Chromesthesia Simulation Device

Angel Garcia, Nicholas Alban, Brooke Roeder, Wesley Ellery

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — Chromesthesia is an audio-visual variant of the phenomenon called Synesthesia. Synesthesia is when the simulation of one sensory or cognitive pathway leads to involuntary experience in a second sensory or cognitive pathway. The purpose of this project is to create a device that represents this phenomenon, where a user will vocalize sounds at different frequencies, and have that sound associated with a certain color. The device will also show on a display, sourced from a camera module, the location of where this sound is being vocalized from. Successful demonstration of this project is aimed at giving awareness to this phenomenon, which affects 1 in 3000 people.**

***Index Terms* — Algorithms, software, simulation, embedded systems, power amplifiers, microphones.**

I. Introduction

This project is to design a device that will take audio and visual inputs and produce a visual output that simulates/recreates the visual experiences that a person with chromesthesia would experience. The way it will work is by combining audio input devices (i.e., microphones) and an optical capture device (i.e., a camera) on a single platform that can be oriented and pointed in any direction. This platform will receive audio and visual inputs from the direction it is pointing, and then perform a computation process to recreate the same imagery from the visual input edited with visual effects influenced by the audio input. The visual output will be displayed to a screen for the user to view. The platform containing the audio and visual input devices, the PCB/computational hardware, and the display are all integrated together into one device that the user can carry and thus, aim the microphones and camera in any direction and receive a corresponding visual output, as if looking around with their own eyes and experiencing chromesthesia.

Chromesthesia is an audio-visual variant of the phenomenon called "Synesthesia". Synesthesia is when the stimulation of one sensory or cognitive pathway leads to involuntary experiences in a second sensory or cognitive pathway. Chromesthesia is the involuntary association of sounds to visual effects such as shapes and colors. The design for this project will incorporate many different components including software and hardware which will work closely together to perform the desired task.

The software is extremely important in communicating to the hardware so that everything functions accordingly. The components picked for the project will be chosen based on cost, efficiency, and availability. Due to COVID and certain shortages, we must take availability of certain components into consideration because the pandemic caused components to take longer to ship or just be completely unavailable.

The goal is to make a user-friendly device that is efficient, compact, and inexpensive. It should be comfortable to wear and should only be where the user has to look around and aim with their head to operate the device. The device should be portable and easy to operate for the user. The current design idea is to incorporate some sort of microphone array and a camera to collect audio and visual inputs. These inputs will be fed into a microcontroller unit on a printed circuit board. The microcontrollers will perform logic and other audio and image processing functions to then produce one single output in the form of an image which is identical to the input image that the camera captured, except that it is altered and edited with visual effects that were influenced by the audio inputs. Depending on the audio inputs, variances in frequency and location will determine the visual effects. This will produce an experience that simulates the phenomenon of Chromesthesia. although many people experience Chromesthesia with varying degrees of difference when questioning each individual person who has the Phenomenon with their brain, this device aims to find a general neutral ground that can portray the phenomenon in a way that is not necessarily accurate, but entertaining and accurate enough to convey the concept to the people who are using the device. Chromesthesia is extremely hard to exactly recreate or simulate, but we can educate people about the phenomenon and provide them with the experience of it by creating an experience that shares parallels with the natural phenomenon.

II. Project Goals

The main objective of this project is to design an efficient consumer device that can simulate the chromesthesia phenomenon, so people can see what it is like for synesthetes. Secondary goals would include a low-cost design and good product longevity, allowing for a low-end pricing for the average user. The following will be the main ideas and concepts that will guide our design choices. They will constantly be referenced and used to determine important design choices such as what component to buy and what functions should be prioritized. In order to simulate Chromesthesia, our device needs to be accurate in determining audio frequencies and be able to determine the exact location of the audio source. Any inaccuracies will cause incorrect visuals and disrupt the recreation of the phenomena.

Processing of the audio inputs will be especially important. The device needs to produce an image in real time. Ideally, the time between the user aiming the device and the device creating an output image will be as short as possible to simulate "looking" with their own eyes. This means the processes need to be fast and efficient. This will be heavily dependent on the software optimization and hardware design and quality. The software must be able to communicate with the microprocessor and be able to perform complicated algorithms including noise processing, 3D sound localization, and image processing.

