Handheld Color Sliding Game

EEL 4915: Senior Design 2

Group Number: Group 24

Group Members:

Josh Bell - Computer Engineering
Eric Espinosa - Electrical Engineering
Linus Fountain - Electrical Engineering

Customers / Sponsors / Significant Contributors:

Matthew Gerber - Computer Science
Wei Sun - Electrical and Computer Engineering
Masoumeh Kalantari Khandani - Electrical and Computer Engineering

Chapter 1: Executive Summary

In navigating the dynamic landscape of gaming, our group has observed the shifting desires in consumer preferences, with a specific focus on the evident decline in standalone handheld game devices. The recent statistic indicating an 11.5% reduction in sales of these electronics over the past year serves as a poignant marker of the challenges faced by this market [6]. This decline is, in part, attributed to the gravitational pull of alternative gaming platforms, with a substantial portion of the target audience moving towards more immersive experiences offered by video game consoles such as the Nintendo Switch or PlayStation. Nevertheless, during this shift, our team draws inspiration from the collective nostalgia associated with handheld devices like the iconic Bop It from our own younger years. This shared nostalgia fuels our desire to confront the prevailing market challenges head-on, driving us to create this handheld color sliding game that not only echoes the sentiment of the past but also resonates with the diverse preferences of contemporary consumers.

At the center of our ambitious goals lies a firm commitment to bringing back to life the handheld gaming experience. We envisioned our final product as a flawless fusion of sleek design, quality mechanical components, and, most importantly, gameplay that strikes the perfect balance between challenge and reward. We look to begin on a journey to redefine the narrative of handheld gaming. Recognizing the multifaceted nature of this undertaking, we establish the foundation through this comprehensive technical paper, outlining not only our guidelines and immediate goals but also charting a visionary trajectory for the evolution of the project. This document serves as a dynamic roadmap, evolving in unison with our progress, adapting to emerging insights, and embracing the iterative nature inherent in the research, development, and design refinement phases.

As this journey unfolded we implemented a dynamic approach. In the initial stage, we dived into extensive research and development, meticulously identifying and acquiring the necessary components to breathe life into our handheld color sliding game. Subsequent phases witnessed the diligent testing of our initial design, a critical step that allowed us to gain invaluable knowledge, address potential challenges, and refine elements in need of enhancement. This iterative process ensured that we remained agile and responsive, consistently striving to elevate the user experience to the highest levels. When we finally converged on our ultimate design, we engaged in a meticulous fine-tuning process, making nuanced adjustments that helped create a product that not only meets but surpasses our own expectations. Through this comprehensive and deliberate approach, we aimed not only to revitalize the standalone handheld game market but to set a new standard for immersive and engaging gaming experiences.

Chapter 2: Project Description

2.1 Project Motivation

Since the start of our group, we've all agreed on one thing – our project should bring joy to consumers and ignite our own passion. All three of us share a background in a tech-driven and ever-evolving gaming era. Growing up surrounded by constant technological advancements and new gaming experiences, we naturally gravitated towards this realm for our project.

Our inspiration stems from our childhood, and our goal was to create a fresh and distinctive game. We've carefully designed our project to be compact, entertaining, and easy to handle, especially for those familiar with modern consoles. We aim to provide not just entertainment but also a comfortable user experience. The design is user-friendly, ensuring people of all ages can easily engage with it.

This project was more than just a creative pursuit; it was a valuable opportunity to assess and expand our existing skill set. As we dove into the development process, we refined our abilities and acquired new ones. The ever-changing nature of our chosen field guarantees a continuous learning experience, pushing us to adapt and innovate. By combining our passion for technology and gaming with consumer enjoyment, our goal was to not only showcase our capabilities but also make a meaningful contribution to an industry that has shaped our lives. This ambitious venture is driven by our shared desire to create something special – a game that resonates with both our childhood memories and the diverse audience we hope to engage.

2.2 Game Rules

When the game commences, eight random colors and directions will be presented to the player on the LCD in a way that visually wraps around the toggle's area of motion. A color on each side near the toggle corresponds to one of these eight arrows and colors, informing the user of their current target. To advance in the game, the user must manipulate the toggle in the direction past the arrow that matches the current assigned color, and in the direction of the arrows. When the user moves the toggle past the correct arrow zone it will be marked as complete and the user will get a point once they have done this on both sides. Then the color on the side will change to the next target color. After the user completes both arrows the game will continue looping until the user runs out of time. The middle of the display will showcase both the remaining time and the user's current score, which will showcase their progress. The main menu will have the option of seeing all high scores. The system has the option of having the game play on both toggles at the same time or a single toggle that the user can choose. Below is a render of a basic design of the sections.

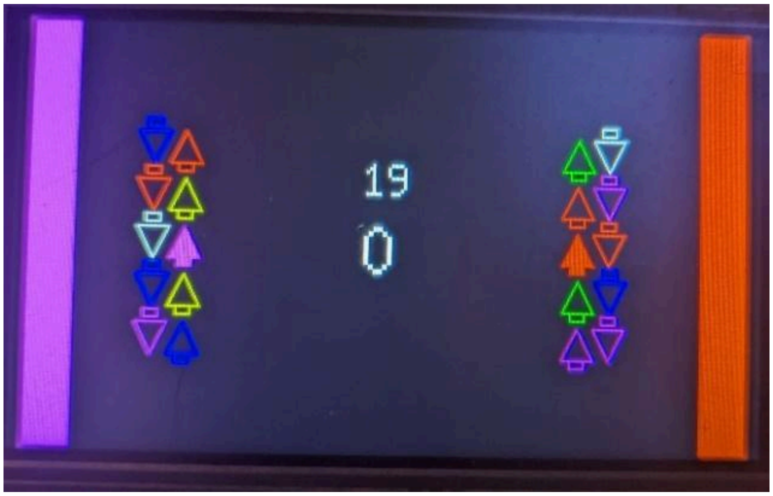

Figure 1 - Render of game rules

Precisely the game rules are as follows: As seen above there will be 10 arrows on each side, five up and five down. Arrows one through ten will each be assigned a random color (duplicate colors may occur).There will be a colored rectangle on either side indicating which color is the target. For example, if arrow three is red, the toggles current color is red, the player must move the cursor from arrow four, to three, to two; and the arrow three would be cleared.

2.3 Project Market Analysis

We have looked at comparable devices that are already in stores such as the games Bop It, Rubiks Revolution, and others that have similar characteristics. While looking at these games and their sales history we have found that similar type games have accounted for 4 billion dollars worth of sales over the course of 2022 which is identical to the previous years sales as well as up half a billion dollars from 2020 [6]. We believe that there is plenty of room for a competing product to join the market as the type of product we are designing only takes up about 7.3% of the current toy market,
leaving only room for growth with the insertion of a new product. Compared to the Bop It that is currently selling between $12 - $44, and the Rubiks Revolution that is currently selling at $20-$22 we think that hypothetically we would be able to sell our product for $35. While this is greatly below our estimated total cost of production we think that if we were to mass produce this there would be much room to bring the total cost of production down well below the $35 price point.

2.4 Project Goals and Design Requirements

Our primary goal for this project was to create and design a completely functional, handheld, mechanical game that promises entertainment and is easy to use for a wide spectrum of consumers. We wanted this device to have a screen consisting of eight sections on each side that will be used for the gameplay itself. On these eight sections we would have colored arrows, each arrow would light up a specific color. For this to be a stimulating and fun experience to the consumer, we wanted to have a central toggle with set zones for each arrow to be able to determine the path the user goes. We also wanted to have a speaker that will play a timer of sorts, so the user would know how much time they have left for each level, and some feedback music that plays once you pass an arrow. Once the user fails, we would have a system to keep track of the high score to provide a challenge for the user, so they can keep pushing themselves to beat their score, or invite a friend to compete with them. We also wanted to have the main screen have a scrolling text system that would allow us to present messages such as “game over” or “new high score!”.

2.4.1 Core Goals

- Creating gaming device
- Thumb stick control
- Speakers
- Display screen to display game
- Making the device portable

2.4.2 Advanced Goals

- Haptic feedback
- Mute option

2.4.4 Functional Requirements

The functional requirements this game project must have is being able to start and run the game. The LCD should provide visual feedback by showing the current color target closer to the toggle, and the targets to hit on the main board. The middle of the LCD screen should display game related information, such as the highest score, a timer, and pause/start, and scores should be updated in real time.

The device has to be user friendly, that is comfortable and easy to use. It should have a functioning toggle joystick that operates for our intended use to control the game and menu. It should be responsive and provide feedback to the user. The game should have a difficulty or progression system. Where from the start you can pick your difficulty, or if level based, then when you beat level 1, you go to 2, etc. This way the game can keep being enjoyed, while also challenging the user to progress and get better.

A nice requirement to include is some type of feedback, whether it be physical, audio or visual. The feedback can include the thumbstick or controller vibrating to let you know when you have passed a color. An audio that plays when you pass a color, or beat a level. Or visual, if we do an LCD screen we can display text on the screen to inform the user of certain actions.

2.4.5 Technical Requirements

We have chosen to use an LCD screen rather than the originally planned LEDs, it is a small 4-5 inch x 1-2 inch screen. The LCD displays the arrows going left and right, as well as give the arrows the color needed for the game. It also displays a type of cursor that moves in the direction you are choosing, to let the user know which direction they are going so they do not get lost. The cursor is updated in real time, without delay to ensure a smooth experience for the user.

The main menu LCD screen needed to display the menu at the start of the game, which must have a start option to start the game, a high score option to check what the highest score is on the device, and a settings option for the game settings. When the game starts, the screen must display the current score of the game, and a timer that counts down to showcase how much time the user has to complete it.
For the microcontroller unit, we were deciding between using something like the MSP430FR6989 or an arduino kit, since we are most familiar with C, C++ and Java. The device will be powered by a rechargeable power source in order to avoid electronic waste of having to replace the batteries every so often. It must have built in speakers and must have a switch to power the device on, and it must have an audio switch as well.

The controller must use non-volatile memory, in the form of an electrically erasable programmable read-only memory(EEPROM) or non-volatile RAM(NVRAM) in order to store score data and high score information. This type of memory is used in arcade games and pinball machines, as it is essential for maintaining records and will save the data even after turning off.

2.5 Related Products

In the realm of interactive devices, our product finds itself among counterparts like the aforementioned Bop-It or Rubik's Cube; however, it is poised to carve its distinct niche as a unique and unparalleled concept. Notably, no other games currently on the market share the innovative features we envisioned for our game. The introduction of a novel idea inherently invites evolution, and we as we anticipated, certain aspects including game rules and physical appearance, underwent slight modifications as we delved into further research and encountered new inspirations and troubles. Despite this adaptability, our primary commitment remained to preserve the integrity of our initial concept, ensuring that any changes implemented served to enhance and refine rather than drastically alter the essence of our original vision.

2.6 Engineering Specifications

The outlined engineering specifications below serve as the backbone of our final design goals, detailing specific criteria we aim to meet. Acknowledging the constraints of time during the product demonstration, we opted to focus on key components that will be showcased prominently: controller feedback, toggle response, and boot up time, all highlighted below. These selected elements represent critical aspects of user interaction and visual experience, offering a small look into what our end goal was. We believe that by strategically emphasizing these highlighted features, we can effectively convey the excellence and uniqueness of our product within the given timeframe, ensuring that the viewer gains a greater understanding of its qualities in the competitive landscape.

Dimensions	3.543” x 7.824” x 1.25”
Projected weight	~ 600 grams
Screen Refresh Rate	60 HZ
Screen Resolution	480 x 27 pixels per panel
Battery life (AA)	**~ 3 hours**
Charge time	~ 7 hours
Boot Time	**1.42 seconds**
Toggle response	**~200 msec**

Table 1 Technical Specifications of device

2.7 Hardware Block Diagram

Regarding the hardware block diagram, the integral aspect of our design revolves around the microcontroller, which serves as the central hub for all input components within the system, as depicted in the diagram provided. The microcontroller plays a pivotal role in not only orchestrating the connections but also in processing and managing the input signals, primarily stemming from the toggle device. The microcontroller's interaction with the input components is a fundamental element of our project's functionality. It acts as the bridge that facilitates bidirectional communication, receiving input from the connected devices, most notably the toggle. This input is then subjected to the execution of our programmed code, which resides on the microcontroller.

The code embedded in the microcontroller operates as the decision-making brain, dictating the course of action in response to the input it receives. It formulates precise instructions that govern the behavior of both the screen module and the speaker. These instructions dictate not only what actions to perform but also the timing of these actions.

In essence, the microcontroller serves as the conductor of an intricate symphony, orchestrating the harmonious interplay between the user's inputs and the tangible feedback provided through the screen and speaker modules. This symbiotic relationship ensures that the user's interactions are translated into meaningful responses, thereby delivering an immersive and responsive experience in line with our project's objectives.

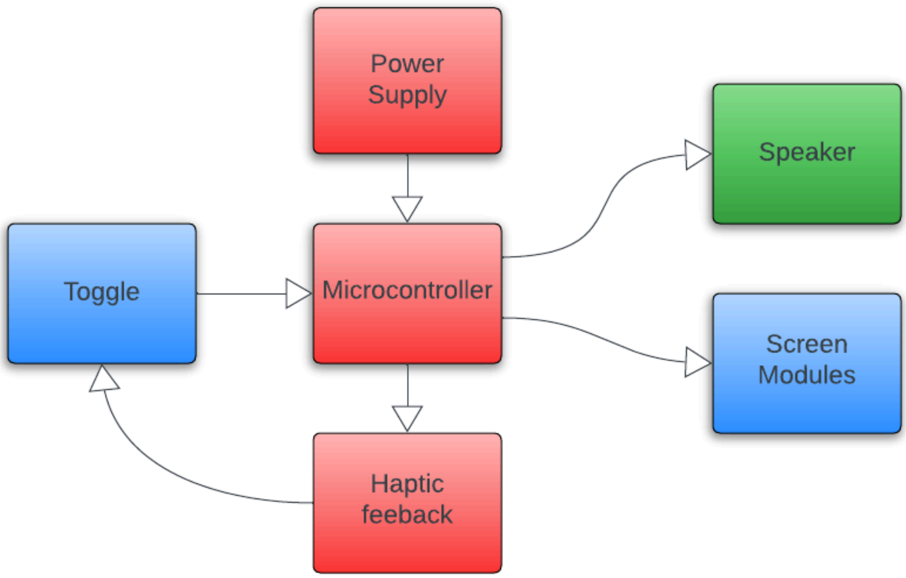

Figure 2 - Hardware block diagram

2.8 Software Block Diagram

Regarding the software block diagram it was envisioned that the software would be designed to operate with different "zones or arrows" to track the user's interactions effectively. This approach was aimed at facilitating a checkmark system, which will be utilized to monitor the user's progress pathing and assist in keeping points.

The core concept revolves around the user's interaction with a toggle, which needs to be moved in the direction indicated by an arrow. Users are expected to navigate the toggle through a series of arrows in a specific order, as indicated by the arrow's direction. The purpose of these zones is to verify if the user successfully passes through them in the prescribed sequence. Once the toggle successfully traverses all three zones in the correct order, a counter will keep track of how many arrows have been cleared.. The completion of a stage is determined by comparing the number of cleared arrows with the number of initially illuminated arrows. When these two quantities are equal, the stage is considered complete. This process will repeat for each subsequent level.

When the timer runs out, the system will initiate a check to determine whether a new high score has been achieved. This is done by comparing the current score with the previously stored high scores. If a new high score is attained, it will replace the previous record, and the system will reset to its initial state, enabling a fresh game to begin. If no new high score is achieved, the system will skip the high score update and proceed with a system reset in the same manner as if a new high score had been established.

