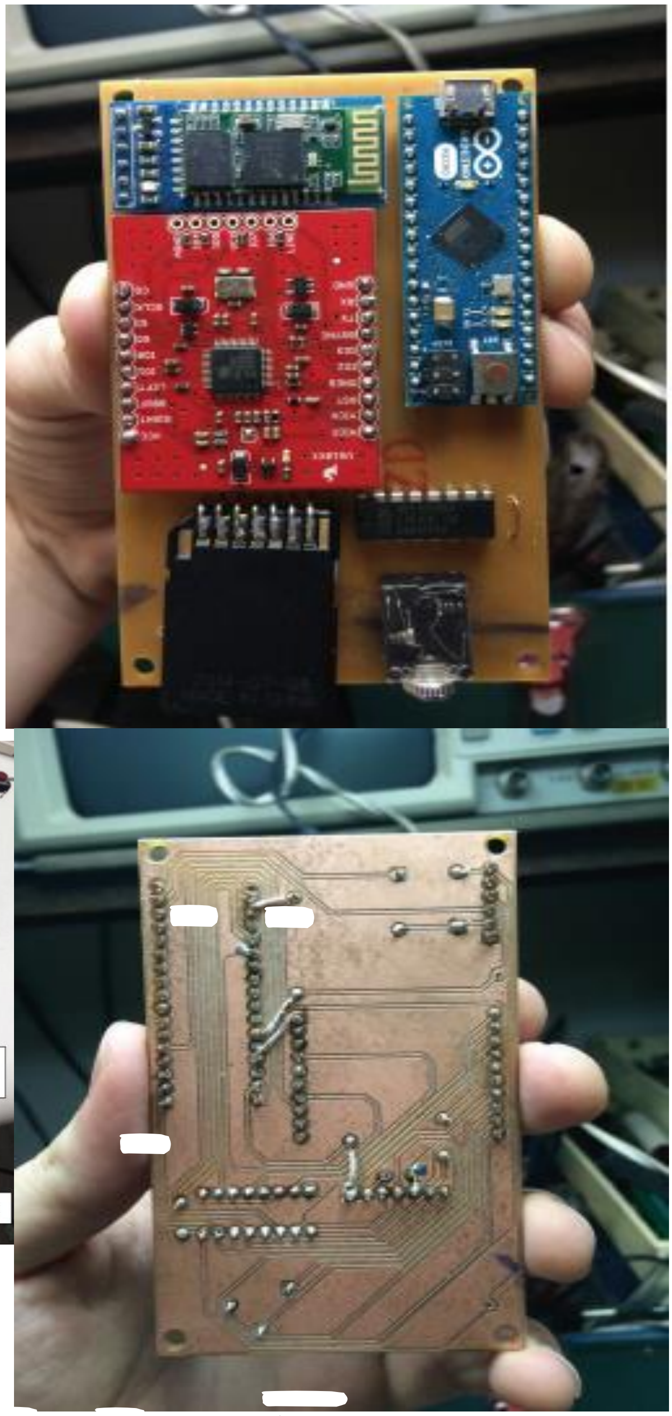


Figure 6. PCB 1.
Source: Own authorship.

Figure 6 shows the two sides of the first PCB assembled. The first side, at the top of Figure 6, has the Arduino micro, VS1053 decoder, Bluetooth module, SD card, Voltage Divider CI, and Audio Jack soldered to the PCB. The other face at the bottom of this figure shows the tracks that replace the wires in PCBs. The second PCB is shown in Figure 7.