These algorithms must be as quick and efficient as possible to determine what color correlates to the audio frequency and how it should be displayed and produce a near real-time picture. The hardware will consist of a microprocessor, multiple microphones, and a camera able to display images of quality 720p or higher. These will all be the key components to the Chromesthesia device. The hardware should be chosen carefully to optimally perform the calculations needed to process audio and visual inputs so the device can operate at the desired speed

Long term considerations will include a Virtual Reality component, however due to time and budget constraints, this is better suited as a long term idea.

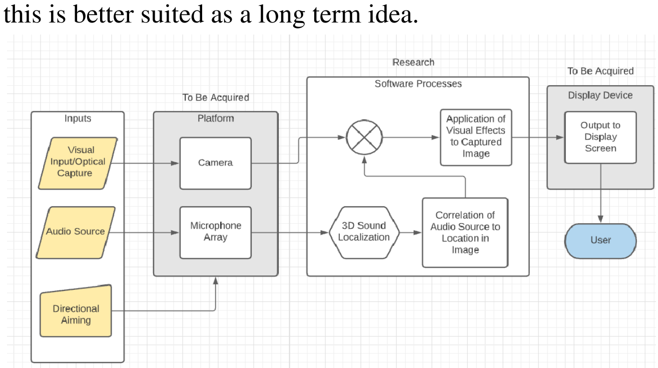


Fig. 1. Device Block Diagram.

III. Design overview

For this project, there are two basic features; one is the ability to process audio inputs with a microprocessor and perform a basic algorithm that can display colors correlating to the frequency of the soundwaves from the audio input. The second basic feature is the ability to capture visual images in real time, parallel with detecting and processing the audio inputs. As can be seen in the following figures, there are two separate instances. These examples provide an outlook as to what the desirable results would be. The two systems, i.e. the audio processing and the visual processing units are both operating at the same time, but they are independently collecting data and Performing tasks. This is the minimum functionality of the device. Without this function, the device simply could not work or deliver on its more advanced features.

PURE SONOCHROMATIC SCALE		
(invisible)	Ultraviolet	Over 717.591 Hz
	Violet	607.542 Hz
	Blue	573.891 Hz
	Cyan	551.154 Hz
	Green	478.394 Hz
	Yellow	462.023 Hz
	Orange	440.195 Hz
	Red	363.797 Hz
(invisible)	Infrared	Below 363.797 Hz

Fig. 2. Pure Sonochromatic Scale. A color is associated with a certain frequency.

SONOCHROMATIC MUSIC SCALE (basic 12/360)		
	Rose	E
	Magenta	D#
	Violet	D
	Blue	C#
	Azure	C
	Cyan	B
	Spring	A#
	Green	A
	Chartreuse	G#
	Yellow	G
	Orange	F#
	Red	F

Fig. 3. Sonochromatic Scale. A color is associated with a musical note.

A. Requirements

Design requirements are critical to the success of any project, and the following section details those limits and standards. We were able to deal with those requirements appropriately by using an engineering design method. For the success in the creation of this project, we will go over the safety and technological requirements.

Safety considerations are important in both design, and demonstration of our device. While the focus of this device will mostly be software related, we need to consider the fact that faulty software can lead to unintended consequences, especially with a design based around heavy computations. One such issue can be the overheating of our Microprocessor. Depending on the severity, this can cause a fire hazard or injury if touched. Another safety consideration is making sure we keep to IEEE standards, as we will be using microprocessors and various other electronics with this device. Failure to follow these standards can result in dangerous exposure to electricity.

In order to simulate Chromesthesia, our device needs to be accurate in determining audio frequencies and be able to determine the exact location of the audio source. Any inaccuracies will cause incorrect visuals and disrupt the recreation of the phenomena. Processing of the audio inputs will be especially important. The device needs to produce an image in real time. Ideally, the time between the user aiming the device and the device creating an output image will be as short as possible to simulate "looking" with their own eyes. This means the processes need to be fast and efficient. This will be heavily dependent on the software optimization and hardware design and quality.

The software must be able to communicate with the microprocessor and be able to perform complicated algorithms including noise processing, 3D sound localization, and image processing. These algorithms must be as quick and efficient as possible to determine what color correlates to the audio frequency and how it should be displayed and produce a near real-time picture.