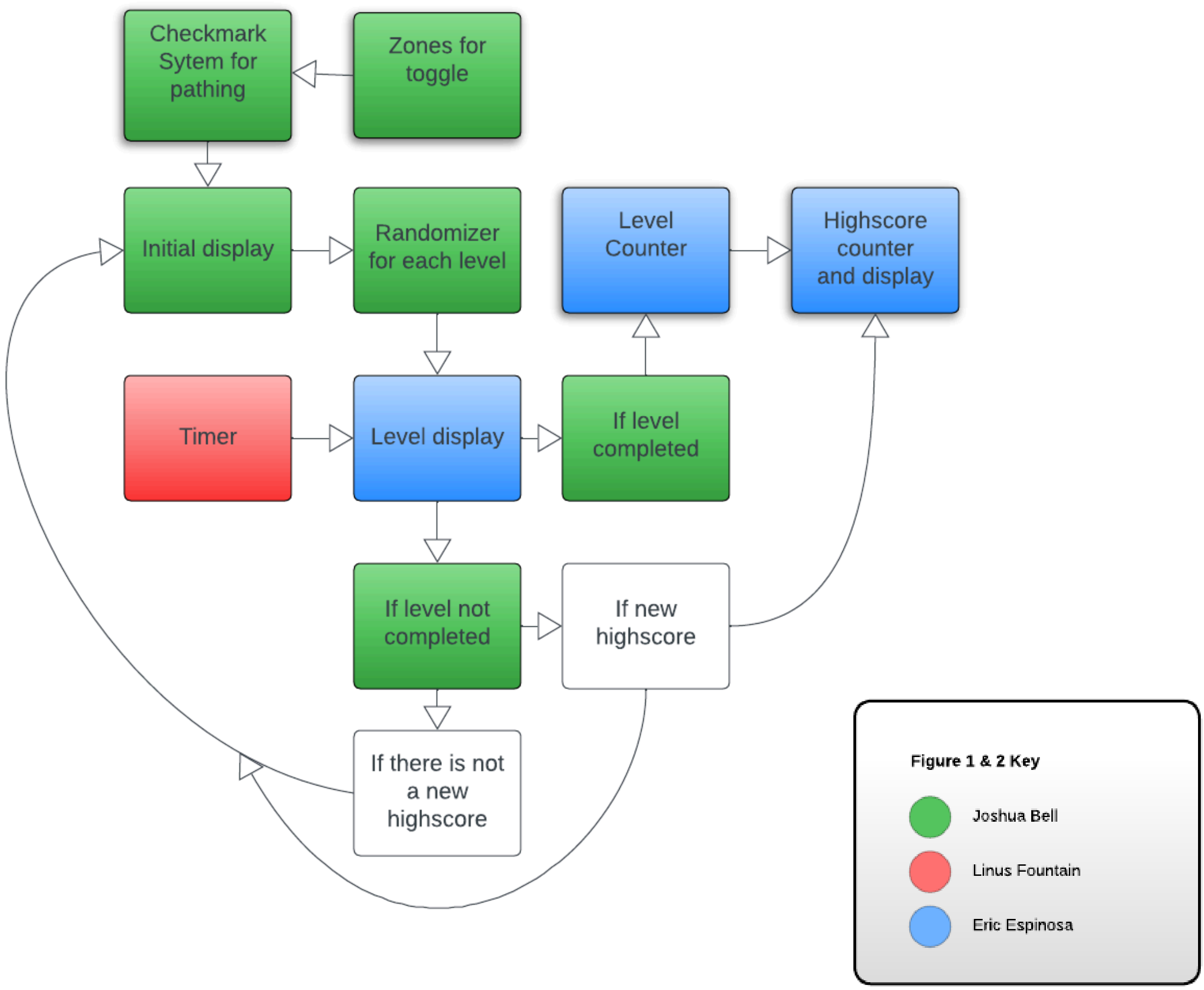

Figure 3 - Break down of all input and output components to be installed

The representation below shows the distribution of responsibilities among our team members, with each individual playing to their respective strengths. Given the varying specializations within the group, those more inclined towards hardware components have naturally chosen tasks of that nature. Conversely, a member with a strong background in software development and programming took the lead in these areas.

Block Description	**Group Member Assigned**	**Block Status**
Power Supply	Linus Fountain	Completed
Haptic feedback	Linus Fountain	Completed
Toggle	Eric Espinosa	Completed
Screen Modules	Eric Espinosa	Completed
Microcontroller	Linus Fountain	Completed
Speaker	Joshua Bell	Completed
Zones for toggle	Joshua Bell	Revised Due to Storage Limitations
Checkmark System - pathing	Joshua Bell	Revised Due to Storage Limitations
Initial display	Joshua Bell	Completed
High score counter and display	Eric Espinosa	Completed
Level Counter	Eric Espinosa	Completed
Level display	Eric Espinosa	Completed
Timer	Linus Fountain	Completed
If level not completed	Joshua Bell	Completed
If level completed	Joshua Bell	Completed
Randomizer for each level	Joshua Bell	Completed

Table 2 - Represents who is assigned to acquire the component or complete the task.

It is important to note that while we had tasks assigned to specific team members, our project was a highly collaborative effort. We recognized that the fluid exchange of ideas, insights, and assistance among all group members was integral to the project's success. This collaborative spirit ensured a well-rounded approach, allowing us to tap into the collective expertise of our team as we worked harmoniously towards our common objectives.

2.10 House of Quality

Below, you can see our House of Quality, which serves as a visual representation of our project's key objectives and priorities. We placed a strong emphasis on ensuring that our device was not only portable but also user-friendly, catering to a broad audience. This is clearly reflected in our prioritization of "ease of use" as our highest-rated standard, with "weight" closely following as a significant factor. However, it's crucial to emphasize that while we prioritized portability and user-friendliness, we remained committed to maintaining the overall quality of the device. Our goal was to strike a balance that allows us to deliver a high-quality product that is accessible and convenient for all users.

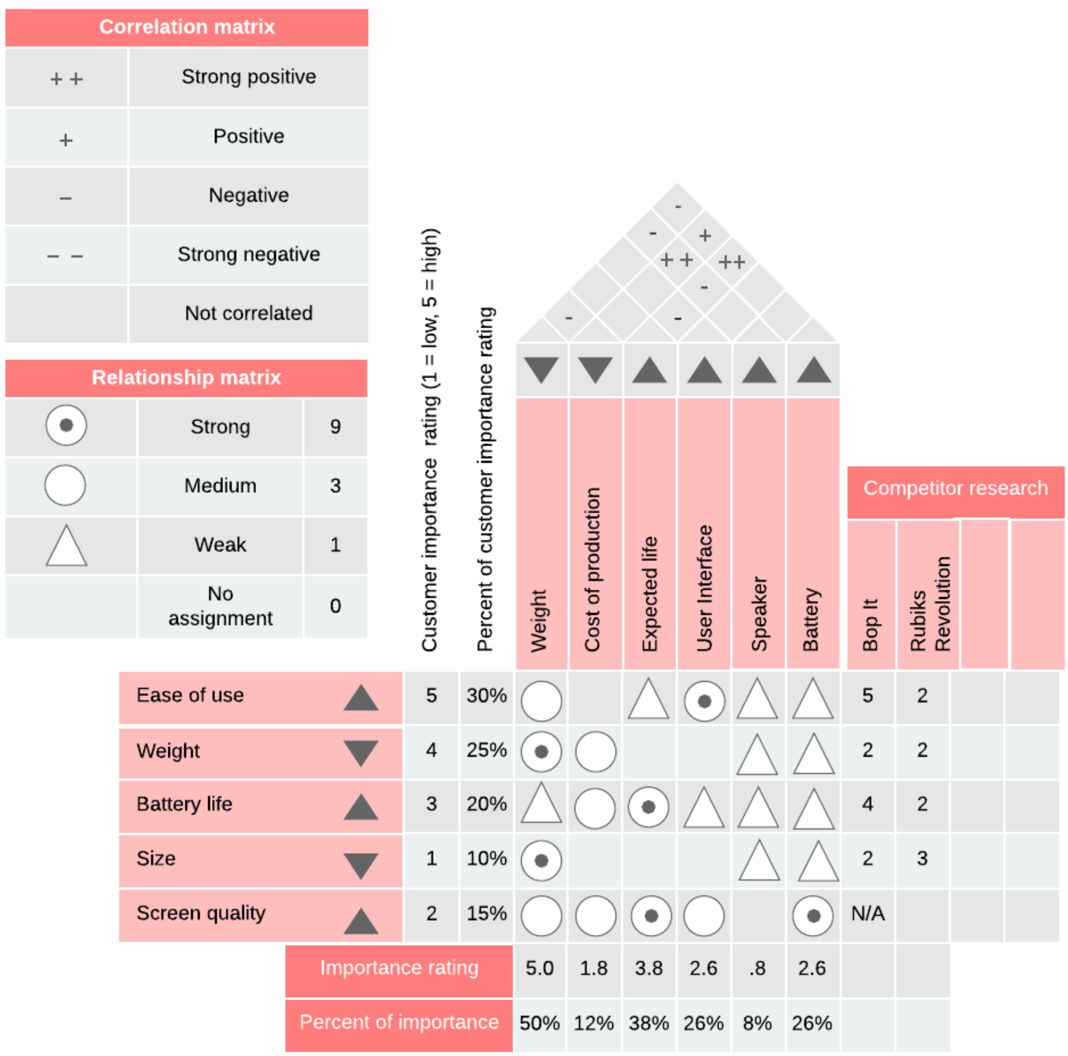

Figure 4 - House of Quality

Chapter 3: Research

3.1 Display

In order to bring our project to life, it was imperative for us to select an appropriate display that will serve as the canvas for showcasing the main gameplay of our game. At the start of this decision-making process, we found ourselves presented with a plethora of options, each with its own unique advantages and considerations. Therefore, the following section will go into the collective thought process of our team, outlining the extensive research that did ultimately guide us toward making a well-informed choice regarding the type of screen that is most likely to maximize the chances of success for our project. By thoroughly exploring the available display alternatives and carefully weighing the pros and cons associated with each, we aimed to ensure that the display we ultimately selected aligns perfectly with our project's specific requirements and goals.

3.1.2 Display Progression

When deciding what type of display to have for this project we looked into a few different possibilities for what type of display we will be using. We have had the idea of going extremely basic and having just single led lights under a clear plastic top that would be in the shape of an arrow. This would have worked well and be extremely cheap however we realized we would have issues with colors bleeding into one another. We also felt that while this would be a cost effective option we thought it would look cheap which we wanted to avoid.

We then moved onto the idea of having small individual screens that we could program to have arrows on each screen. This would work by having all of the screens programmed together so that they could display the game properly. However we wanted to have the option to show scrolling text on the screens so this did not prove to be a viable option.

Another consideration we had when deciding which display to use was we wanted to have the display be curved. We thought that this would provide a clean look to the user and allow for a unique look of the project to make our product standout. However, we quickly realized that a curved display that would fit the exact dimensions we desired is not readily available so we shifted our focus. Not being able to have the curved screen led us back to having one large screen that would contain all of the display, or having multiple smaller screens connected together.

3.1.3 Universal-Solder Electronics Ltd 1502

When considering the integration of smaller screens for our project, we explored several options. The most economical choice among them is the Universal-solder Electronics Lcd 1502 2 x 16 Blue-white LCD screen, which comes at a budget-friendly price of

$3.66 per unit. With dimensions measuring 80.00mm x 36.00mm x 11.00mm, we estimated that we would require approximately 4-5 of these screens, resulting in a total screen length ranging from ~352mm to 440mm and ~$14.64 to $18.30.

A noteworthy aspect of this option is that each team member already possessed this particular screen from previous coursework, contributing to potential cost savings and expediting the project timeline. However, upon further exploration into the details of this display, it became obvious that it may not be the most suitable choice for our project.

Connecting multiple displays to one another and integrating them into the system as a whole is unnecessarily complex. This process is complicated, especially when compared to simply opting for a single, larger display. Despite its affordability and our team members' existing ownership of the screens, the decision to employ these smaller displays could potentially hinder the project's efficiency and design.

Moreover, the output quality of the Universal-solder Electronics Lcd 1502 may not align with the project's requirements. While the display excels in embedded systems and presenting data or information, its performance may fall short when it comes to smoothly rendering a game at the desired frames per second. Essentially, the limitations of this display may compromise the overall gaming experience we aim to deliver, making the pursuit of an alternative, larger display a more viable and practical choice for our project. [14]

3.1.4 Blue LCD Module Controller HD44780

Exploring alternative options, we've delved into the features of the 16x2 Blue LCD Module Controller with the HD44780, specifically engineered for compatibility with Arduino and Raspberry Pi. The advantage lies in its seamless integration with our Arduino-based project, making it a potentially fitting choice for our display requirements. However, it's worth noting that this particular screen carries a slightly higher price tag of $5.99 per unit, a factor to consider, especially if we anticipate the need for multiple screens. Additionally, it boasts slightly larger dimensions, measuring 80mm x 36mm x 15mm.

This display shares the same characteristics as the previously mentioned option, which means it is a display that is designated to control an alphanumeric LCD module. The display is primarily designed to showcase letters and numbers, rendering it ideal for embedded system or information display. However, this does not meet our project needs, and is less optimal for our intended use. Therefore, despite its compatibility with Arduino and raspberry pi, there are more versatile alternatives better suited to our project. [12]

3.1.5 RGB LED Matrix Panel

Recognizing the limitations posed by smaller LCD displays in terms of both size and functionality, our exploration into suitable display solutions for our project led us towards larger screens. Given that our project centers around a color-based game, we looked into the idea of utilizing an RGB LED Matrix panel as a potential candidate, since it has a capacity to showcase a spectrum of colors across its numerous LEDs, arranged strategically on a board. The ability to program the colors of individual LEDs opens up possibilities for creating intricate patterns and designs, aligning seamlessly with our original concept for the game. The one we specifically looked more into was found on Amazon, its dimension is 192mm x 96mm, which is vastly larger than the previous two options. However, the cost of this matrix panel is a steep price of $33.

While initially considering the use of a matrix panel for our project, we ultimately decided against it as our project and design evolved. Recognizing the need for an enhanced user interface and the ability to display dynamic menus, our focus shifted towards incorporating an LCD screen that would better align with our evolving goals.

The matrix panel, characterized by individual LEDs spread across a board, presented potential challenges in delivering an optimal visual experience. Given the dynamic nature of our project, where the game commences and colors are shifting, relying on a matrix panel could result in a less-than-ideal viewing experience. The scattered nature of the LEDs might make it challenging for users to follow the game seamlessly.

In our goal to deliver a user-friendly and enjoyable interface, we prioritized a display solution that could offer superior visual clarity and a more cohesive user experience. By opting for an LCD screen capable of presenting a UI and facilitating menu displays, we aimed to enhance the overall interaction for users, ensuring that they can fully enjoy the project without the potential frustration associated with a less sophisticated display. [6]

3.1.6 4.3” TFT Display

The Adafruit 7 inch TFT display is an LCD marvel employing thin-film transistors (TFT) to elevate image quality and responsiveness, essentially serving as a miniature monitor tailor-made for our project. This option outshines its predecessors by a considerable margin. Boasting a pixel density of 800 x 480, the display delivers a visual experience that is leagues beyond our prior considerations, and is slightly more economical than the matrix panel at $34.95. This display is also a touch screen, a delightful addition that, while not a necessity for our project, holds potential for future utilization should we have decided to incorporate touch-based interactions.

However, the manufacturer recommends coupling this display with the Adafruit RA8875 companion driver board to establish seamless connections with external single board computers. This device is another $39.95, so the total to use this display comes out to about $75.00. Another issue with this display is that the cheaper version that would be ideal is out of stock currently. [4]

	Universal-solder Electronics Ltd	**16x2 Blue LCD Module**	**7” Matrix Panel**	**4.3” TFT Display**

	1502 2 x 16 Blue-white	Controller HD44780		
Color variety	Blue background, white text	Blue background, white text	24 bit color display	24 bit color display
Price	$3.66 x 5	$6.00 x 5	$35.49	~ $75.00
Dimensions	80.00mm x 36.00mm x 11.00mm	80mm x 36mm x 15mm	7.56 x 3.78 x 0.59 inches	6.5 x 3.9 0.2 inches
Pixel density	N/A	N/A	N/A	800 x 480

Table 3 - LCD Screen Comparison

3.1.7 Final Decision on Display

Following a period of research into various display options, it is evident that the first two mentioned, the Universal-Solder Electronics Ltd 1502 2x16 and the 16x2 Blue LCD Module Controller HD44780, did not meet our project requirements. Primarily due to the displays being alphanumeric modules, allowing only for text and numbers to be displayed. Given our project's reliance on the use of displaying multiple colors, these displays fell short. Trying to integrate them into our project would have meant having to drop most of our original concept in trying to maybe design a game around shapes instead of colors in order to stay somewhat true to our idea. Due to this, we decided to exclude the first two options and instead take a further look at the 7” LCD TFT display.