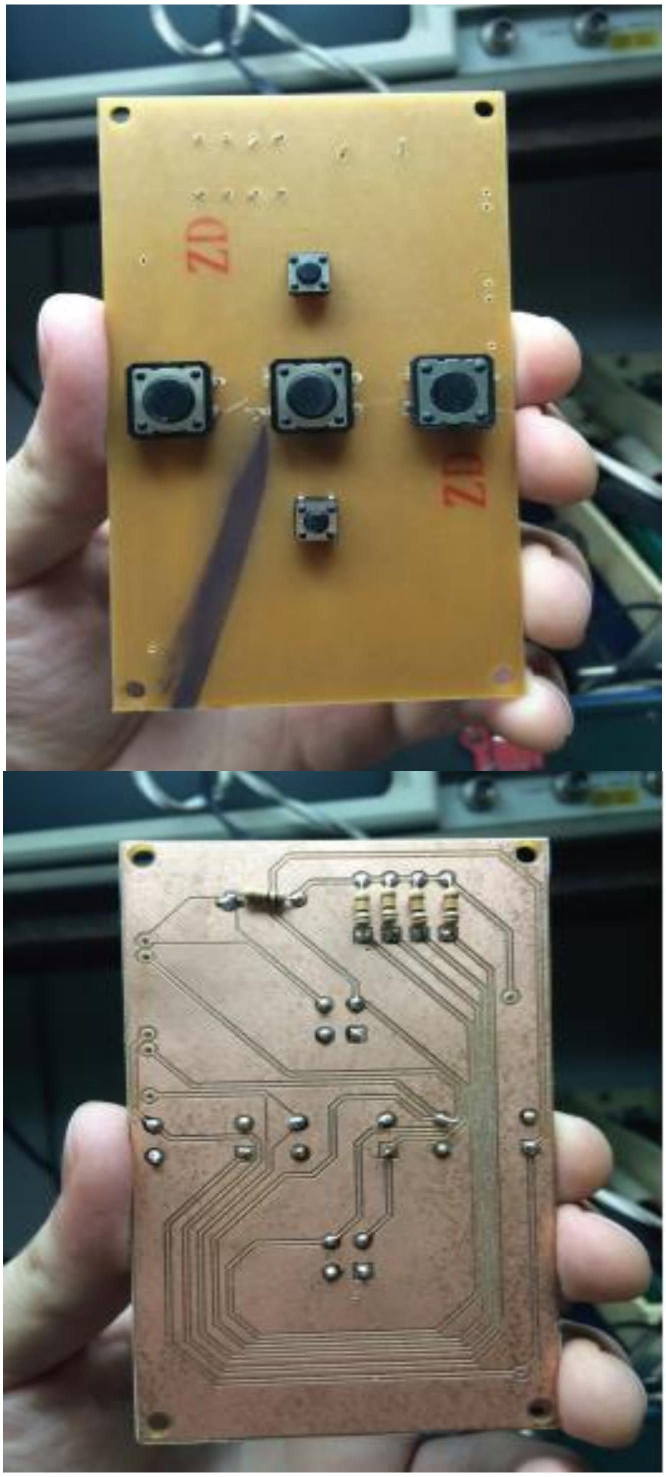


Figure 7. PCB 2.
Source: Own authorship.

Figure 7 shows the two sides of the second PCB assembled. The first side, at the top of it, has the three big switch buttons and the two small ones soldered at the board. The other face at the bottom of this figure shows the current tracks of the PCB 2.

In the interest of connecting one PCB to the other, four ceramic tubes were used, four crews of 1/8 (3.2 mm diameter), four nuts of 1/8, and wires.

In the interest of developing a PCB case to hold the audio player, a 3D box was designed using the software SketchUp. Initially, the dimensions of the PCBs were measured. Then, the PCB case was designed following the criteria of covering all the PCB's area except for the hardware that must be accessed by the user (5 switches, the output to the audio jack connector, the Arduino power output, and the micro-SD Card); the thickness size of the sides should be less than 1cm. The PCB case design is shown in Figure 8.

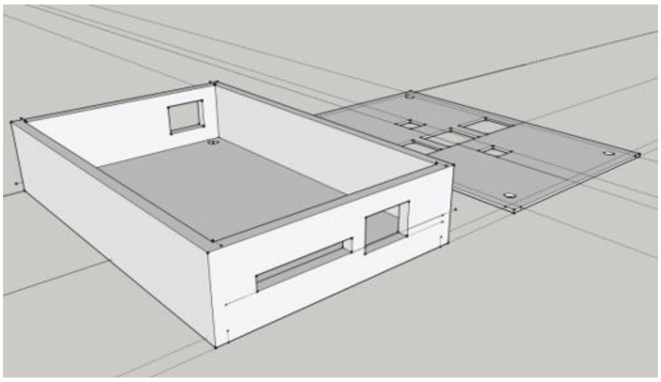


Figure 8. 3D box design.
Source: Own authorship.

Figure 8 shows the 3D box modeled to fit the player inside of it. On the left side, it shows most of the case and its holes for the audio player headphones connection, Arduino power and SD removal. The left side of the figure shows the top of the box with the holes for the switch buttons.

The last development step was performing hardware and software tests to avoid errors and improve software quality. Correcting errors related to the switches and reducing code lines from the main code source were possible.

5. Results and Discussion

By the end of this work, an embedded system was designed and developed from the integration of different digital and analog electronic components controlled by an Arduino Micro microcontroller. The dimensions of the final system are 9.9x6.1x1.8 cm. The final system is shown in Figure 9.