The hardware will consist of a microprocessor, multiple microphones, and a camera able to display images of quality 720p or higher. These will all be the key components to the Chromesthesia device. The hardware should be chosen carefully to optimally perform the calculations needed to process audio and visual inputs so the device can operate at the desired speed.

Our long-term vision for the project is for it to be a pair of goggles or any other similar item that you can wear, and one would be able to see through them and experience Chromesthesia. Alternatively, the device will be handheld and aimed with the user's hands like a camcorder. Both designs demand ease of use with minimal inputs from the user (i.e., the user only needs to aim the device with their head or hands) and a light weight (ideally under 5 lbs). While this is not part of the initial prototype requirements, the design consideration has helped make this device as compact as possible for the demonstrated prototype.

The device should adequately represent the phenomenon of chromesthesia, and the final visual output should have a semblance of the experience, as if "through the eyes" of someone with chromesthesia. Chromesthesia varies between people, and it has nuances in how it is experienced. People should be excited to use the device and want to experiment aiming it at different sound sources with different pitches.

B. Software Design

The audio processing procedure involves noise cancellation, fourier transforms, 3D sound localization, application of frequency identification, receiving an input from the camera module inputs and using this to determine a correlation between 3D sound localization and the pixels on the display captured by the camera, and finally an association between the detected frequency and the respective color paste on our Chromesthesia algorithm. Beginning with the capture of an image, a single frame is collected with the camera, and it is sent to the MSP430. A function determines the size of that image communicating the width and height of every frame a retrieve and sends this to the audio handling MSP430 chip on our PCB. Once the audio processing on our PCB completes this process, it sends its data back to the visual processing on the Raspberry Pi. The visual processing then uses the data to redraw the image with the desired visual effects and apply it. Then, it will send this new image to the display and output each frame.

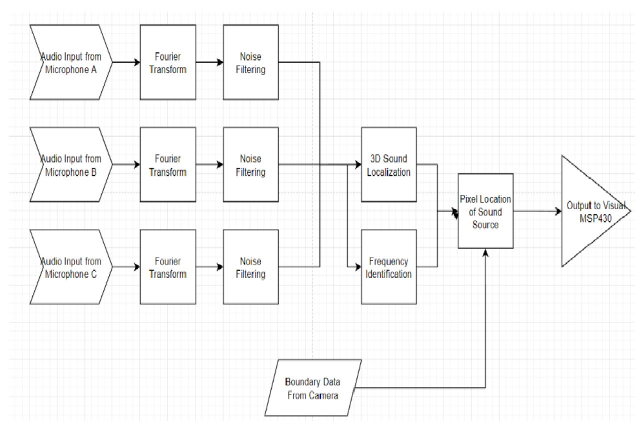


Fig. 4. Audio Processing Software Flowchart

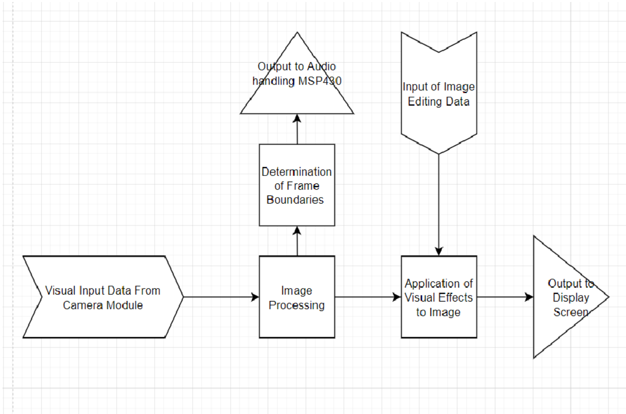


Fig. 5. Visual Processing Software Flowchart

C. Algorithm Design

The frequency-to-color algorithm developed is based off of Case 12 from the study "Musical pitch classes have rainbow hues in pitch class-color synesthesia" and the color scale created by D.D. Jameson in the book "Colour-Music", where colored keys are used to create a sound-music color scheme. Luminance/lightness of the color is then chosen based off of the octave of the key. The "middle" octave of a piano is the 4th octave, so the lightness for that value is set to 0.5 (range is between 0 and 1). The lightness increases by 0.08 as the octave increases and decreases by 0.08 as the octave decreases from the "middle" octave. Lower octaves have a lower pitch and frequency, while higher octaves have a higher pitch and frequency, so darker and lighter colors are then correlated to these octaves. The resulting color scheme can be seen below