With more consideration and extensive discussions, there was some realization that our main options were the 7 inch TFT display and an identical, but smaller 4.3 inch TFT display. Both of these components share exact identical specifications, other than one being 2.7 inches smaller. As a result the 7 inch offers a larger screen with a pixel density of 800mm x 480mm, while the 4.3 inch TFT display features a smaller pixel density of 480mm x 272mm.

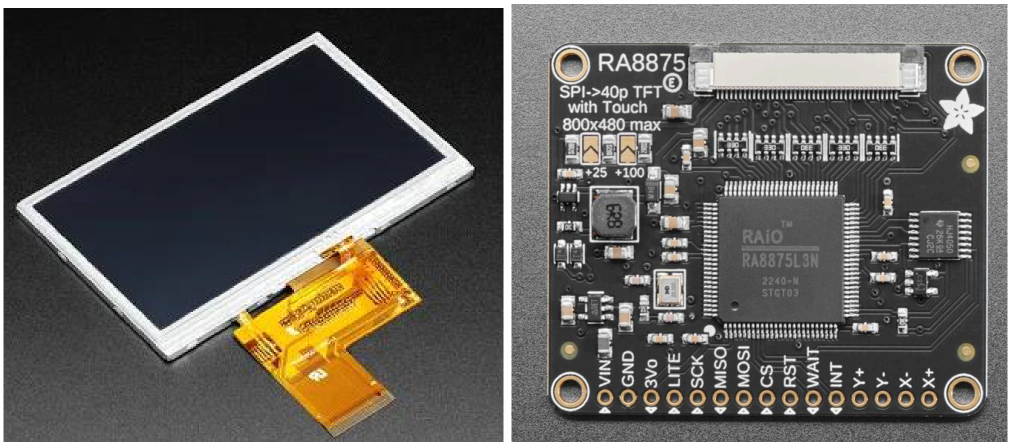

Figure 5 - 4.3” TFT Display and RA8875 Driver Board

Initially, we were striving towards finding a large screen, either by attaching multiple small screens, or buying one large screen. However, we reconsidered the benefits of a smaller display, shifting our attention from the 7 inch to the 4.3 inch. The 4.3 inch TFT display emerged as a strong contender for several reasons, one of which is that it's more budget friendly at a price of $29.95, in contrast to the $34.95 of the 7 inch. We also took into account the user experience and visual aesthetics of the device. Although the 7 inch offers a larger screen, it might have been too excessive for the simplicity of our game. Opting for the larger screen would have increased the size and weight of our final design, making it slightly more heavier, bulkier and clunkier to hold, which can make the user experience decline.

Because of these considerations, our consensus leaned heavily towards the 4.3 inch TFT display as the optimal choice for our project. This design aligns perfectly with our project goals, providing us the potential to fulfill all our screen related objectives while offering a perfect balance of cost-effectiveness, availability, and high-quality display.

3.1.8 Display Driver

Since we decided on the 4.3 inch TFT display, one thing that we were required to obtain for the screen to work as intended is a recommended driver board, since it is similar to the 7 inch and both require this board. This board, known as the RA8875 driver board, is recommended by adafruit. The board allows direct connection onto the development board as there is no way to connect our display. Our display required a 60Hz refresh and 4MHz pixel clock that is not supported by the arduino that we will be trialing. The arduino also does not even consist of enough pins to connect the screen. These conditions lead to adafruit to recommend adding this board to your project if you plan to test with an arduino board to greatly increase quality and simplicity. This board also has a basic graphic library that we can download for it which will help tremendously on all programming fronts. The way that this board works is it allows the screen's ribbon cable to be directly connected to the input port. Then on the RA8875 board there is a driver chip and also 768 KB of ram that will allow the board to buffer the display.

This product will help dramatically increase the quality of our product and also dramatically increase our quality of life when using the corresponding screen. However we also considered the price point of $34.95 and also where we would put the board inside of our controller. The price point while slightly more than expected is in fact something we considered reasonable for all of the capabilities it provides. As for the size of the device, there are no specific measurements of the device available in its datasheet or description, however it is shown that it is not much larger than a quarter which should fit nicely into some of the different models of our device. After considering the price of the board and the size that it would take up in our device we deemed this product to be essential for our project. We have used this board to allow us to produce the most high quality and professional look for our end device. [39]

3.2 Single Board Computers

A single board computer is the central nervous system of our device, orchestrating and overseeing every part of our project. Its primary responsibility lies in executing the code we programmed onto it and transmitting precise commands to each individual component. It is imperative that we meticulously select a microcontroller capable of shouldering the multitude of tasks we assign to it.

This microcontroller must possess the strength to seamlessly manage a variety of functions. First and foremost, it must support the operation of a screen, ensuring that it runs at a commendable frame rate of no less than 20 frames per second. In addition, it should seamlessly accommodate a small speaker, enabling audio feedback and enhancing user interaction. The inclusion of an analog or digital control toggle is also a prerequisite, offering intuitive and versatile control options. Moreover, our microcontroller must be equipped with the necessary memory resources to store essential data, such as high scores for our game. This ensures that players' achievements are preserved and can be accessed seamlessly.

In essence, selecting a microcontroller that can proficiently handle all these multifaceted processes is crucial to the success of our entire system. Without this core component operating seamlessly, our vision cannot be realized. The subsequent section delves deep into the intricate aspects and responsibilities this microcontroller must shoulder, underlining its paramount role in the realization of our project.

3.2.1 Single Board Computer Options

In this day and age, there are a magnitude of different single board computers with different microcontrollers to pick from, all varying in components, functions, memory, and speed depending on what you are looking for in the microcontroller and project. Researching the different single board computers and microcontrollers, we stumbled upon some that we thought would work well for our project and idea in mind. The ones we had considered to pick from are:

- Raspberry Pi Zero W - Broadcom BCM2835 SoC
- ESP32 - Espressif ESP32 Microcontroller

- MSP430FR6989 Microcontroller
- Arduino Board - ATmega328P Microcontroller

3.2.1.1 Raspberry Pi Broadcom BCM2835 SoC

The Raspberry Pi Zero W is a single board computer, with the microcontroller Broadcom BCM2835 SoC. It is a remarkably compact and budget-friendly single board computer that has been at the heart of a wide range of innovative projects. Its adaptability knows no bounds, allowing it to be transformed into diverse creations, including a security camera, portable media center, Wi-Fi hotspot, or even an internet radio station. The versatility of this microcontroller makes it a top choice for enthusiasts and tinkerers seeking to bring their imaginative concepts to life. It was among the very first single board computers that came to mind, thanks to its extensive capabilities and the creative possibilities it offers. One specific project in mind that made this one of our options is that Raspberry Pi's can be used to make a retro gaming console by using software like RetroPie. This would allow the user to play games from consoles such as the Nintendo Entertainment System (NES), the Super Nintendo Entertainment System (SNES), and Sega Genesis, all of which are more graphic and processor intensive than our project. So, we knew this microcontroller would be able to handle our design.

This single board computer model is powered by a 1 GHz single-core ARM-S processor, which we believe would be more than enough for our project. The controller has 512MB of RAM that is shared with the on-chip GPU, making it suitable for running basic applications and projects. This board features a mini-HDMI port and micro-USB, giving us more options for our components and back up selections. It also features onboard WI-FI and Bluetooth 4.0, which won't be needed for the core of our project, but this would have allowed us to expand and add more features if needed. The Broadcom BCM2835 SoC can run various operating systems and supports a wide range of languages including Python, C/C++, and more. This is great because our group is more familiar and comfortable with C/C++, so we would not have had to spend extra time learning a new language. It is also known for its low power consumption, making it suitable for our handheld, battery powered project. The Broadcom BCM2835 SoC is one of the most cost-effective Raspberry Pi controller models making this one of our top contenders for our project, especially since one of us owns a Raspberry Pi Zero W already, saving on costs.[10]

3.2.1.2 ESP32-WROOM-32 Espressif ESP32

The ESP32 is indeed an intriguing series of single board computers known for their cost-effectiveness and power efficiency. These compact boards come equipped with integrated Wi-Fi and Bluetooth features, making them quite versatile for a broad spectrum of projects. They can be used for creating simple games, such as 2D and retro style games, making them suitable for our project design. Among the array of applications that the ESP32 is particularly suited for, Internet of Things (IoT) projects stand out. Smart thermostats, door locks, and weather stations represent just a few examples of IoT applications that can be enhanced with the ESP32, offering remote monitoring capabilities. Similarly, home automation enthusiasts appreciate the ESP32 for its ability to control various appliances and security cameras with ease. The reason we considered this one as an option is because the ESP32 has been used to play games such as Pac-Man, Space Invaders, and platformers.

This microcontroller features a dual-core LX6 microprocessor, each core can be clocked to up to 240 MHz, which allows for multitasking and handling real-time tasks efficiently. The dual-cores would be great for game processing. The Espressif ESP32 microcontroller includes 520KB of SRAM, which is divided into RAM, data RAM, and other memory. It also has 448KB of ROM, which is used for system functions and initializing code. The Espressif ESP32 microcontroller does not include a dedicated hardware for graphics, but is able to use libraries and functions to draw images on the screen. The microcontroller can be programmed with a variety of tools and languages, including Arduino IDE(C/C++), MicroPython, and PlatformIO. Of these we are familiar with the Arduino IDE, since it is a C/C++ based IDE. The costs of the board range from $5-$10, making it a very affordable option.

Considering all these specifications, we decided that the ESP32-WROOM single board computer with the Espressif ESP32 microcontroller would be a good option for us, but we chose not to proceed with it considering we have other single board computers already in our possession, saving us time and money to use the options available to us already. This is a good microcontroller to keep in mind, and we could have ordered one as well as a backup. [15]

3.2.1.3 MSP EXP430FR6989

The MSPEXP430 proved to be a practical choice for our project, aligning seamlessly with the familiarity we all share from our past engineering courses. Opting for this board would benefit us by not only reducing our project costs, but also get around the delays associated with purchasing and shipment of the board. After researching this single board computer, it was revealed that it does not excel in handling graphically intensive games. Nonetheless, it can be used to create simple or lightweight games, such as Pong, Snake, Simon Says, and some simulation-based games. We believed it held the potential to fulfill our project requirements.

This single board computer features a 16-bit RISC CPU, which operates at a maximum clock frequency of 8 MHz. This is not designed for high performance computing but is practical for many low powered and embedded applications. The MSP comes with 128KB of Flash memory for program storage and 2KB of RAM for data storage, this is used for storing code and data. This board offers a wide range of integrated peripherals, none of which would be useful for our project. The single board computer uses Code Composer Studio, which is C/C++, which we are familiar with. Since we each own our own MSP430 board, we would save on costs for purchasing and shipping.

In summary, this single board computer does not meet our expectations for our needs. We had initially believed that it would suffice, and fulfill our needs, but the limitations of a low processing power and memory give rise to concerns regarding its ability to meet our requirements. It could potentially address our needs, but just barely. So the prospect of investing time and effort into utilizing this board where success is not certain, would not be a worthwhile choice when there are other options available, where success is more attainable than using the MSP430. [33]

3.2.1.4 Arduino Uno ATmega328P

Arduino's represent an exceptional and open-source single board computer platform, offering a remarkable degree of versatility for creating electronic projects. These devices are designed for a wide range of enthusiasts who want to build and prototype various systems. What truly stands out about this diverse board and chip series is the expansive array of options it gives us, allowing for a spectrum of creative possibilities to be explored. Among these choices, our preference leans towards the highly regarded Arduino Uno, which came recommended by many of our classmates and previous Senior Design class takers. However, should the need for more capabilities or expanded resources had arisen, we had other microcontrollers and options available to us.

The Arduino Uno is powered by the ATmega328P microcontroller that operates at 16 MHz, which is sufficient for our tasks. It has 32 KB of Flash memory for storing program code, 2KB SRAM for data storage and 1KB of EEPROM for non-volatile data storage, this gives the Uno a versatile memory hierarchy. The Uno lacks a dedicated GPU, but it can interface with various display modules and sensors for graphic display. The ATmega328P has various peripherals and I/O pins we can use, making this a great option for us to connect our toggles, speakers, screen, and haptic feedback. The ATmega328P uses the Arduino IDE for programming, the IDE is user friendly and supports C/C++. And luckily, one of us has an Uno in our possession, saving us costs and time. After comparing and researching boards, the Arduino Uno is a cost-effective single board computer and microcontroller that meets most of our criteria and needs.[7]

	MSP430FR6989	**Arduino Uno**	**Raspberry Pi Zero W**	**ESP32-WROOM-32**
Microcontroller	MSP430FR6989	ATMega328P	Broadcom BCM2835 SoC	Espressif ESP32
CPU	8-Bit	8-Bit	16-Bit	32-Bit
Core Count	Single	Single	Single	Dual-Core
Clock Speed	16 MHz	16 MHz	700 MHz	240 MHz
Integrated GPU	No	No	Yes	No
RAM/Memory	2 KB Ram/128 KB Flash	2 KB SRAM/32 KB Flash/1 KB EEPROM	512 MB RAM	540 KB SRAM/448 KB ROM

Environment	Code Composer Studio	Arduino IDE	Raspberry Pi OS	Arduino IDE, PlatformIO, MicroPython, and the Espressif IoT
Coding Language	C/C++	C/C++	Python, C/C++, Java, Ruby	C/C++, MicroPython, JavaScript
I/O Pins	83 I/O pins	23 I/O pins	40 I/O pins	38 I/O pins
Voltage Range	1.8V to 3.6V	5V to 12V	3.3V	3.3V
OperatingTemp	-40°C to +85°C	-40°C to +85°C	0°C to 70°C	-40°C to +85°C
Price	$0	$0	$0	$5-$10

Table 4 - Single Board Computer Comparison

3.2.2 Single Board Computer Considerations

After a thorough examination of the available boards, weighing the advantages and disadvantages of each, while considering our project demands and functionality, we have decided between the Raspberry Pi Zero W and the Arduino Uno. Both of these can meet our criteria with relative ease, but a more in-depth analysis was needed to see which MCU would align with our requirements.

3.2.2.1 Display compatibility

The display quality was largely dependent on which microcontroller we would have running the device. A weaker microcontroller, for example, may not have the ability to run a display at the frames per second that we desire. Because of this we needed to make sure that we chose a microcontroller with enough ability to run such a screen. By doing so, we eliminated any devices that fall short of the requirements with such limitations.

However, this did not mean that we must've chosen a microcontroller at the extreme end of the spectrum that can run a display upwards of 100 frames per second. Such a choice would have been excessive and most likely would place undue stress on the power supply and be more costly in the long run. We wanted to find a middle ground between the two in order to stay under a reasonable budget but also still provide the highest quality device we can.

The ATmega328P is commonly used in various projects with displays. It is capable of handling basic displays, but has certain limitations when it comes to handling graphics and display related tasks. It may struggle to achieve a smooth frame rate for complex games, but our project idea game design is not complex graphically. The ATmega328P is not suitable for high frame rates, and is more commonly used with low resolution displays. It lacks processing power and memory for fluid graphics, as well as built in video interfaces and may require additional components and drivers.

The Broadcom BCM2835 SoC offers significantly more computational power and graphics capabilities than the ATmega328P. It is better suited for running a simple handheld video game with a frame rate of 20 frames per second on an LCD screen. It offers a good balance of computational power and graphical capabilities. It can handle a variety of display resolutions, but not high-resolution displays commonly found in smartphones or tablets. It also supports HDMI and composite video outputs.

3.2.2.2 Control toggle compatibility

Another critical component that we had to ensure can be seamlessly controlled by our chosen microcontroller is the control toggle, a pivotal aspect of our project. Within the section discussing control toggle options, we explore two main types that are under consideration. The type of control toggle we ultimately selected significantly influenced our decision regarding the microcontroller best suited for our project.

This seems to be a very key function in most microcontrollers, however if a microcontroller for whatever reason does not have this capability, then we will have had to immediately exclude it out of consideration for this project. The control toggle is an essential component of our device operation and therefore if a chip is unable to handle it, we could not use it. Fortunately, as previously stated, most microcontrollers that we researched have some sort of control toggle used with it, making this a highly unlikely hurdle in our selection process with microcontrollers.