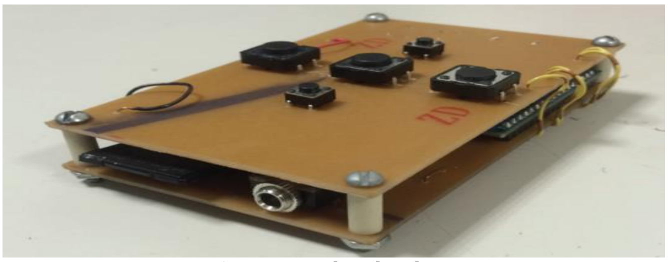


Figure 9. Final Audio Player.
Source: Own authorship.

Figure 9 shows the final audio player that consists of the two PCBs connected by screws and nuts with electronics components soldered.

In order to start the audio player, the user must connect the Arduino output cable to a 220V power outlet. The Arduino will be initialized, and the songs that are inside the SD Card will start to be reproduced. The user can listen to the songs using headphones or a speaker.

The device can be controlled locally or remotely. The local control is made by pressing switches integrated into the hardware system. The user can listen to different songs present on the SD card and decide which songs he wants to listen to by either pressing the big switch buttons in Figure 9 to change the actual song that is playing to the next or previous ones and play or pause songs. Also, by pressing the small switches in Figure 10, the user can increase or decrease volume.

The remote control is made via Bluetooth through a digital interface on the Mac Book. In order to start the remote control of the device, the user should use the compiler DevC with loop() and setup() functions and configure the serial port dev/tty.HC-05-DevB. After that, the user should use the serial monitor and launch the MAC Interface Application.

If the user decides to use the interface on MAC Book, he should use the buttons “>>” and “<<” to change the actual song to the next or previous one, press the “Play/Pause” button to play or pause songs and press the “+” and “-” buttons to increase or decrease volume.

If the user wants to replace the device songs, he must take the micro-SD from the system, connect it to a micro-SD adapter, and connect it to a MacBook.

Therefore, audio player design and development are hard because they rely on comprehensive electronics knowledge. This work shows a profound vision of players' working principles and architecture. It shows in detail the integration of analog and digital electronic components using microcontrollers and how to develop different ways of control. It also shows the possibility of developing a GUI and assembling PCBs.

The knowledge shown in this work may be used to develop new audio player architectures and similar devices with new functionality and user cases.

6. Conclusions

Music can generate brain activity and change mood. It can be perceived as notes or as electronics and physics. Music perceived as notes has a melody and a harmony depending on how one sings or plays an instrument. On the other hand, music perceived as electronics can be captured in nature using electronic devices and converted to digital formats to be listened to using audio players.

The digitalization of an analog audio signal generally follows three steps: Sampling, Quantification, and Encoding. Digital audio has different compression rates.

This paper presented the design and development of an audio player with local and remote control with a Digital Interface using VS1053 MP3 and Midi Breakout Board, HEF4050-BP CI, TRS 3.5 mm Audio Jack connector, 8 GB Micro SD Card, USB cable, switches, resistors, RS232 TTL HC-05 Bluetooth RF Module Serial Transceiver, MacBook Pro and PCB’s.

Initially, the author developed and tested an essential audio player in a breadboard using an Arduino micro to read songs from an SD card and send song frames to VS1053 Decoder.
The local control was developed with different size switch buttons connected to Arduino micro input pins. The remote control was developed using an RS232 TTL HC-05 Bluetooth RF Module Serial Transceiver and Arduino Software Serial function.

The digital interface was developed on a Mac book using the Xcode software to facilitate the user experience on the audio player remote control.
The audio player was physically assembled in PCBs using the software’s P-CAD Pattern Editor and P-CAD PCB. A 3D box was designed using the software SketchUp to hold the player and protect it against mechanical collisions.

The study of this work may be used as a reference to understand the behavior of the audio players inside in detail. Using the paper guidelines, it is possible to develop new architecture, variations of the player using other microcontrollers, different ways of control, new interface designs and add new features.

Limitations and Future Work:
Analyzing the results and the player working behavior, it was possible to propose a few changes to be performed later: print the PCB case designed in this work using a 3D printer; generate a malfunction signal in case of device error; develop the song transfer from the MacBook to the device using Bluetooth; develop the functionality of seeing the SD music titles from the MacBook GUI; develop songs frequency control.

Data Availability
All data in support of the findings of this paper are available within the article.

Conflict of Interest
The authors declare that there are no conflicts of interest.

Funding Source
The author provided all the funding sources.

Authors’ Contributions
Marcelo Moreira made all the paper development stages, including literature research, methodology, design, development, and writing.

References

[1] Joana Kwiecien, Pawel Skrzynski, Wojciech Chmiel, Andrzej Dabrowski, Bartlomiej Szadkowski, Marek Pluta, "Technical, Musical, and Legal Aspects of an AI-Aided Algorithmic Music Production System," *Applied Sciences*, Vol.**14**, **2024**.