Octave #	Luminance												
1	0.18												
2	0.26												
3	0.34												
4	0.42												
5	0.50												
6	0.58												
7	0.66												
8	0.74												
9	0.82												
10	0.90												

Fig. 6. Correlated Octaves

The following equations convert frequency to the numbered key on an 88 key piano. This equation is also inverted:

$$f = 2^{\frac{k-49}{12}} \times 440\ Hz \tag{1}$$

$$k = 12log_2(\frac{f}{440}) + 49 \tag{2}$$

Offset this by 9, so at note C at octave is n=0:

$$n = k + 8 = 12log_2(\frac{f}{440}) + 57 \tag{3}$$

To identify pitch class, which is the note in an octave, as well as the correlating hue and brightness, we use the following equations as part of our algorithm.

Frequency to key number:

$$n = 12log_2(\frac{f}{440}) + 57 \tag{4}$$

Pitch-class to hue equation:

$$h = \begin{cases} 20p, & 0 \le p < 2 \\ 10p + 20, & 2 \le p < 4 \\ 60p - 180, & 4 \le p < 5 \\ 45p - 105, & 5 \le p < 6 \\ 30p - 15, & 6 \le p < 7 \\ 45p - 120, & 7 \le p < 8 \\ 30p, & 8 \le p < 9 \\ 15p + 135, & 9 \le p < 11 \\ 60p - 360, & 11 \le p < 12 \end{cases} \tag{5}$$

Octave to lightness equation:

$$l = 0.8n + 0.18 \tag{6}$$

D. Embedded Software

Two different microcontrollers are used for handling processing, one for audio processing and the other microcontroller will be responsible for handling image processing and other image related functions. The reason behind this is because audio processing and image processing are both known to captivate rigorous processes. Therefore, our team decided it would be beneficial to split up the workload between two microcontrollers. The microcontrollers will be interacting with these different inputs by running code and logic on them. Different software will be running in parallel with each other, with the audio processing being the primary code that had to be developed. For this to work effectively, both microcontrollers must be working in tandem with each other and be on the same steps of the process to ensure that they produce the correct and desired output. We utilized the development environments provided with each board to develop our Embedded code.

E. Microcontrollers

A microcontroller is an integrated circuit that controls some or all of the functions of an electronic device or system. It includes a microprocessor, memory, and associated circuitry. It's a compact microprocessor designed to operate the functions of embedded systems, as well as any product that can calculate or display data. A microcontroller consists of a processor, non-volatile memory, a clock, an input/output control unit, and more. Due to these many factors, microcontrollers can be divided into different categories in relation to their specifications. The original goal was to create a simple board that would allow us to work quickly and worry less about potential problems. Each component was chosen with care. The most important things that we need to consider when choosing a board include Processor Speed, available RAM, and sufficient peripherals. If our algorithm does not have proper resources to conduct the calculations, we will run into issues throughout usage. Therefore, CPU and RAM will be carefully analyzed and compared between the various Microprocessors. We also need to consider more than one may be used, in order to facilitate integration of the various amount of processing that are occurring, from the image processing to the interpretation and locating of the sound. To add on to this, other factors that must be considered when doing research on microcontrollers is whether they are single board chips, or microcontrollers. The difference between them would be their processing power and seeing if it aligns to our needs for our device. The MSP430 is a Microprocessor that supports many functions. This can be programmed using C and C++ and is designed to be integrated with outside controllers. The MSP430 contains several optimizations related to power consumption and is ideal if power in our system becomes a concern. The CPU contained on this particular board is a 16-bit CPU. This microcontroller has multiple registers that can take data for input for high volume applications and still give a high calculation performance. The MSP430 family of architecture is well documented, and the datasheet provides enough information that we can choose this for our device. In order to meet the requirements, we have specified, the MSP430 must meet several computation and protocol standards in regard to integration of a camera and microphone. The Raspberry Pi 4 is a small, compact microprocessor that has the functionality of a personal computer. This microcontroller can run several different compilers, from Linux, C++, Java and Python. The Raspberry is a well-known, and well utilized microprocessor that has extensive documentation available. The Raspberry Pi 4 contains an HDMI port that we can output any process through, making it ideal for displaying what we process through it. In addition, it contains four USB ports, and GPIO pins for attaching components. The 64-bit architecture of this system is ideal, since we require processing and RAM usage of a 64-bit system. This is essential for the graphics processing we will be doing.