Both the Broadcom BCM2835 SoC and ATmega328P can be programmed to control and interact with various external components, including toggle switches. So we can connect a physical toggle switch to them and program it to respond to certain inputs.

3.2.3 Single Board Computer Familiarity

A key factor to consider that guided us in choosing between single board computers is our familiarity with these boards and chips. As a group our prior exposure to single board computers in the past has been very limited, so it is important to us to find a chip that we have at the very least heard of before. We do not want to go into this project with a complete lack of knowledge about every single aspect of a single board computer and microcontroller, so it is imperative we decide on a single board computer that we either have had previous experience with or a popular board that we can gather resources on if possible.

For example, we all have experience with the single board computer from the MSP430FR6989 Launchpad board. Since we all have previous experience with this chip, we have a basic understanding of what this single board computer is capable of, how to use the software that goes along with it and the fundamental principles on how to program the chip. Our ability to use these skills already will at the bare minimum save crucial time that we can use on other aspects of the project. Of course, we will still have to learn many new techniques and skills but having a base knowledge, or an easy way to obtain a base knowledge, of the single board computer will play a heavy role in our final decision.

Due to the specifications and limitations of the MSP430FR6989, we have decided to pick between the Raspberry Pi Zero W and Arduino Uno single board computers for our project. We are not too familiar with them, but feel comfortable and confident enough to use them. They both can run Arduino IDE, which is a C/C++ based language that we are most familiar with. And they both have big communities behind them that we would be able to seek guidance in case things do not go according to plan.

3.2.4 Single Board Computers Memory Capability

Something that we had to keep in mind when picking a single board computer was its storage capabilities. Part of what makes most games fun is the competitive aspect of them and in order to make our game give players a competitive drive, we wanted to have high score data stored within the device in order to provide a goal for the users when they use our device.

We took inspiration from older arcade games and pinball machines as a palace to look at what type of memory we may need. We found that Electrically Erasable Programmable Read-Only Memory (EEPROMs) is a popular way to achieve what we are looking for as well as Non-Volatile Random Access Memory (NVRAM). EEPROM is a type of non-volatile memory that can be electronically erased and reprogrammed. As previously stated, this is a popular type of memory that is found within arcade games as it is easy to retrieve data from and does not need a battery backup to maintain data during power-off situations. As for NVRAM, it is similar to regular RAM, however, it combines features of RAM and non-volatile storage allowing data to be written to it and read like normal RAM. It retains data when power is removed which is essential to our project because we do not want the high score data to be lost when the device is powered off. There is also a third type of plausible memory we could have used as well. That memory would be flash memory. Flash memory is another non-volatile storage that retains data even when powered off which is essential for this part of the project as previously stated.

Fortunately, many single board computers are designed to meet these requirements, so they come equipped with at least one of these options integrated onto the chip, making the selection of a single board computer relatively straightforward. This allowed us to freely select a chip and double check at the end that our chip has one of these options as it more than likely will.

The ATmega328P has only 2 KB of SRAM, making it a challenge to store graphics, game logic, and other game data, but we did not expect this to be an issue with our game, since its idea was not graphics intensive. It also comes with 128 KB of flash RAM that is a necessary component for our game design, since it was required to store data once the game is powered off.

The Broadcom BCM2835 SoC comes with 256 MB of RAM making this a great option for any form of game below simple 2D games, and emulators for older games. It may be limiting for more complex 3D games, and not suitable at all for newer generations of games, but playing these graphic intensive games is not necessary for our project, as our gameplay will be at the level of a 2D game or less. The downside to this microcontroller is that it does not include flash memory, which is necessary for our design. In order to overcome this obstacle, we would have had to purchase and incorporate a microSD or external storage, as these have flash RAM included.

3.2.5 Final Decision

Going from a few single board computers from the MSP430FR6989, Arduino Uno, Raspberry Pi Zero W, and the ESP32-WROOM-32, we narrowed it down to the Arduino Uno and Raspberry Pi zero W. After thoroughly researching and comparing between the two, we have decided that the Raspberry Pi's Broadcom BCM2835 SoC would most likely be a better fit for our project. After researching the Raspberry Pi's Broadcom BCM2835 SoC is the better choice, with better graphical performance, better processing speed, better components, and for the same cost. Yet, we were still adamant on using the Arduino Uno's ATmega328P as a personal project. This was our original pick for microcontroller and had our plans originally planned for it, so we wanted to see how it would test the Broadcom BCM2835. They both run Arduino IDE, making the coding nothing more than a simple transfer with a few function changes here and there. Below is a full comparison of the two.

Final Microcontroller Comparison		
	Broadcom BCM2835 Soc	ATmega328P
Price	★★★★★	★★★★★
Display Compatibility	★★★★☆	★★☆☆☆
Memory Availability	★★★★☆	★★★★★
Toggle Compatibility	★★★★★	★★★★★
Group Familiarity	★★☆☆☆	★★☆☆☆

Table 5 - Final Microcontroller Comparison

3.3 Control Toggles

Our device is operated through the integration of two distinct control toggles, each exerting a pivotal influence on the overall functionality of our device. These control toggles will assume a great significance, serving as the primary interface through which users engage and navigate within our gaming environment. The selection process for the appropriate toggle requires a meticulous evaluation of various factors, including but not limited to price, size, compatibility, and comfort. In our pursuit of creating a high-quality device, it is imperative that we meticulously weigh these considerations to pinpoint the optimal toggle solution. This selection is essential to ensure the creation of a structurally sound, highly responsive, and comfortable device, thereby enabling us to deliver a product that meets our standard of quality and performance.

3.3.1 Analog Toggles

Analog toggles, commonly found in various electronic devices, operate on the principle of continuous variable resistance. Within the toggle mechanism, there is a movable component, often a joystick or thumbstick, that users manipulate to generate input. The movement of this component is detected by potentiometers, which are variable resistors. As the user moves the toggle, the potentiometers alter their resistance accordingly, producing a proportional electrical signal. This analog signal is then translated into corresponding data by the device's control circuitry, providing a continuous and nuanced range of input values. The degree of resistance in the potentiometers determines the precise positioning of the toggle, offering a granular and responsive user experience. Analog toggles excel in tasks that require fine control or a spectrum of input levels, such as navigating through a three-dimensional space in gaming or adjusting the speed of a vehicle in a simulation. [48]

3.3.2 Digital Toggles

Digital toggles function on a fundamentally different principle compared to their analog counterparts. Instead of relying on continuous variable resistance, digital toggles operate through discrete states, typically represented as binary code (0s and 1s). These toggles have specific, predefined positions or states, and they transition between these states based on user input. Digital toggles often employ switches or buttons that are pressed or toggled, generating a clear and distinct electrical signal. The device's control circuitry interprets this signal as a binary input, indicating either an "on" or "off" state. This simplicity in signaling makes digital toggles well-suited for tasks that require straightforward, discrete actions, such as activating or deactivating a feature, executing commands in software, or toggling between different modes. Digital toggles are renowned for their reliability and precision in providing clear-cut input, making them particularly useful in applications where simplicity and accuracy are paramount. [46]

3.3.3 Round Capacitive Touch Sensor

An alternative to toggles entirely is capacitive touch sensors. In the past few years we’ve seen capacitive touch introduced to gaming peripherals mostly in VR, being used by the Vive as their main precision input and the Valve index as a precision input and for individual finger movement detection. In addition to this, Valve's arguably experimental controller replaced the traditional toggles with two round capacitive touch sensors; this is what inspired our consideration in using them for our project. The overall purpose of using capacitive touch instead of a traditional toggle is mostly user preference, in the case of VR capacitive touch offers immersion. In our case, capacitive touch would offer a smoother feeling. Capacitive touch functions based on the principles of electromagnetism. A grid of electrodes run beneath a layer of dielectric material, creating a magnetic field roughly even across the sensor's pad. When something conductive is introduced to this field, such as a finger, the electric field is disrupted. The electrodes involved with the particular part of the field detect the change, and the grid locations are used to calculate where the finger or stylus is.

3.3.4 Directional Pad

The last and least likely to have been chosen option to consider is directional pads. Directional pads are a ubiquitous element on gamepads, this is due to the precision and reliability of the commands the user can enter with them. Directional pads are simply four buttons: up, down, left, and right. The element that makes directional pads unique is that these four buttons share a cover. The shared cover allows the user a seamless transition from one button to the other, this allows for eight distinct inputs: one for each individual button and one for each combination of two neighboring buttons. The use case for such a thing is when the user has more commands than there are buttons on the controller, a sequence of fast button presses on a directional pad allows for an input list much longer than individual buttons could reasonably offer on the space of one controller. Below is a full comparison table of each type of toggle.

	Analog Toggle	**Digital Toggle**	**Capacitive Touch**	**Directional Pad**
Power Consumption	High	Low	High	Low
Input range	Continuous input values	Predefined positions or states	Continuous input values	Small set of states
Use case	Tasks requiring specific input values	General button presses	Tasks requiring low accuracy but high precision	High and defined input lists
Precision	High	Reliable and precise input but in discrete steps	High, but unclear to the user	High
Programming complexity	Higher complexity	Simpler due to on or off	High	Low

	due to continuous input	ideology		
Signal Representation	Continuous electronic signal	Shows input in binary	Continuous electronic signal	Shows input in binary

Table 6 - Input Technology Comparison

3.3.5 Deciding Between Toggle Type

The initial decision that had to be made was between the three toggle-esque options and the directional pad. We decided to dismiss the directional pad because the idea for our game does not revolve around a series of fast, precise inputs. Additionally, due to our experience in using capacitive touch in VR games, we decided not to go with it because we believed it does not offer the concrete user input that our game requires.

Upon careful consideration we, as a group, have decided that we will be proceeding with an analog type toggle. This decision came from a variety of reasons. The first being this allowed with greater flexibility in the design of our device. With this game idea being a one of a kind original, it evolved as we learned more information and if we were to decide on a digital toggle then we would box ourselves into the limitations that come with a digital toggle, not allowing us to expand if need be. Second, we decided that having a continuous source of x and y values would allow for easier programming. We initially believed that only having to figure out if the button was pressed or not would allow for us to only read that input and would be easier, however, after testing both types of toggles we decided that having a continuous loop that read x and y coordinates would allows us to always know exactly what the user was doing. We did take into consideration the higher cost and higher power consumption of the analog toggle however we decided that the increased accuracy and overall user feel would be more important than the previous concerns.

3.3.6 Specific Types of Analog Toggles

Following the collective decision to persist with an analog-type toggle, our next undertaking involves a comprehensive exploration into the nuanced realm of various analog toggles. Within this endeavor, we have identified three prominent contenders that warrant an in-depth comparison within the scope of this document. The subsequent list enumerates the specific types and brands of toggles that will be subjected to meticulous scrutiny in order to arrive at a judicious selection:

- COM-09032
- Joystick 2765 – Adafruit industries
- COM-16273

3.3.7 COM-09032

The COM-09032 is what you would think of for your traditional joystick that you may find on a modern-day gaming controller. It has a sleek design that is comfortable and well known. The x and y axis are measured individually as you would expect from an analog type toggle and it is equipped with the ability to press down on the top of the toggle to read a button press from the user. The COM–09032 works by having two potentiometers – one for each axis that are continuously measured to provide that sought after x and y output. Another key aspect of this analog toggle is that it is not on its own PCB. This design factor will play a key role in deciding which type of toggle we will choose because we are required to integrate the toggle directly onto our custom PCB and with this specific toggle being detached from any PCB it will allow easier integration onto our board. [13]

3.3.8 Joystick 2765 – Adafruit industries

This specific toggle is a slight variation of your traditional joysticks that are commonly found on modern-day gaming controllers. It provides a sleeker more compact design that does not have a rounded top compared to the previous toggle. It also has slight grooves and bumps on the top of the rubber part of the joystick that provide a slight grip to the user. Functionally this toggle is very similar to the previous device having two potentiometers, one for up/down and the other for left/right that are continuously measured to read the x and y values. Like the previous control stick, this specific toggle also does not come on its own PCB which as previously stated is ideal for our situation. However, this one does have the option to purchase a small breakout PCB alongside it that we can plug this onto. This would have allowed for easier testing since it would handle the outputs of the potentiometers, this was something to consider when making our final decision. Finally, there is no mention on the datasheet of having the ability to push down on this toggle for an extra input was a limiting factor that we considered. [34]

3.3.9 COM-16273

The COM-16273 is physically quite different from the previous two out of the box. Purchased straight from the manufacturer this joystick is simply the functional mechanism and a small metal bar sticking out of the top. This design is so that way the user (this group) is able to attach any sort of rubber head onto the top. This provides great creativity and personalization. We would have then been required to find a rubber top that we like and attach it to the top which could have allowed us to customize it as much as possible. There could have been consideration of 3D printing a top to fit the shape of a thumb to provide increased ergonomics. However, this would come with an increased total cost as the base price of this toggle is not cheaper than the previously mentioned devices. Along with an increased price it would also be an increase in workload as we would be required to design something that already exists, proving to be an excessive task. Mechanically this device again works identical to the previous two joysticks having two potentiometers that are used to calculate the x and y values. Below is a full break down of the three previously mentioned toggles. [43]

	COM-09032	Joystick 276 – Adafruit industries	COM-16273
Potentiometer resistance	2 - 10K ohms	2 - 10K ohm	2 - 10K ohms
Switch function	4-Way Directional	4-Way Directional	4-Way Directional
Compatibility	Fully compatible	Fully compatible	Fully compatible
Center select button	Yes	No	Yes
Dimensions	~ 30mm x 30mm	~17.5mm x 17.4mm	~ 17.9mm x 16mm
Price	$4.50	$2.50	$3.95

Table 7 - Final Toggle Comparison

3.3.10 Toggle Compatibility and size

Taking a deeper look and comparing the individual toggles to one another, it's noteworthy that all three contenders share a similar connection type, three pins from each of the devices potentiometer that will be connected. This similarity presented a flexibility advantage to us, as it opens up various possibilities for customization at the ends to seamlessly integrate them into the final design. This also allowed us to order a multitude of the toggles and test them as needed.

3.3.11 Price

Finally, we examine the price points of these devices. Remarkably, all three options come in at a cost of less than $5, making the price virtually negligible. This affordability factor was advantageous as it allowed us the flexibility to purchase multiple units for testing, enabling a thorough assessment of their performance against one another. Having the flexibility to do this allowed us to not only stay within a reasonable budget but also to make the most knowledgeable conclusion to which device we may have needed to use as it will allow us to test them head to head.

3.3.12 Final Decision

After considering all of our research and everything previously stated we decided to initially go with the COM-09032. With all of the devices almost being very similar in system functionality it mainly came down to two things. The first being the ability of having that extra center button available. The other thing we took into consideration was the actual shape and design of the toggle itself. The COM-09032 follows the design of many common analog sticks meaning it is very well received by the general consumer. The other two devices were smaller and assumed to be less comfortable due to their design, this led us to the choice of the COM-09032.

3.4 Sound Devices

Although initially, a sound device wasn't deemed essential for our project, we decided to incorporate one for added functionality. The chosen sound device is a compact unit with a singular purpose: to emit a timer accompanied by a beeping noise, gradually escalating in intensity with each advancing level. This feature serves as an auditory cue for users, signaling when their allotted time for a given level is drawing to a close.

Given the simplicity of this task, we didn't require an elaborate or high-quality audio device. Our primary considerations revolved around power consumption, structural durability, and sound quality. The emphasis was on finding a device that efficiently serves its purpose without unnecessary complexity. While the sound device plays a crucial role in enhancing the user experience, our objective was to minimize its footprint within the overall device.

By opting for an audio device that prioritizes power efficiency and sound clarity, we could redirect our attention to other critical aspects of the project. This strategic choice allows us to strike a balance, ensuring the sound component remains effective without unduly impacting power consumption and battery life.

3.4.1 Sound Device Options

In this modern era, there are a plethora of sound and audio devices and speakers each with different features, specifications, and capabilities that are available for selection. These devices exhibit variations in components, functions, audio quality, and speed, giving us more choices tailored to our project requirements. Diving into the world of audio devices, we have chosen a few options to pick from that would have worked well with our project in mind. The candidates that we are considering include:

- Soberton Inc. SP - 3605
- Adafruit industries LLC 4227
- Soberton Inc. SP-2804L

3.4.1.1 Soberton Inc. SP-3605

The Soberton Inc SP-3605 stands out as a versatile general speaker suitable for a wide range of applications.This compact and budget-friendly device, priced at approximately $2.90 or less, proves to be an excellent choice for our project. It can be used for timers, alarm systems, music, or even entertainment. Its cheap cost and general purpose use is why we considered this as an option for our project.