[2] Zhejing Hu, Gong Chen, Yan Liu, Xiao Ma, Nianhong Guan, Xiaoying Wang, "Make a song curative: A spatio-temporal therapeutic music transfer model for anxiety reduction," *Expert Systems With Applications*, Vol.**240**, **2024**.

[3] Anantha Narayanan, Mareshwar Naidoo, Victor Kong, Lydia Pearson, Kevin Mani, James P. Fisher, Manar Khashram, "Broad Responses and Attitudes to Having Music in Surgery (The BRAHMS Study): An Australia and Aotearoa New Zealand Perspective," *Surgery Open Science*, Vol.**17**, pp.**30–34**, **2024**.

[4] Sayali Bhandarkar, Bhagyashree V. Salvi, Pravin Shende, "Current scenario and potential of music therapy in the management of diseases," *Behavioural Brain Research*, Vol.**458**, **2024**.

[5] Ingrid Erhardt, "The power of music: Psychoanalytic explorations," *International Forum of Psychoanalysis*, Vol. **33:1**, pp.**66-68**, **2024**.

[6] Christof Weiss, Meinard Muller, "From Music Scores to Audio Recordings: Deep Pitch-Class Representation for Measuring Tonal Structures," *ACM Journal on Computing and Cultural Heritage*, Vol.**17**, Issue.**3**, pp.**1-19**, **2024**.

[7] Jessica Beatriz Tolare, Mariangela Spotti Lopes Fujita, Fabiano Ferreira de Castro, "Musical score representation and retrieval in digital environments," *International Federation of Library Associations and Institutions Journal*, Vol.**50(2)**, pp.**408-415**, **2014**.

[8] Pal S., Singh V. P., "A Predictive Voice-Input and Output Alternative Communication System using MFCC & DTW," *International Journal of Scientific Research in Computer Science and Engineering*, Vol.**4**, Issue.**3**, pp.**7-10**, **2016**.

[9] Chunyan Zeng, Shixiong Feng, Zhifeng Wang, Yuhao Zhao, Kun Li, Xiangkui Wan, "Audio source recording device recognition based on representation learning of sequential Gaussian mean matrix," *Forensic Science International: Digital Investigation*, Vol.**48**, **2024**.

[10] Nidhi Chakravarty, Mohit Dua, "A lightweight feature extraction technique for deepfake audio detection," *Multimedia Tools and Applications*, Vol.**83**, pp.**67443-67467**, **2024**.

[11] Chunyan Zeng, Shuai Kong, Zhifeng Wang, Kun Li, Yuhao Zhao, Xiangkui Wan, Yunfan Chen, "Digital audio tampering detection based on spatio-temporal representation learning of electrical network frequency," *Multimedia Tools and Application*, **2024**.

[12] Shweta Sinha, "Message-Driven Generative Music Steganography Using MIDI-GAN," *International Journal of Scientific Research in Computer Science and Engineering*, Vol.**3**, Issue.**5**, pp.**1-5**, **2015**.

[13] Zhaopin Su, Guofu Zhang, Zhiyuan Shi, Donghui Hu, Weiming Zhang, "Analysis and Recognition of Dialects of Hindi Speech," *Journal Of Latex Class Files*, Vol.**14**, **2015**.

[14] Salma Masmoudi, Maha Charfeddine, Chokri Ben Amar, "MP3 Audio Watermarking using calibrated side information features for tamper detection and location," *Multimedia Tools and Applications*, Vol.**83**, pp.**64637-64662**, **2024**.

[15] Lisarte Cristovao Mendes Barbosa, "Uma abordagem ao áudio pela perspectiva de ensino," *Instituto Superior de Engenharia do Porto*, **2012**.

[16] Fernando Pereira, Sergio Faria, Carlos Salema, Pedro Assuncao, Anibal Ferreira, Isabel Trancoso, Paulo Correira, "Comunicações Áudio Visuais: Tecnologias, Normas e Aplicações," *Instituto Superior Tecnico*, **2009**.

[17] Rafael Korycki, "Authenticity examination of compressed audio recordings using detection of multiple compression and encoders' identification," *Forensic Science International*, Vol.**238**, pp.**33-46**, **2014**.

[18] Nasser Kehtarmavaz, Namjin Kim, "Digital Signal Processing - Level Design Using Labview," *University of Texas at Dallas*, **2005**.

AUTHOR PROFILE

Marcelo Moreira is a powertrain automotive software analyst at KPIT on behalf of Stellantis. He received his BS in electronic engineering from the Federal University of Campina Grande, Brazil, the University of Nebraska – Lincoln, and Stanford University, U.S. 2016. He received his MS in Computer Engineering at the Universidade da Beira Interior, Portugal 2021. His current working areas are automotive design, embedded systems, and tests.