F. Printed Circuit Board

The PCB is one of the most important components for our Chromesthesia Simulation Device. Performing algorithms and other software functions to achieve the desired experience that we want the users to have. The device is heavily based on collecting visual input data and audio input data, and both of these processes are notorious for being very demanding computationally, so we decided to use a microcontroller unit, which is the MSP430. To design our board, we needed to pick a PCB design program that we would use, and ended up using KiCad. In addition, the printed circuit board also includes connections to the microphone array and the camera module.

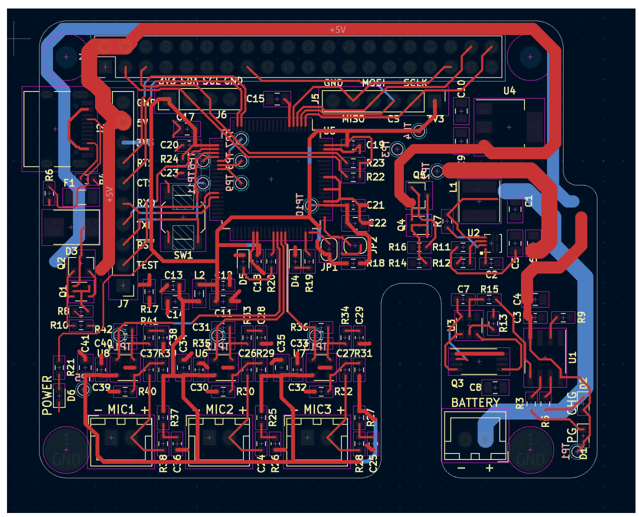


Fig. 7. PCB Board file.

Wiring for the power supply also runs through the PCB. Power delivery is extremely important, especially since there will be two microcontroller units doing very demanding copy stations constantly for prolonged periods of time.

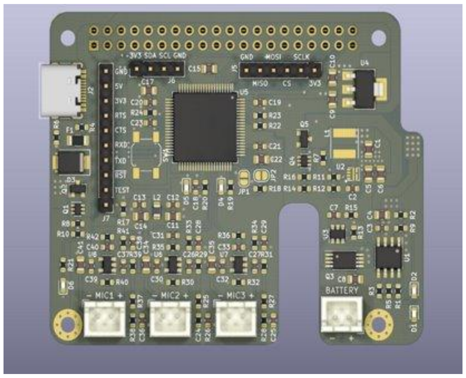


Fig. 8. Render of the Chromesthesia Simulation Device PCB.

G. Camera Module

The Arducam Mini camera is a 2 Megapixel, high-definition camera module. This particular camera utilizes the I2C protocol, which is beneficial to connecting it to our Microprocessor. Since we need a clear, precise image, the Arducam Mini is the camera we selected for this project. In addition to its 2 Megapixel sensor, this camera provides adequate resolution per our requirements at 720p, enough to display on our screen. Power consumption in low power mode is 5V at 20mA, helping keep our device efficient.

H. Microphones

When choosing our microphones, we needed one that could detect audio signals with a narrow cone. The reason for this is because we want the device to work in a directional manner, for the 3D sound localization to work properly. We do not want a microphone that can detect audio signals with a large cone. This would result in detecting signals beyond the scope of the camera's aperture. Ideally, we would like to use a microphone that has a cone similar to that of the camera's visual collection, however it is understandable that this might be hard to achieve, especially considering the low-budget nature of our microphones that we will be choosing from. In this situation, it would be preferable to have a microphone whose cone is larger than the camera rather than narrower. It would be easier and more convenient, and cater towards a more enjoyable user experience, if we had to limit the size of the microphone's reading ability rather than that of the camera. if we wanted to cut off the cameras, this would involve reducing the resolution and pixel output. The ramifications of this decision, if we did it, are not explored. However, we can foresee this being a complicated issue to handle. Therefore, it would be more desirable to have microphones that have a cone larger than that of the camera rather than a camera with a column larger than that of the microphones. We have identified several microphones that we can use as potential components for our device. For each of them we briefly explore them and share some information about them. In the end, we selected the Adafruit I2S Microphone as the proper solution. When considering microphone usage for this device, several factors need to be analyzed when selecting a proper device that can properly localize the sound and meet our requirements. Our microphone should be able to adequately interpret and pick up the voices of our test subject, while minimizing the impact from outside sound sources nearby. In addition, the selected microphone should be taken and after extensive research, it was determined that the Adafruit I2S Microphone is a suitable choice. The I2S standard will complement our Microprocessor, as this microphone uses digital audio, instead of analog. The Raspberry Pi and MSP430 both use the I2S standard, so integration of this microphone was the correct solution.