This device is powered by 1 to 1.5 watts, which is about the average power consumption of small speakers. Boasting a sound pressure level of 90 decibels and a frequency of 550 Hz, it produces a semi-high-pitched output reminiscent of emergency alarms. There are two wires attached to the device, designated for power and ground, which simplify the installation process, and they both meet at a connector towards the end. Furthermore, its circular design, similar to the size of a half dollar, adds to its convenience in the installation process. [42]

3.4.1.2 Adafruit Industries LLC 4227

The Adafruit Industries LLC 4227 shares similarities with the SP 3605 in the sense that they are both a general speaker suitable for a wide range of applications. We have seen projects using this speaker for other digital games, such as a simple fishing game or 2D platformer. We have also seen it used to simulate certain notification noises depending on the application used. Notably priced at $1.95, this speaker presents a budget-friendly alternative, costing $1 less than the SP-3605. Its affordability makes it an excellent choice for integration into projects requiring audio capabilities.

In terms of power consumption, the Adafruit Industries LLC 4227 operates efficiently with 1 watt or less, showcasing an improvement over the SP-3605. With a sound pressure level of 96 decibels and a frequency of 670 Hz, this speaker produces a higher pitched audio output compared to the SP-3605. It also has two wires attached to it, one designated for power and the other ground, and they both converge at a connector towards the end.

However, it's worth noting that the Adafruit Industries LLC 4227 features an oval shape, which may impact the device's overall design. The slightly larger form might be a consideration depending on the project's size requirements. [35]

3.4.1.3 Soberton Inc. SP-2804L

The Soberton Inc. SP-2804L is also a general speaker, just like the Adafruit 4227 and SP-3605. So it can be used for most projects requiring some form of audio output. This speaker is priced at $2.60, making it more expensive than the Adafruit 4227, but cheaper than the SP-3605.

This speaker requires 1.5 to 2 watts of power, consuming more power than the other two speakers. This is something to heavily consider for our project, since we are aiming for low power consumption. With a sound pressure level of 92 decibels and a frequency of 450 Hz, it produces an audio sound that is relatively low compared to the other speakers. This also has two wires, one for power and one for ground, that converge to a connector.

The SP-2804L features a round shape, similar to the SP-3605, but is still smaller than it by 7 mm in diameter. This is a plus, cause the smaller our components are, the smaller the handheld design can be. [41]

3.4.1.4 Comparison Table

Below is a visual breakdown of the selected speakers compared to one another in different technical aspects. We used this chart as a guide to allow us to ensure a high quality product was chosen.

	SP-3605	**Adafruit 4227**	**SP-2804L**
Power Rated	1 W	1 W	1.5 W
Power Max	1.5 W	1.5 W	2.0 W
Sound Pressure Level	90 dBA	96 dBA	92 dBA
Frequency	550 Hz	670 Hz	450 Hz
Impedance	8 Ω	8 Ω	4 Ω
Termination	Wire Leads	Wire Leads	Wire Leads
Shape	Round	Oval	Round
Dimension	36.00mm	30.00mm x 20.00mm	28.00mm
Height	5.00mm	4.80mm	4.40mm
Price	$2.90	$1.95	$2.60

Table 8 - Final speaker comparison

3.4.2 Speaker Compatibility

Taking a deeper look and comparing the individual speakers to one another, it's noteworthy that all three contenders share a similar connection type, two wire ends that converge to a connector. This similarity presented a flexibility advantage to us, as it opened up various possibilities for customization at the ends to seamlessly integrate them into the final design. Considering power consumption, a critical aspect in our deliberations, each speaker aligns closely in this regard. Specifically, both the SP-3605 and Adafruit Industries LLC 4227 boast a power consumption of 1.5 W, while the SP-2804L has a slightly higher consumption at 2.0 W. This makes us lean more towards the SP-3605 in regards to power consumption.

Distinguishing itself in audio output, each speaker in our consideration offered a unique audio signature. The SP-3605, characterized by a semi-high pitched audio, sets a certain tonal standard. In contrast, the Adafruit 4227 takes it a notch higher, delivering an even more elevated pitch than the SP-3605. On the other end of the spectrum, the SP-2804L offers a relatively lower sound compared to its counterparts.

In the context of our device, the choice for a more high-pitched sound becomes calculated. The reasoning behind this decision lies in the thoughts that a subtle or low audio tone might risk being overlooked or dismissed by the user. By opting for a higher pitch, we ensured a more pronounced and attention-grabbing auditory experience, creating an improved user engagement. This consideration highlights the importance of not just sound quality, but also the psychological impact it has on user interaction and responsiveness.

3.4.2.1 Speaker Dimension Compatibility

An integral aspect of our component selection revolves around the speaker size, a feature that bears significance not only in its individual performance but also in its impact on the overall project design. To delve into the specifics, the SP-3605 boasts dimensions of 36 x 5.0mm, while the Adafruit Industries LLC 4227 unfolds with measurements of 30.0 x 20.00 x 4.80mm. Meanwhile, the SP-2804L stands out with its compact size, measuring 28.0 x 4.4mm. Recognizing the pivotal role of size in project integration, it became crucial to compare these dimensions to ensure a seamless incorporation into the controller's framework.

Noting the size of the speakers, the Adafruit Industries LLC 4227 emerges as the largest of the three, following the SP-3605, and lastly the SP-2804L as the smallest. All options seem promising of fitting well into the designated space, emphasizing the versatility of our design choices. This meticulous examination ensured not only compatibility in terms of size but also aligns with the ergonomic considerations for the controller's form factor.

3.4.2.2 Speaker Price

Finally, we examine the price points of these devices. Remarkably, all three options come in at a cost of less than $3, making the price virtually negligible. This affordability factor was advantageous as it allowed us the flexibility to purchase multiple units for testing, enabling a thorough assessment of their performance against one another. Having the flexibility to do this allowed us to not only stay within a reasonable budget but also allowed us to make the most knowledgeable conclusion to which device we needed to use as it allowed us to test them head to head.

3.4.3 Final Decision

In light of our comprehensive research, our initial course of action involved the selection of the Soberton Inc SP-2804L speaker. This decision was based on it being the smallest device among the options, as well as having a low tonation, so the sound would not irritate the user over time. Given that nearly all other assessed categories showed comparable results, these two specific specifications emerged as the most important to our decision-making process. It's noteworthy that, despite these considerations, the cost-effectiveness of the speakers allowed us the flexibility to conduct thorough testing of all options. This approach enabled us to identify the speaker that not only aligns with our technical requirements but also complements the type of audio cues we desire for our game.

3.4.4 SD Card Reader for the Speaker

Upon examination of the speaker performance and delving deeper into the subject, we have come to a realization that we might potentially use an external SD card reader in our design, it is not certain depending on how the speaker plays the audio files we choose to incorporate into our game. The Arduino Uno is able to play audio from a speaker without the SD reader. Doing so makes it easy to use, since we can just plug the speaker directly into the board, rather than have to rely on extra components. It is cheap, since we do not have to purchase any extra components, compared to just using the board and speaker. And lastly, when executing this method, it plays the audio relatively loud, making it easier to hear and increasing the user experience. The drawbacks when using this method, is that the audio it produces is relatively low quality, meaning we would have to invest time into experimenting with sounds to find the right output we would like for our project. It can only play tones, not actual music, and it can not read from any files we want to incorporate. Thus, our reasoning for maybe having had to attach an SD reader if we want to improve on our audio, but it is not a necessity.

Venturing down the SD card reader route would have heavily improved the audio, creating a superior and pristine auditory output for the user. It gives the ability to produce and play multiple different sounds, so we would not be limited to just tones. And it would use a little storage, so we could have gotten a cheap, small card at around 16-32GB, making this an affordable option. Despite the considerations, we did not adopt the use of SD card readers for our design.

3.5 Haptic Feedback

Haptic feedback refers to the technology that simulates the sense of touch through tactile sensations in electronic devices. It enhances user experiences by providing physical feedback or vibrations in response to user interactions. Commonly found in smartphones, gaming controllers, and wearable devices, haptic feedback technology adds a new dimension to human-computer interaction. It allows users to feel a sense of confirmation, such as a subtle vibration when pressing a virtual button, providing a more immersive and intuitive connection with digital interfaces. Haptic feedback plays a crucial role in creating a richer and more engaging user experience by incorporating the sense of touch into the otherwise visual and auditory realms of technology.

Although initially, a haptic feedback device wasn't deemed necessary for our project, we decided to use one in order to enhance the users experience and increase functionality. The chosen haptic feedback device is a compact unit with a singular purpose: provide a physical vibration to the user when a specific task is complete. While the overall task of this device was relatively simple this does not mean there was a lack of research. To ensure that we chose the best device possible we will break down the different types of haptic feedback and also individual devices in order to have the highest quality product in the end. The main two types of haptic feedback that we are going to investigate are Eccentric Rotating Mass feedback and Linear Resonant Actuator feedback. There is also a third type of feedback called Piezoelectric actuators. This type of feedback was ruled out quickly, however to ensure we cover every option we still looked into it. Below is an example of the ERM and LRA haptic feedback devices.

Figure 6 - Haptic Feedback Modules

3.5.1 Eccentric Rotating Mass (ERM) feedback

Eccentric Rotating Mass (ERM) is a common type of vibration motor. The way that it works is that it has an unbalanced mass attached to a motor shaft. When that motor spins, it allows the unbalanced mass to create vibrations. This provides tactical feedback that the user can feel. These specific motors are usually found within smartphones and gaming controllers. The strength and frequency of the vibrations can be controlled to suit different applications and user preferences. Despite their simplicity, ERM-based haptic feedback remains integral to delivering a more engaging and responsive interface in modern electronic devices. [5]

3.5.2 Linear Resonant Actuators (LRA) feedback

Linear Resonant Actuators represent a more sophisticated mechanism. An LRA consists of a mass attached to a spring within a magnetic field. When an electromagnetic coil within the actuator is energized with an alternating current (AC), the coil drives the mass to vibrate at its resonant frequency. This resonant frequency is a natural frequency at which the mass-spring system oscillates most efficiently. The controlled vibrations produced by LRAs are highly precise, allowing for a more nuanced and varied tactile feedback experience compared to traditional vibration motors. LRAs are commonly employed in electronic devices where a more refined haptic response is desired. This technology finds applications in smartphones, wearables, and gaming peripherals, enhancing the sense of touch for users. The advantages of LRAs include faster response times, lower power consumption, and the ability to deliver subtle and distinct vibrations, contributing to a more immersive and customizable user experience.[19]

3.5.3 Piezoelectric Actuators

Piezoelectric actuators utilize the piezoelectric effect, a phenomenon exhibited by certain materials that generate mechanical motion when subjected to an electric field. In the context of haptic feedback, these materials are often crystals or ceramics with piezoelectric properties. When an electric voltage is applied, the piezoelectric material undergoes deformation, creating precise vibrations. In the pursuit of a more immersive user experience, piezoelectric actuators contribute by providing haptic feedback that is not only responsive but also highly customizable. The applications of piezoelectric actuators extend beyond consumer electronics, influencing various fields where precise tactile feedback is crucial for user interaction and feedback mechanisms. [38]

3.5.4 Comparison table

The below comparison table shows a general take on each of the types of haptic feedback compared to each other. The following sections will provide a more specific breakdown of the individual devices that we researched after deciding which type of haptic feedback to use.

	ERM	**LRA**	**Piezoelectric**
Precision and Control	Moderately precise	High precision	Excellent precision
Response Time	Fairly fast response time	Fast response time	Extremely fast response time
Power Consumption	Moderate power consumption	Less power consumption compared to ERM	Lowest energy consumption of the three
Size	Larger and bulkier	More compact and lightweight	More compact and lightweight

Table 9 - Comparison of haptic feedback type

3.5.5 Deciding between types of Haptic Feedback

When deciding which type of haptic feedback our group wanted to use for this device it came down to a few different criteria. We did not need anything incredibly precise since we are not asking a whole lot from the device. The haptic feedback we planned to involve was just a simple buzzer at certain points to give the user feedback on completing a task. Another criteria was size. The device must have also been able to fit into the base of the handle for our device. We ideally wanted it to take up as little room as possible to allow for other, more essential devices and cables. Finally we must look at the price of all the possible choices. The general price of the ERM and LRA devices are relatively cheap, however the Piezoelectric can get a bit more pricey due to their overall better performance. With these criteria taken into consideration the group has decided to proceed with both the ERM and LRA. Since the price of both of these are inexpensive we decided it would be best to obtain both parts and test them ourselves to see which better aligns with our vision. From our research it appears the LRA will outperform the ERM however we thought it would be beneficial to physically test both devices.

3.5.6 Haptic feedback options

With all of the present-day devices that use haptic feedback we are presented with many options to choose from when looking at ERM and LRA haptic feedback options. All of these devices present their own strengths and weaknesses having different weights, sizes, RPM and voltages. Out of all the options available we chose a few that we believed would work well for our device and researched them further. The following options are all the ones we considered:

- Tatoko vibration motor
- Speed technology – 316040001
- JIE YI Electronics – JYLRA1040ZF

3.5.6.1 Tatoko Vibration motor (ERM)

The Tatoko motor immediately grabbed our attention due to its compact size and budget-friendly price of $12.99 for a pack of 10 (less than $2 per unit). This makes it an ideal choice for our project, allowing us to purchase them in bulk and conduct numerous tests to ensure they meet our standards. The versatility of this product makes it suitable for any device requiring feedback, given its small, lightweight design and the ability to generate a powerful vibration with a rotating mass speed of 12000 RPM.

With a 3V rating, this motor aligns with the specifications of similar devices we have considered. All these features are housed in a compact 10 x 6mm shell, making it suitable for integration into any part of our device. All of these specifications were taken into consideration when deciding which device to proceed with. [31]

3.5.6.2 Speed Technology – 316040001 (ERM)

This haptic feedback device shares similarities with the Tatoko motor as it belongs to the ERM type category. With a 3V rating, a compact form factor, and an affordable price of $1.20 per unit, it presents an enticing option. However, notable distinctions exist between this device and the Tatoko motor. The former features a small, flat disc shape, in contrast to the previous long, round cylindrical shape of the Tatoko. Additionally, it is slightly smaller, measuring just over half the size of the Tatoko at approximately 5 x 5mm.

Another key difference lies in the speed of the rotating mass. While the Tatoko motor operates at 12000 RPMs, the speed technology device falls just below that at 10000

RPMs. This results in slightly less powerful vibration feedback for the user. Considering our project's requirements, where we need the vibration to be distinctly felt through the casing to the user's hand, this discrepancy in vibration intensity may play a significant role in our decision-making process. [33]

3.5.6.3 JIE YI Electronics - JYLRA1040ZF (LRA)

The JIE YI Electronics haptic feedback option stood as our sole LRA choice for consideration in this project. While it shares some similarities with the previous two items, being part of a distinct class of haptic feedback introduces unique characteristics. Notably, this device is rated for a slightly lower voltage at 2.5V, with a crucial difference—it operates on 2.5V AC, unlike the previous two that were designed for DC power. Opting for this device would necessitate incorporating an AC to DC converter within the device to provide the required AC voltage.

Furthermore, the JIE YI feedback device is the largest among the three, boasting a 10mm diameter. Assuming a correlation between size and weight, this also implies that it is the heaviest, potentially posing challenges in balancing our project. The increased size might complicate fitting other electronic components within the controller, and integrating the converter into the device presents an additional hurdle.

Despite these potential drawbacks, our research indicates that, compared to ERM devices, this LRA option offers higher accuracy and precision in vibrations. However, this heightened quality comes at an increased cost, with the device priced at $3.79. While still within a reasonable budget, the expense could have escalated if multiple units were required for testing purposes. Hence, careful consideration was needed to weigh the benefits of enhanced performance against the potential challenges and additional costs associated with this LRA option. [35]

3.5.7 Comparison Table

The below comparison table shows a general take on each of the specific products compared to each other. We made use of this table as a guide to help us make the most informed decision for which product to use.