I. Power Supply

Lithium-ion batteries are rechargeable batteries that are commonly used to power gadgets that require a large amount of electricity. These batteries were developed in the 1970s and first commercialized in Japan in 1991. They are now used to power cellphones, computers, electric vehicles, and other electronic devices. Furthermore, as these batteries' use has grown in popularity, their safety has improved, and they now contain fewer contaminated metallic elements than other batteries. Lithium-Ion batteries that are 1kg can generate up to 150 watt-hours of electricity. We can see how a lithium-ion battery can be useful because it can be compact while still storing a considerable quantity of energy. When completely charged, a typical lithium-ion battery has a voltage range of 3.0 to 4.2V and a lifespan of two to three years, or 300 to 500 control cycles. Our team decided that utilizing this as our power supply was crucial, to meet our requirement of being compact and still have enough energy to power our components.

J. Future Development

Our Chromesthesia Simulation Device has the ability to incorporate additional features, as these were part of the design considerations we set. Due to the extreme scarcity of parts and long delivery times, certain features that were researched and considered were not included for this initial functional prototype. Instead, the groundwork is available for these to be added. One such example is a virtual reality feature. Simply put, we could utilize all our algorithms, and use libraries provided by a virtual reality headset to see where the frequency and color will be coming from.

IV. Testing

Prototyping the audio processing unit and ensuring the capabilities of audio and frequency detection was one of the first and primary stress points that we tested, to ensure that we had a desirable reading. To test this, a test was set up in a similar fashion to the black box nature that was described above. This demo feature was to essentially make a spectrum analyzer that can display and read out the audio inputs of the microphone using the MCU we used. The audio detection and processing of the audio inputs from the microphone is one of the most important baseline tests that can be accomplished with this final stage of development. The reason is because all of the technical complexity of the system and the device as a whole depends strongly on the audio processing aspects of this device. The audio processing steps involve everything from the level of the audio inputs from the microphone itself to the utilization of that collected data in the software residing in the MCU. The first and most basic test was to ensure that the microphones are able to detect varying frequencies of sound, that they are able to convey this data to the MCU via pin connections, and that the MCU is able to use this data in real time and output values based on those inputs. To ensure that these three levels of the audio processing work, we would need to conduct a black box test to determine if a system that is set up for this specific function using the exact hardware we intend to use would indeed be able to perform these processes from end-to-end.

Fig. 9. Display with color associated with frequency.

The output data collected from this was checked and measured in real time, in direct response to audio input signals collected from the microphone. The data was observed by the operator and the data was compared to a predetermined target value of the frequency measurement. To ensure this was accurate, a sound was played into the microphone of a predetermined frequency. There are many resources online that can generate frequency noises that are open and available to the public. We generated a frequency pitch based on an input value. We could range the frequency to 20,154 Hz to 1 Hz. For the sake of the test, we would select a specific frequency, i.e. 440 Hz and would play the audio output through speakers placed nearby the microphone

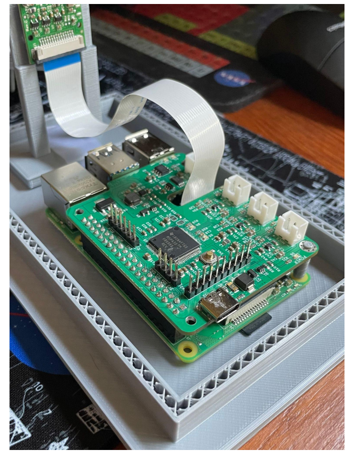


Fig. 10. Finished PCB on top of Raspberry Pi.