	Tatoko vibration motor	**Speed technology**	**JIE YI Electronics**
Rated voltage	3V DC	3V DC	2.5V AC
Size	10 x 6mm	5mm diameter	10mm diameter
RPM	12000	10000	N/A
Price	$1.30	$1.20	$3.79

Table 10 - Final comparison of haptic feedback devices

3.5.8 Final Decision

After thorough consideration, we opted to move forward with the Tatoko vibration motor as our primary device for this project. Our decision is rooted in the belief that it presents the optimal combination of power, size, weight, and affordability. Although there is a slightly higher cost associated with the Tatoko motor, we consider this difference negligible, especially when compared to the Speed Technology alternative, which is slightly larger. We anticipate that the Tatoko's higher RPM will contribute to a more satisfying user experience, with vibrations palpable through the device's casing.

Acknowledging the capabilities of the Speed Technology device, we recognized its potential compatibility with our project and decided to order both options. The minimal cost of each unit made this testing approach possible. While the Tatoko motor is our initial preference, we remained open to the possibility of adopting the Speed Technology device should it prove superior during testing. Our commitment is to deliver the best-performing device for our project, and we were prepared to adapt our choice based on physical results.

3.6 Battery

For our project we did not anticipate a large battery to be necessary for our device. Almost all of our components would draw less than one to two watts of power a piece. This allowed us great flexibility in our battery decision. Since we needed to pick a battery that would have a large enough output to handle all of these components, the final battery choice was decided close to the end of our research and development phase. When looking at which battery to select we had a few different types of batteries to choose from. We could use pretty much any type of common battery so we were able to choose from Alkaline Batteries, Lithium-ion batteries, or Nickel Metal Hydride batteries.

3.6.1 Alkaline Batteries

Alkaline batteries, recognized as one of the most prevalent types of household power sources, are available in a variety of sizes, including but not limited to AA, AAA, C, D, and 9V. Formulated with zinc and manganese dioxide as their primary electrodes, these batteries boast a reputation for reliability and efficiency. It's important to note that Alkaline batteries are classified as non-rechargeable, a characteristic that may raise concerns in terms of environmental impact. Despite this drawback, their widespread use is attributed to their ability to maintain a consistent voltage output.

One of the distinguishing features of Alkaline batteries is their remarkable adaptability to diverse environmental conditions. Operating seamlessly across a temperature spectrum that spans from a freezing 0 degrees Fahrenheit to a scorching 130 degrees Fahrenheit, these batteries prove to be a dependable power source in various climates.

This unique attribute positions them as a go-to option for a broad spectrum of electronic devices, ranging from everyday household items to essential outdoor equipment.

As we delved into the power requirements of our specific device, it became evident that a robust and extensive power supply is not a requirement for optimal performance. In light of this insight, a prudent choice would involve considering the utilization of a modest number of AA batteries or perhaps one or two 9-volt batteries. This strategic approach not only aligned with the compact power needs of our device but also leverages the inherent benefits of Alkaline batteries, ensuring a balanced synergy between reliability and energy efficiency. By harnessing the strengths of these batteries, we can confidently power our device, maintaining a harmonious balance between functionality and sustainability. [9]

3.6.2 Lithium-ion batteries

Lithium-ion batteries (LIB), commonly employed in various electronic devices such as laptops, phones, and electric cars, comprise key components such as lithium, graphite, cobalt, and manganese. The operational principle involves the movement of lithium ions within the internal cell during battery usage, generating a charge to power electronic devices. This internal cell includes electrodes, anodes, cathodes, electrolyte, current collectors, and a separator. Notably, lithium-ion batteries are rechargeable, offering environmental benefits and eliminating the need for frequent battery replacements, thereby reducing long-term ownership costs. However, the downside lies in increased repair costs when the battery loses its ability to hold a substantial charge. With a typical lifespan of two to three years, users may face the periodic need for battery replacement, necessitating a user-friendly design for easy replacement. Additionally, the inclusion of a charging cable with devices can escalate production costs. In a hypothetical market scenario, these factors need consideration to balance the advantages and challenges associated with lithium-ion battery technology. [20]

3.6.3 Nickel Metal Hydride batteries

Nickel-metal hydride (NiMH) batteries are commonly used batteries composed of nickel oxyhydroxide as the positive electrode, a metal hydride as the negative electrode, and potassium hydroxide as the electrolyte. The operation of NiMH batteries involves the movement of hydrogen ions between the positive and negative electrodes during discharge and recharge cycles. This reversible electrochemical reaction generates electrical energy. Unlike lithium-ion batteries, NiMH batteries do not contain toxic materials such as cobalt, making them more environmentally friendly.

In terms of suitability for our device, NiMH batteries have both advantages and drawbacks. On the positive side, they are rechargeable, which is convenient for users who can recharge the batteries and avoid frequent replacements. NiMH batteries also exhibit a lower self-discharge rate, meaning they can retain their charge for a longer period when not in use. However, NiMH batteries do have limitations. They tend to be bulkier and heavier than lithium-ion batteries, which could affect the design and overall weight for our device. Additionally, NiMH batteries may have a lower energy density and voltage compared to lithium-ion batteries, potentially impacting the overall performance and runtime of the device. [37]

3.6.4 Final decision

Lithium-ion batteries have been chosen as the optimal solution for our device, after evaluating the advantages and disadvantages of alkaline and nickel-metal hydride (NiMH) batteries. Lithium-ion batteries are superior due to their rechargeability and environmental benefits, despite their higher repair costs and limited lifespan. These batteries also negate the need for frequent replacements and the environmental burden of disposable batteries. Furthermore, while NiMH batteries are also rechargeable and environmentally friendly, their heavier and bulkier design could compromise the aesthetics and functionality of our device. In contrast, alkaline batteries, while cost-effective, do not offer the rechargeability that our device requires for long-term usability and performance. The decision to choose lithium-ion batteries supports our commitment to sustainability and enhanced user experience. The table below visually explains our decision to go with lithium-ion batteries.

Final Battery Comparison			
	Alkaline	LIB	NiMH
Weight	★★★★☆	★★★★☆	★★☆☆☆
Environmental Impact	★☆☆☆☆	★★★★☆	★★★★★
Accessibility	★★★★★	★★★★☆	★★★★★
Price	★★★★★	★★★★☆	★★★★☆
Reliability	★★★★★	★★★★★	★★★☆☆

Table 11 - Final Battery Comparison

3.7 Case Material

Since we manufactured our own case, the different 3D printer filaments available must be considered. In this section we will discuss the pros and cons of a handful of popular filaments; considering metrics like how hard the material is to print and modify, how the texture feels to the user, how durable structures made with the material are, the color availability, and the safety concerns when running the printer.

3.7.1 ABS

Acrylonitrile butadiene styrene is the most difficult to print with of the three options. It has the highest required print temperature at ~250 °C. Due to this high temperature and the time it takes for something the size of our case to print, the bed of the printer must also be heated to prevent the structure from warping if the temperature difference between the top and bottom of the print becomes too high. For a print to properly cool, the entire model should ideally cool at the same time; this is the case with all filaments, but ABS encounters the most warping when cooling is not even. These higher printing requirements come with the benefit of being overall the most durable and flexible of the materials. For reference on how durable ABS is, it is the material used to make LEGO® bricks and construction worker hats. Due to this strength however, it is much harder to modify after printing than softer materials like PLA and PETG. Additionally, when printing the material releases fumes that are harmful to humans when inhaled.

3.7.2 PLA

Polylactic acid, a material derivative of corn starch and sugar cane, is the most popular material overall for use in 3D printer filament. PLA prints at a lower temperature than ABS at ~225 °C, this lower temperature means that warping is much less of a concern than with ABS; a heated bed is optional for PLA depending on the geometry of what is being printed. PLA is however the most brittle of the options, making it best suited for reference models and prototyping rather than functional tools. This brittleness especially affects our design due to our decision to use threaded brass inserts and screws to connect the main pieces of the case; every time we screw the pieces together the brittleness of PLA threatens to crack the connection points. The brass inserts are discussed further in section 8.1.2 - ‘3D Printing and Fastening.’ Due to its popularity PLA comes in the most color options of all the filament types. Printing with PLA releases no hazardous fumes. Due to the lower durability of PLA it is the ideal material for post processing, if a part has a slight flaw in the design it can be manually fixed most easily with PLA.

3.7.3 PETG

Polyethylene terephthalate glycol is the most unique among the three filament types we considered. At the same print temperature as PLA, ~225 °C, PETG offers the highest flexibility of the options, and stands the middle ground between PLA and ABS for durability. This combination of flexibility and durability makes it ideal for functional applications like the inner mechanisms of a device. The material is however, sensitive to moisture, which affects both the difficulty of printing with it and the environments in which it can be applied. Other than flexibility the main strength of PETG is that it can come in translucent variants, which is useful as a way to display LEDs through the case via PETG inserts. (ABS and PLA can both technically be made in translucent varieties, but that comes with a mess of considerations on how the required change in the chemical structure affects the durability of the material.)

3.7.4 Final Decision

Based on all the considerations outlined above and summarized in table 13 below, we have chosen to use PLA. We believe that for the final design PLA suits our needs the best as a case for the device. PLA offers the durability and flexibility to prevent the device from being brittle and breaking, but not so much flexibility that the device can be bent and the components dislodged from their locations. While the ability for PETG to be translucent is a big bonus as a material, we currently have no plans for our design that require translucent pieces. Additionally, the flexibility of PETG is not necessary: while it was considered for making sure the toggles have a durable attachment point to the system, we believe that the durability and flexibility of the PCB itself should prove sufficient for this purpose.

Final Print Filament Comparison			
	ABS	PLA	PETG
Ease of Use	★★☆☆☆	★★★★☆	★★★☆☆
Post Print Modifiability	★★☆☆☆	★★★★☆	★★★☆☆
Durability	★★★★★	★★★☆☆	★★★★☆
Flexibility	★★☆☆☆	★☆☆☆☆	★★★☆☆
Color Availability	★★★☆☆	★★★★★	★★★★☆
Ergonomics	★★★★☆	★★★★☆	★★★☆☆

Table 12 - Final Print Filament Comparison

3.8 Software and Programming Language Selection

Selecting the appropriate software and Integrated Development Environment (IDE) is of paramount importance in engineering projects. The choice of software and IDE significantly influences project efficiency, accuracy, and the ease of collaboration among team members. The right tools not only streamline the design, simulation, and analysis processes but also facilitate effective debugging and testing. Moreover, they ensure compatibility with hardware components, enabling seamless integration and efficient utilization of resources. Making the correct software and IDE choices optimizes project development, reduces errors, enhances productivity, and ultimately contributes to the successful and timely completion of engineering endeavors.

When deciding between which software and even programming languages to use our team really wanted to use something that we were familiar with. We decided that the difficulty of learning an entirely new programming language was unnecessary given the options that we had available. When deciding which programming language to use it is heavily dependent on what MCU we will use. These two decisions went hand in hand as each MCU uses their own specific programming language. The Arduino Uno uses a variation of C# as well as the Raspberry Pi since they both can share the same Arduino IDE.

After the selection of our MCU and programming language we had to look at the softwares to use for our MCU. Arudino does in fact have its own IDE that is recommended, however we wanted to consider all options. Some other popular options are Microchip studio, Visual Code Studio (with the PlatformIO extension) and avrdude. Microchip studio is a great IDE however it has a very steep learning curve that we did not want to take on or have the time to do. Visual code studio was a very plausible option as we are all familiar with it. Visual Code studio also known as VS code is a very popular IDE used by many software engineers. It was definitely in our final consideration. As for avrdude none of us have ever heard about it before. After further research we have found that avrdude is a software designed for the retrieval and transfer of data in the on-chip memories of Microchip's AVR microcontrollers. This versatile tool can be used to program the Flash and EEPROM memories, and when supported by the programming protocol, it can configure fuse and lock bits. Furthermore, AVRDUDE offers a direct instruction model that permits users to issue custom programming commands to AVR chips, regardless of whether AVRDUDE natively supports a specific feature of a particular chip.

Final Decision 3.8.1

Following an extensive deliberation of our available choices, our inclination led us towards embracing the recommended Arduino IDE as the primary programming environment for our project. This decision is based on the anticipation of complete support for our specific microcontroller chip, coupled with the potential wealth of resources and libraries it may provide to help our development. However, it is crucial to acknowledge that, at the time of drafting this document, we had not yet acquired the microcontroller hardware. This circumstance left us unable to be entirely sure, which programming platform would best align with our preferences and requirements. In light of this inherent uncertainty, we maintained an open stance towards Visual Code Studio (VSCode), recognizing it as a viable alternative in our contingency plan. The reason for this consideration comes from our team's collective familiarity and comfort in using VSCode. Should unforeseen challenges or complications have arisen with the Arduino IDE, we could have seamlessly transitioned to VSCode as a backup option, ensuring an uninterrupted and efficient project development process. This approach afforded us the flexibility and adaptability necessary to ensure the successful execution of our project, irrespective of the chosen programming environment. The table below shows a comparison of the two final IDE's.

Final IDE Comparison		
	Arduino	Visual Studio Code
Familiarity	★☆☆☆☆	★★★★☆
Library Support	★★★★★	★★★☆☆
MCU/Dev Board Support	★★★★★	★★★☆☆
User Experience	★★★☆☆	★★★★★

Table 13 - Final IDE Comparison

Chapter 4: Standards and Design Constraints

Each engineering endeavor inherently comes with constraints that define its boundaries. These limitations, far from being obstacles, often serve as guidelines, streamlining efforts and enhancing interoperability among autonomously developed systems. This next segment aims to uncover the external constraints and standards that influenced this project, while also delving into how these revelations shaped and supported its objectives.

4.1 Related Standards

In the following sections you will find public standards that had a direct impact on the design of this project. There are many advantages historically to following standards however the main benefit we received is that it will allowed us to avoid redundant work. They also allowed us to increase the project's overall functionality by ensuring that others can easily utilize and comprehend it. Some of the more emphasized standards that are used in this project are USB 2.0, IEC 61508:2010, and IEC 60086-1:2021.

4.1.1 USB 2.0

The Universal Serial Bus also known as USB is an integral part in almost any electrical or computer device. The USB is a worldwide port that is used on many different devices for numerous reasons. There have been different iterations and advancements made to the USB port. The USB 2.0 is the successor of the USB 1.0 and 1.1. The original USB was introduced in 1996, being developed by multiple companies including IBM, Intel Corporation, and microsoft. The invention of the USB port allowed for a standardized way to share data between devices, allowing for huge advancements in the computing world. USB 2.0 is fully backwards compatible with the USB 1.0 and 1.1 however it must follow the speeds of those ports. USB 2.0 was developed to allow for higher transfer rates, when it was release in April of 2000 it debuted a data transfer rate of 480Mbps, however with certain bus limitations it could only reach data transfer speeds of 280 Mbps at the time this was still an increase from the original USB which had data transfer speeds of 12Mbps. Alongside the deployment of much faster transfer speeds the USB 2.0 allowed for plug and play ability for storage devices and multimedia as well as also having support for power sources using USB connectors upwards of 5 V and 500 mA.

When a device is plugged into a USB 2.0 port, it establishes a bidirectional communication link with the host, enabling data exchange and, in many cases, a power supply to the connected device. USB 2.0 uses a shared bus architecture, allowing multiple devices to be connected through hubs while maintaining compatibility with various peripherals. This standardized connectivity has made USB 2.0 an essential and convenient technology for a wide range of applications, including external hard drives, keyboards, mice, and more.

One of the main uses of the USB port is to be able to communicate with another device in order to exchange data. This was the original reason that the USB was developed. This allows for all different types of data to be sent between the two devices that are connected together by the USB. This is an incredibly efficient and convenient way to transfer data between two devices. [49]

4.1.2 IEC 61508:2010

IEC 61508 stands as the cornerstone in the realm of functional safety, holding extreme importance across a wide variety of engineering industries, and it is equally crucial to the success of our project. Its origin dates back to the early 1990s, driven by the urgent necessity for a comprehensive and universally applicable standard that could effectively address the challenges associated with ensuring the functional safety of electrical systems. Since its inception, IEC 61508 has undergone several iterations and revisions, displaying its remarkable resilience and adaptability in keeping pace with the rapidly evolving landscape of the electrical world. This standard's evolution over the years reflects its unwavering commitment to staying relevant and effective in an ever-expanding field of electrical technologies. IEC 61508 remains a cornerstone in the pursuit of functional safety, not only serving as a foundational reference for numerous engineering sectors but also offering invaluable guidance and structure to our project, ensuring that safety and reliability are at the forefront of our thought process.