The PCB was tested in the Senior Design lab to verify the microphone signal was being amplified. After verification, we took our design and connected the microphones and camera module to our board. One of the more difficult issues we faced in testing was getting our 3D sound localisation to work properly. After many hours of trying to work through issues, we determined that due to hardware limitations of our chosen hardware, sound localization would be too slow to implement on this initial prototype. Our budget was heavily thrown toward getting our PCB created, as our design required extensive components for our microphone amplifier.

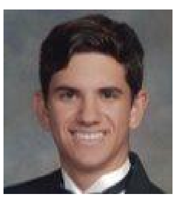
VIII. Conclusion

In conclusion, our Chromesthesia Simulation Device was a functional prototype that worked as expected, but with some setbacks. Complicated algorithms and design solutions were used during development, but with only University knowledge and a limited budget, there was only so much development that could be done. However, there is plenty of room for more efficient development and design in the future. Bringing awareness to this phenomenon was a success, as we feel that we have provided a device that can accurately showcase how people worldwide see their environment around them. Our attempt at creating 3D sound localization was not successful due to hardware limitations, however the research and algorithms we developed should allow for implementation when we get a budget to utilize better hardware.

V. The Engineers

Wesley Ellery is a graduating Computer Engineering student at the University of Central Florida. Wesley has an interest in using his skills he learned at the University of Central Florida in the aviation industry. After graduation, he will be working full-time at Collins Aerospace as a Systems Engineer, working primarily on integration of avionics on the Comac C919 airplane. Wesley is also a Private Pilot and holds a Seaplane Rating.

Brooke Roeder is a graduating Electrical Engineering student. She currently works as an intern at Lockheed Martin in systems engineering, supporting the testing and implementation of sensor technology. In addition to supporting the conductions of tests, she has also automated and developed scripts used for image processing. She is currently transferring to the electrical engineering department at Lockheed Martin and will begin to work in hardware design.



Nicholas Alban is a graduating Electrical Engineering student. He is currently interning at General Dynamic Mission systems, supporting live service and development of government contracted military simulation and training software and frameworks. He's taken a few classes related to microcontrollers and circuit boards, and many courses related to programming, machine learning, and computer vision. These skills are anticipated to be useful and strongly applicable when it comes to applying algorithms and complex functions to the board itself.

Angel Garcia is a graduating Computer Engineering student at the University of Central Florida. Angel has an interest in using his skills he learned at the University of Central Florida in the cybersecurity side of the technology industry. After graduation, he will be working full-time at Ultimate Kronos Group as a Product Support Specialist, working primarily on product management, product engineering and operations that support a Global Support team.

Acknowledgement

The authors wish to acknowledge the assistance and support of the Senior Design instructors, Dr. Samuel Richie and Dr. Lei Wei, of the College of Engineering and Computer Science at the University of Central Florida.

References

[1] Calkins, Mary Whiton. "A Statistical Study of Pseudo-Chromesthesia and of Mental-Forms." *The American Journal of Psychology*, vol. 5, no. 4, 1893, pp. 439–64, https://doi.org/10.2307/1411912. Accessed 14 Apr. 2022.

[2] "MSP430FR6989." *MSP430FR6989 Data Sheet, Product Information and Support | TI.com*, TI, https://www.ti.com/product/MSP430FR6989.

[3] ISO. (2021, 10 14). ISO/IEC 14882:2017. Retrieved from iso.org: https://www.iso.org/standard/68564.html

[4] "ECMA-334." *Ecma International,* 27 Nov. 2020, https://www.ecma-international.org/publications-and-standards/standards/ecma-334/.

[5] "Sound Color Project." *Sound Color Project,* https://soundcolorproject.com/.

[6] "SPI Explained." *SPI Explained | Dev Center,* developer.electicimp.com/resources/spi

[7] "Getting Started with STM32 - I2C Example". *DigiKey Electronics Belgie,* https://www.digikey.be/en/maker/projects/getting-started-with-stm32-i2c-example/ba8c2bfef2024654b5dd10012425fa23.

[8] Goss, Clint. "The Color of Sound." To, https://www.flutopedia.com/sound_color.htm\

[9] "How I2C Works (I2C Explained Simply)." *The Geek Pub*, 19.Aug2021,https://www.thegeekpub.com/18351/how-i2c-works-i2c-explained-simply/.

[10] "ISO/IEC 13816:1997." *ISO,* 1 Oct. 2007, https://www.iso.org/standard/22987.html