This standard plays a crucial role in ensuring the safety and reliability of systems such as complex electrical systems, programmable electronic components and much more. These systems are abundant in our world. They are found in a wide variety of things such as a small game controller all the way up to aerospace vehicles and many more devices. IEC 61508 aims to keep risks low with these devices by establishing guidelines for risk assessment, safety integrity levels or SILs and the whole lifecycle of safety-related systems all the way from the design to the operation and maintenance. This standard is relevant to our project because it will help guide us to create a product that is safe for public use. [21]

4.1.3 IEC 60086-1:2021

The International Electrotechnical Commission (IEC) serves as a preeminent global standards organization, crafting and updating international standards for electrical, electronic, and related technologies. Among these standards, IEC 60086-1:2021 stands out, particularly in the realm of batteries. This comprehensive standard outlines critical specifications, encompassing performance, dimensions, and labeling requirements. Given our project's reliance on battery power, strict adherence to IEC 60086-1:2021 is essential, signifying our commitment to aligning with international industry best practices and ensuring the safety and reliability of our solution.

Complying with IEC 60086-1:2021 extends beyond mere formality; it underscores our dedication to delivering a safe, reliable, and technologically robust project. Batteries are pervasive in the world of electronics, making adherence to this standard not just a technical necessity but also a means of promoting compatibility and safety. This commitment assures that our project meets international norms and standards, fostering quality, safety, and technological excellence while facilitating seamless integration within the global landscape of electrical and electronic technologies.

The 60086-1:2021 standard is intended to standardize batteries. This is with respect to dimensions, nomenclature, terminal configurations, markings, test methods, typical performance, safety and environmental aspects. Some of these standardizations heavily relate to our project. Stated later on in this paper, specifically in the safety and environmental constraints we will talk about just how important these two aspects are for our project. [23]

4.1.4 IEEE Standard 754

The Institute of Electrical and Electronics Engineers standard 754 is a standard regarding floating point numbers. This standard was established in 1985 as a way to address problems in how floating point numbers were implemented. Before this standard, many floating point number implementations were difficult to read and were not very reliable. This reduced their portability from computer to computer. With this standard being implemented it helped the standard of floating point numbers become consistent throughout engineering, thus increasing portability and efficiency across all devices.

The standard says that floating point numbers should have 3 components. It should have The Sign of Mantissa, The Biased Exponent, and The Normalized Mantissa. The sign of Mantissa is where a 0 represents a positive number and a 1 represents a negative number, the Biased Exponent is a field that can represent either and The Normalized Mantissa is a part of the number that will hold the significant digits in exponent form.

We followed this standard throughout our project in our programming section. This was necessary to help keep our program portable in case of needing to switch to another device. This also helped us be standardized, meaning that if the project were to hypothetically get commercialized and other engineers that were not this group were to work on it, then they would also understand our implementation of floating point numbers. [24]

4.1.5 IEEE standard 1857

The Institute of Electrical and Electronics Engineers standard 1857 is a standard regarding advanced audio and video. This standard was approved by IEEE in June of 2021, setting a standard for how all types of video and audio coding should be done. Before this standard there were communication errors between devices and people because there was no standard set.

This specific standard establishes a suite of tools designed for the effective compression of video, encompassing features like directional intra prediction, variable block size inter prediction, and context-adaptive binary arithmetic coding, along with their respective decoding processes. The intended applications and services span a broad spectrum, ranging from TV over the Internet and user-generated multimedia content to IP-based video conferences, surveillance, and various video/audio-enabled applications like digital television broadcasting, digital storage media, and communication.

This standard applies to our project because of us specifically coding video and display for our device. We adhered to this standard for a few reasons. One, to ensure that if our project were to ever be looked at by another group they would understand to the fullest extent what we were trying to convey. Secondly, we can use these protocols as a guide to assist us in case we need to compress our video onto some sort of portable storage device in order to transfer it to another device. [25]

4.1.6 IPC-2221A

The Institute for Interconnecting and Packaging Electronic Circuits (IPC) standard IPC-2221 plays a crucial role in guiding the design and layout of printed circuit boards (PCBs). This standard, approved by the IPC, provides comprehensive specifications for the design of PCBs, covering aspects such as conductor widths, spacing, and other critical parameters. Approved and adopted in 2003, IPC-2221 has since become a widely recognized reference in the electronics industry, ensuring consistency and reliability in PCB design.

As we begin on the development of our device, adherence to IPC-2221 is paramount for several reasons. This standard offers a systematic approach to the placement of components, including critical features like mounting holes, which are essential for securing the PCB within the device's enclosure. IPC-2221 ensures that the PCB layout adheres to industry best practices, promoting optimal electrical performance and minimizing the risk of interference or signal degradation.

In the context of our project, IPC-2221 provides guidelines that are particularly relevant to the coding and display components. While IPC-2221 primarily focuses on the physical aspects of PCB design, its influence extends to the overall reliability of electronic systems. Adhering to this standard enhances the comprehensibility of our project documentation, ensuring that other groups or collaborators can interpret and build upon our work with clarity. Additionally, as we ventured into video coding for our device, IPC-2221 indirectly supported our efforts by fostering a well-organized PCB layout, contributing to the overall success and reliability of our small video game device. Whether it involves optimizing the placement of video processing components or incorporating mounting holes securely, IPC-2221 served as an invaluable guide in achieving a standardized and robust PCB design for our project. [17]

4.1.7 i2C

I2C, or Inter-Integrated Circuit, is a widely used synchronous serial communication protocol that enables communication between various integrated circuits on the same circuit board. It is characterized by its simplicity, efficiency, and the ability to connect multiple devices on the same bus.

This protocol was relevant to our project because the RA8875 relies on i2c. We used the RA8875 to allow our display to communicate to our microcontroller. I2C is commonly used as one of the communication interfaces between the microcontroller or host processor and the RA8875 display controller. The I2C bus facilitates the configuration and control of the RA8875, allowing the microcontroller to send commands, set display parameters, and interact with the graphics controller. For instance, the I2C interface might be utilized to adjust screen brightness, set pixel colors, or manage touch screen functionality. The use of I2C in this context streamlines the communication between the microcontroller and the RA8875, contributing to the efficient operation of the graphics display in various electronic devices and in this case with our TFT display. [45]

4.1.8 SPI

Serial Peripheral Interface (SPI) is a widely used synchronous serial communication protocol employed in embedded systems and electronic devices to facilitate data exchange between a master device, typically a microcontroller, and one or more peripheral devices. SPI operates on a master-slave architecture, where the master device initiates and controls the communication, and the peripheral devices respond accordingly.

When using SPI with the RA8875, the microcontroller sends commands, data, or control signals by transmitting data on the MOSI line. The RA8875 responds on the MISO line, providing status or data back to the microcontroller. The SCK line carries the clock signal, synchronizing data transfer between the master and the RA8875. The SS/CS line is used to select the RA8875 among multiple SPI devices on the bus.

This SPI communication enables the microcontroller to configure the RA8875, send graphical data, and control the display parameters efficiently. It's a fast and versatile protocol, well-suited for applications where high-speed communication is necessary, such as graphics displays in embedded systems. [46]

4.2 Design Constraints

In the following sections you will find different design constraints that had a direct impact on the design of this project. There are many advantages to having followed these constraints as they allowed us to stay on time with our project as well as in good standing economically and socially.

4.2.1 Time Constraints

Time constraints for this project were set upon us by the professors and instructors above us. They have laid out different check-ins and milestones that we had to complete in order to achieve a passing grade for this class. Since a passing grade is required for all members to graduate it was imperative that we all contributed equally in order to complete these milestones. These milestones are designed to keep us on track so that we could have a completed design by the end of the fall 2023 semester and a fully completed project by the end of the spring 2024 semester. If we were unable to meet these deadlines we would collectively not pass the class and therefore not graduate, so it was imperative that we budget our time and resources extremely efficiently.

A significant time constraint that required our careful consideration is related to the design and production of the printed circuit board (PCB). It was imperative to keep in mind that not only would crafting the PCB design be a time-consuming process as we had to redesign the PCB a few times until we are happy with a final design, but there was also a considerable duration involved in its actual printing and subsequent delivery to us. We also had to consider the chance that we obtain a PCB that we are not happy with and had to redesign and ship a new one which would slow down development but we planned accordingly and incorporated enough time for this to happen. Because of these constraints, we had to factor in these time requirements when planning our project's timeline and ensured that we allocate ample time for both the design phase and the production logistics of the PCB.

Going along with the PCB, a significant time constraint we also considered was the wait for parts. We of course ordered almost all of our parts online and so we had to wait for them to arrive. This in itself is a time constraint as we must wait for the parts and then we also must test and put together the parts. We also needed to account for the possibility that the parts may not work as we need them to so we will have to order more and wait again. We had to make sure that we order our parts with plenty of time to test them and make sure that they work. We had to also consider the possibility of parts being out of stock, this is not quite as much of an issue as we will be able to research other similar parts but we still considered this.

While there are many time constraints related to research and design for this project we also considered time constraints related to the actual operation of the device itself. For starters, when the user is going from one level to another we considered a time between the levels so the user has enough time to prepare themselves while also making sure the time between them is not too long that they may lose interest. Also going from each level to the next we considered how the timer will speed up as the game goes on. We might have done a linear progression for slightly easier gameplay or an exponential progression for harder gameplay. Finally our device has haptic feedback within the controller. A time constraint for that is the need to find the perfect timing for the controller to rumble after an input is pressed by the user. We again do not want the delay to be excessive however we do not want it to be abrupt, our group had to find a middle ground for this as well. These can also be considered design time constraints as well because they took time to test and find the perfect setting for.

4.2.2 Economic Constraints

The main economic constraints that we faced were purchasing the components for the device. However, there were other purchases we considered and since our project was only funded by the members of the group we had significantly more constraints than if there were outside funding for this project. It was important for us to select components that are not unreasonably expensive for us as we did not have a sponsor while also remembering that this was a very important project and in order to be successful we needed to be willing to provide enough money to have a design that ourselves and our review committee is satisfied with.

We anticipated not only procuring all the final components for our project but also acquiring intermediate parts that may not ultimately be incorporated into the final device. For instance, if we purchased a speaker but find its sound quality unsatisfactory, or if the screen we ordered doesn't match our size specifications, additional parts would need to be ordered. Some components were dedicated solely to testing purposes. Our printed circuit board integrates the ATmega328P microprocessor from the Arduino Uno, and for testing, we used an Arduino board. This approach was pragmatic given the impracticality of waiting for the PCB to be designed, shipped, and received before conducting comprehensive component tests.

As a group, we collectively covered the expenses related to printing and delivering the PCB, having to budget for multiple iterations when redesigns were necessary. Additionally, the creation of the game's casing represented another financial consideration. Fortunately, one of our team members already possesses a 3D printer, a pre-existing asset that significantly reduces project costs. This not only eliminated the need to invest in a 3D printer but also obviated the need for outsourcing the construction of the game's casing. Moreover, the filament required for the casing is relatively inexpensive, contributing to overall cost efficiency in this aspect of the project.

The final iteration of our device is intended for mass production for consumers to use. With this in mind we made decisions as if we intend to profit from selling our device. Of course in an outside setting where the game is being mass produced, there would be discounts for purchasing components in bulk and also there would not be expenses for research and design. Having these discounts would be necessary because in a retail setting our game would have to be as cheap as a lot of these other devices are so we could not sell it over the price of our current expenses (~$210).

4.2.3 Social Constraints

Our most realistic social constraint we are faced with is how the user interacts with the game, more specifically how we are able to get the user to play the game correctly. In order to get the user to understand the game correctly so that they get the most out of it we need to have a very good explanation of how the game will work. For the purpose of this project this explanation is done in this paper, however in a theoretical retail setting to ensure that the consumer has a sound understanding of the game, we would have to provide a user's manual. In this manual we could have detailed written instructions and potentially a link the consumer could follow that would take them to a short video demonstration. In this retail scenario our group thinks it would be necessary to have both so that the consumer fully understands the game since this is a critical part of our device.

4.2.4 Equipment Constraints

For our project we did not have too many equipment constraints. Our biggest equipment constraint was related to the software and programming language that we used for our microprocessor. Since the microprocessor has a required programming language that it needs to be programmed in to understand we needed to use this language. For the microprocessor that we decided on, the ATMega328p, we must use a variant of C++. Specifically the arduino uno which we will be prototyping on uses this language. The difference between the ATMega328p C++ and regular C++ is that it has special methods and functions added for ease of use. While the group did not have a tremendous amount of experience with C++ we all have plenty of experience with programming languages in general which greatly helped our ability to translate what we know into the C++ programming language.

Another equipment constraint we had is our 3D printer. The 3D printer that we have is a modified Taz 7. Since it is not a commercial 3D printer it has its own set of limitations. Some of which being the size of the bed, its consistency and also our own ability to print the specific shape we are going for. The size of the bed greatly influenced how we print the shell of our project. It determined if we had to print the device in individual pieces and then put them together via glue or snapping them together. If the bed were not big enough to print the entire device in one session then we would have to do one of these options. Fortunately this didn’t change the entire design of the device as we made it work by just printing it piece by piece and putting them together, this only affected how we go about printing it. We also considered consistency with 3D printers. Since 3D printers are a newer technology they do have some errors while printing larger scale projects. Because of this we had to plan ahead and make sure we give ourselves ample time to test different designs, filaments and tactics when printing.

4.2.5 Safety Constraints

The safety constraints of our project were to ensure the group stays safe in the process of building the device as well as being positive that any consumers of the device will be just as safe. The construction of our project had low risk of energy. Since it is a small handheld device we did not plan on having to use any power tools, high voltages, or any dangerous devices. There was the need to solder some components. The main concern when using a soldering iron is lead poisoning. If handled incorrectly lead can get transferred to the hands of the operator and if they do not wash their hands then it can potentially be ingested causing health problems. One of our members has the experience and equipment to do soldering safely.

When it comes to ensuring the safety of our consumers, our responsibility lies in carefully protecting them from any potential hazards. One crucial aspect of this involves meticulously inspecting and addressing the electrical components of our products. We must guarantee that there are absolutely no exposed wires, and all electrical components that may be susceptible to unsafe contact are securely enclosed within our protective casing. This measure not only preserves the integrity of our devices but also mitigates the risk of accidental electrical shocks or other related mishaps.

Furthermore, it was imperative that all internal wiring was meticulously organized and precisely positioned within our device. This meticulous arrangement is essential to prevent any short circuits, which could result in severe internal issues, potentially leading to fire hazards. By upholding this standard of precision, we prioritized the safety of any potential consumers.

Additionally, we cannot understate the concern of battery safety. In order to ensure consumer well-being we made sure that our batteries are not only correctly installed but also firmly secured in place. Mishandling or improper installation of batteries can have catastrophic consequences, such as explosions, fires, and other hazardous incidents. Hence, we leave no room for compromise when it comes to battery safety, understanding the potential dangers that lie in neglecting this aspect.

4.2.6 Environmental Constraints

Environmental considerations for this project primarily revolved around the battery selection. It was imperative that we remained conscientious about electronic waste. This concern extended beyond the battery and encompassed our entire research and development phase. It was essential to avoid unnecessary purchases of excess parts, both due to budget constraints and environmental responsibility. Wastefully acquiring components not only poised to strain our financial resources but also contributes to unnecessary energy consumption in their production, which, if unused, results in eventual disposal.

Selecting the rechargeable battery option aligns with our commitment to reducing electronic waste, as it promotes reuse. To manage excess waste for other components, our approach involved striving to return unused parts or, at the very least, ensuring they are recycled. Implementing such responsible disposal practices not only minimized our environmental footprint but also potentially enhances our funding prospects, making it a win-win strategy in our project's sustainability and financial stability.

4.2.7 Ethical Constraints

Our project did not have many ethical constraints since we will not be asking for any user information. A common concern with all electronic devices is how they are monitoring you. We have seen this time and time again as CEOs are on trial for the misuse of consumer information. We did not need to worry about this as our device is entirely off the grid, there will be no network or internet connection and there is also no internal microphone for our device.

The only ethical constraint that may have been relevant to our device is the previously stated safety constraints. As long as we followed the aforementioned safety constraints we were in good shape to avoid any ethical constraints for the duration of our project even if it were to be commercialized.

4.2.8 Political Constraints

Our project does not have any political constraints. As mentioned before our product is not connected to any sort of internet or other network meaning that there is no possibility for it to collect any user data without consent. If we were to hypothetically commercialize this project, then we would need to do more research to determine if there were more policies to abide by. However, given the project's current scope and design, it falls within the established legal and political framework, thus eliminating potential political constraints at this stage.

4.2.9 Legal Constraints

Similar to our section on political constraints we did not have any legal constraints either. The only possibility would have been any licensing issues if we were to hypothetically commercialize our product. There could potentially be copyrights or licenses on similar products that forbid the design of ours however since we did not fully intend to sell this commercially we have not researched this. However, if that perspective were to change we would most definitely look into such issues but it is unlikely that there are any of these copyrights on our device since it is one of a kind.

4.2.10 Manufacturability Constraints

The manufacturing constraints of our device mainly come from total cost, accessibility of parts and also our own ability to assemble the device. Our project would likely cost no more than \$200 - \$250 which is a great price point to be at since we are all unable to provide too much into this project being college students. This means in theory if we were to sell the device we would initially have to charge over this amount which we think is excessive for the type of device it is. However, previously stated in the economic constraints section there are generally deals for mass producing these types of products. If we were to sell this device commercially we would ideally find a way to bring the price of production to less then \$40 with the idea of marketing the game at \$50 a unit. If we were to put our device on the market at the previously stated price we would not have a very competitive price so it would be mandatory to bring the cost of production down.

4.2.11 Sustainability Constraints

We aimed to create a long-lasting and durable device, we are committed to ensuring that it remains sustainable for as long as its parts permit. Our primary goal was to design a device capable of withstanding moderate wear and tear, even during extensive gameplay sessions. To achieve this, we place a strong emphasis on the build quality, striving to attain the highest possible standards given the materials we have.

A critical factor in determining the device's robustness is the type of filament we used for the 3D printer. This filament choice directly influences the device's ability to handle the previous mentioned wear and tear. While we may have limited control over the inherent quality of individual components, we have the ability to select components of higher quality, thus contributing to the overall durability and performance of the device. It's important to note that our device is not designed for extreme weather conditions. To make this information abundantly clear to hypothetical consumers, we would consider including a warning label or a small handbook in the packaging if we were to bring the device to the commercial market. This precautionary measure is intended to enhance the device's longevity and ensure that users are well-informed about its intended usage parameters.

4.2.12 Constraints Summary

To summarize, there were a number of constraints that provided some guidance for how we researched and designed our project. Our main constraints were without a doubt time and money constraints. We had to be adamant about following these guidelines in order to succeed in this project. The next most important constraints we abided by are the safety constraints as we believe this to be almost just as important. The reason this is not higher up on our priority is because only a select few people will be handling the device in its entire lifespan so its overall safety does not need to be up to the same standards as a commercial product. Also, as stated before the production of the device was not inherently dangerous as we were only using low voltage devices. Our next most important constraints would most likely be social and environmental. As for other constraints such as legal or political we did not anticipate having many if any constraints for these.

To restate again the time constraint was of the utmost importance to the group as this entire project will be the main factor for if any of us graduate in spring 2024. With this in mind we had to remember the due date for this 45 page paper, the dude date for our 90 page paper at the end of this semester and then of course our demonstration at the end of the semester in spring. With all of this in mind we used our time wisely and gave ourselves as much time as possible to order and test components as this was the largest constraint.

For our safety constraints we knew that we must minimize risks by avoiding carelessly handling power tools and high voltages in the construction of our small handheld device, with potential soldering precautions in place. When soldering was necessary, we took measures to prevent lead poisoning. For consumer safety, we meticulously protected against electrical hazards by enclosing components and ensuring organized wiring to prevent short circuits. Battery safety is also a top priority, with a focus on proper installation to prevent dangerous incidents like explosions or fires. Our commitment to safety extends to both our team and potential consumers.

Environmental constraints for this project mainly center on battery selection, with the choice between rechargeable and conventional batteries. The focus is on reducing electronic waste throughout the research and development phase by avoiding unnecessary component purchases, aligning with budget constraints and environmental responsibility. The aim is to reduce waste, promote reuse with rechargeable batteries, and implement responsible disposal practices to enhance sustainability and financial stability. Our main social constraint was ensuring users understand and engage with the game correctly. In a retail context, we would provide a detailed user's manual, including written instructions and a video demonstration, to ensure consumers' comprehensive understanding of the game, which is crucial for our device's success.

Over the duration of this constraints section we have laid out many guidelines for ourselves. We used these constraints in order to allow us to make smart and informed decisions regarding the research and design of this project. It has proven to be important to have researched and displayed all of the constraints for a project this large. It provided a resource that we came back to over the two semesters.

4.3 Project Constraints

Building and designing this device came with certain constraints and limitations, some of those are as follows. One of the constraints was resource limitation. Due to ongoing events, such as the chip shortage, we might’ve not been able to find certain needed parts. Therefore, we had to use what was available to us, in the market, at the moment. Another constraint was our supplies and materials possibly not arriving on time, due to shipping times, Florida storms, or other travel related issues. In order to offset this constraint, we ordered the parts needed as soon as possible, because the earlier the better. Some other issues might’ve occurred, such as receiving defective parts, or parts malfunctioning during the project. Our solution to this was to order double parts on components we think might have some issues. Due to the size of the design, there might have been some size and weight constraints that could have affected our project. We might’ve had to increase the size, decrease the size, or adjust certain areas in order to fit the parts. There were battery constraints as well, relating to size, we had to adjust or make space for the battery source, and ensure our device has a lasting battery life. We expand upon this in much further detail in the design constraints section of this paper.

In Figures 1 and 2, the block diagrams illustrate how the project's hardware and software components are organized. The hardware components necessary for the project were acquired gradually over the semester.

<u>Chapter 5: Comparison of ChatGPT with Similar Platforms</u>

5.1 Introduction

Artificial Intelligence (AI) is an extremely powerful technology that can be used as an aid to one's learning and education path or it can greatly hinder their experience in the same field. AI has been around for longer than the general public might think. With the explosion in November 2022 with the release of ChatGPT many people may be led to believe that AI is extremely new. This is simply not true. AI research first began in the 1950's with Alan Turing publishing "Computer Machinery and Intelligence" ; this was further researched by computer scientist Arthur Samauel who developed a program to play checkers. This program is the first to ever learn a game independently. This is to show that AI has been around all of us for many years, from simple games to advanced projects around the technological world.

With any technology of this size and power there are of course ways that it can be extremely helpful to the user and extremely detrimental to them and others, ChatGPT is no different. ChatGPT being released towards the end of this group's college career is quite an interesting spot. We of course do not want to go against plagiarism and cheating standards by using it for every piece of work we do, however we are aware that it will be readily available to us once we enter the workforce. Knowing that as future engineers we will be expected to further our education as much as possible and stay current with new technologies it would be malpractice to simply ignore such a powerful tool, however it must be used with caution. If we were to use ChatGPT and other similar platforms for every single thing we do, we would lose out on a great deal of learning by having it do all of the work for us. Plus, without further researching what it tells you, you would be blindly following it and have no way to know if the answer you got is right or wrong. However, it has proven to be a great aid and a great resource to guide us in the right direction. We have used ChatGPT in this way a few times over the course of this project. For example, we have asked questions regarding the type of memory we may need to use to store our high score. ChatGPT responded with an answer but we did not blindly follow this response. We took the time to research each of the options it gave us to see if it would be applicable to our project, and this time they in fact were.

In our society there are always companies who take ideas from other companies and put their own twist on them, this is how things grow. With this type of AI there has been no shortage of companies trying to replicate ChatGPT and its outstanding success. For example, there is Bard which is a chat based AI from Google, there is also Microsoft bing chat which of course is from microsoft, as well as Jasper chat. All of these are other chat AIs that have become popularized due to the success of ChatGPT. Each of them have been trying to pave their own way by providing unique services to their platform that will differentiate themselves from the others to generate more profit for themselves. In the following section we will be discussing the differences, and pros and cons to each of these platforms to break down what may be best for future needs. [51]

5.1.1 ChatGPT

ChatGPT, similar to the other AI services explained, functions as an artificial intelligence chatbot utilizing human language processing to generate highly human-like responses and dialogue. ChatGPT stands out for its proficiency in providing prompt and natural responses, yet it lacks certain features offered by other platforms. While regarded as an all-knowing AI chatbot, ChatGPT has limitations, notably in its current knowledge base. As of the time of writing this report, the free version of ChatGPT does not have internet access; only starting March 14th, 2023, did the paid version (ChatGPT 4.0) gain this capability. This poses a significant constraint for the average user seeking the latest information without being hindered by ChatGPT's potential limitations.

Despite this limitation, ChatGPT possesses an extensive amount of data and demonstrates remarkable intuitiveness. For instance, in our project, we sought information on relevant IEEE standards. Describing our device, we asked ChatGPT about standards applicable to its creation. While some responses were outdated or not entirely relevant, ChatGPT guided us in the right direction, enabling us to explore specific standards and determine their applicability. This exemplifies how ChatGPT might not always have the most current information but can serve as a valuable guidance tool, assisting users in refining their queries and directing them toward relevant resources. [50]

5.1.2 Google Bard

Google Bard which was released on March 21st, 2023 is a direct competitor to ChatGPT. Being released roughly 4 months after ChatGPT was released it did not have as much time to be quite as refined as ChatGPT was. Google bard has a history of not being as accurate with the information it provides. This could be due to the fact that Google Bard was trained on different data sources compared to ChatGPT. Google bard was trained on “infiniset” which is a data set that includes, Common Crawl, Wikipedia, and Conversations and Dialogues from the web. This is compared to ChatGPT who was trained on a massive data set of text, which includes all the previously stated sources and also content that was scraped from the internet before going live.

However, in a valiant effort to overcome these defects, Google Bard can be extremely quick. It has shown to be quicker than the free version of ChatGPT which is a plus, however it is not as quick as the paid 4.0 version of ChatGPT. This does bring up the next reason that Bard may be better for some users, it is entirely free. There is no paid premium version of Bard and it allows all users to have internet access unlike ChatGPT where only the users paying for 4.0 are permitted access to internet browsing. Another plus to Bard is that it was recently upgraded to PaLM 2 (Pathways Language Model) which is said to support more than 100 languages over time compared to ChatGPT’s 80 languages and also upgrade its efficiency and accuracy. [8]

5.1.3 Microsoft Bing Chat

The next competitor we have to ChatGPT is Microsoft Bing Chat. Microsoft Bing Chat was released in February 2023, 3 months after the initial launch of ChatGPT. This again was in direct response to the sudden surge of AI Chat platforms being released.

Microsoft Bing Chat is similar to ChatGPT but it also takes a different approach. Microsoft Bing Chat was released as a direct aid to their search engine, Microsoft Bing. This idea is that the chat will be able to provide links to the sources that it pulls from. This also has demonstrated that it will not be quite as precise and specific with the answers it provides however it will allow the user to do their own research by providing a link and brief description.

The idea of Microsoft Bing Chat is that it will cut down on your research time by not showing hundreds of links when you look up an idea on say google. It will provide a concise answer that it gathers from different sources and provides you with those sources as suggested places to research. A plus to this strategy is it will provide different allies for the user to explore by being able to go directly to the source of the information.

Also, a feature that Microsoft Bing Chat has over ChatGPT is it allows for more tailored outputs by providing a conversation style selection. The user can choose one conversation style from the following three options, Creative, Balanced, or Precise. This option allows for much more specific answers. However, for ChatGPT the user could simply ask the AI to “Give a more creative response” or “Give a more precise response” and you are left with a similar outcome.

Another plus to Microsoft Bing Chat is that it is almost like a chrome extension meaning that you do not have to go to their webpage to access the chat. You can simply access it from the sidebar of whatever page you are working on within Microsoft Edge. Maybe the most interesting aspect of these AI chat’s is that they both use OpenAI’s GPT-4 language model, however Microsoft Bing Chat allows all users to use this for free. As previously stated, ChatGPT has a paid premium that allows access to this, however the non premium version of ChatGPT uses GPT-3.5-turbo. [48]

5.1.4 Jasper Chat

Our final competitor to ChatGPT is Jasper Chat. Jasper Chat is a lesser-known AI chat tool, and it takes a slightly different approach than the other previously mentioned AI chats. It is mainly intended for marketing teams where it will have options to “create a social media caption” or “write a document,” making it much more specific than ChatGPT in its prompts. This can be both a pro and a con, as some users may appreciate specifically tailored options, while others may prefer ChatGPT’s open-input system that allows for both broad and specific prompts.

Alongside these specific prompts, Jasper Chat provides other features beyond the AI chat, catering heavily to marketing needs with options like content editor and templates for various content types. On the downside of Jasper Chat, there is no free option available at all. The free trial lasts 7 days, and after this period, plans start at 49 dollars, whereas, as previously discussed, ChatGPT's most popular tier is free for all.

Some more technical drawbacks for Jasper Chat include the absence of data analysis capabilities, and its content generation technology lags slightly behind that of ChatGPT's. Since these two AI tools are intended for different users, the choice depends on your specific needs. For a general chat-based AI, ChatGPT stands out as the better option between the two. [11]

5.1.5 Using ChatGPT

Throughout our group's journey, ChatGPT has proven to be a versatile tool with various applications, especially when regarding this paper. While it has been primarily utilized in guiding our research efforts, we've used it for diverse purposes. As previously stated, we've utilized ChatGPT to delve into specific memory-related topics and to aid us in the research of IEEE standards. These broad applications have served as overarching frameworks, providing us with a guide on what to research and where to begin research for certain topics.

Beyond these broader applications, there have been instances where we've harnessed ChatGPT for more nuanced tasks. One notable example is when grappling with the challenge of conveying a specific idea while lacking the precise vocabulary to articulate it effectively. In such instances, we've turned to ChatGPT to explore synonyms or alternative phrasings, utilizing the tool as a linguistic resource to enhance the clarity and precision of our written content. While a conventional Google search might offer similar results, consulting ChatGPT brings the added advantage of dynamic conversation. This allows us not only to request synonyms but also to delve deeper by seeking contextual usage in sentences, thereby deepening our understanding of the chosen words before incorporating them into our formal documentation.

In reflecting on our experiences, it's crucial to highlight our group's cautious approach in respecting the boundaries of ChatGPT's role. While invaluable as a guiding tool, we recognize the potential pitfalls of overreliance. Overusing ChatGPT could hinder our ability to conduct independent research if we turn to it for every question without subsequently doing our own investigation. Additionally, during the production stage of our project, seeking guidance from ChatGPT on assembly-related questions would be counterproductive due to its lack of contextual awareness about our project specifics. It's imperative to strike a balance, leveraging ChatGPT's capabilities judiciously to augment our learning and research processes without substituting it for our critical thinking and project-specific expertise.

Overall, our group has navigated the use of ChatGPT with a strategic mindset, leveraging its capabilities where appropriate while remaining vigilant about its limitations. This approach ensures that the tool enhances rather than impedes our learning and research endeavors.

5.1.6 Conclusion

ChatGPT stands out as a preferred AI chatbot for its remarkable ability to generate quick, human-like responses, showcasing an intuitive grasp of language. Despite limitations in current knowledge, exacerbated by the free version's lack of internet access until the recent introduction in ChatGPT 4.0, it remains a valuable tool for guidance. In contrast, competitors like Google Bard, released four months later, boast speed and free accessibility, yet suffer from accuracy issues due to different training data. Microsoft Bing Chat, introduced three months after ChatGPT, focuses on aiding searches with concise answers and source links, allowing users to explore further. While offering tailored outputs, it competes with ChatGPT's versatility and user-friendly open-input system. Jasper Chat, designed for marketing, provides specific prompts but lacks a free option, content generation technology comparable to ChatGPT, and data analysis capabilities. Ultimately, ChatGPT's balance of versatility, user-friendliness, and a free tier makes it the preferred choice for general chat-based AI. The table below emphasis ChatGPTs balance of all of these factors.

	ChatGPT	**Google Bard**	**Microsoft Bing chat**	**Jasper Chat**
Price	★★★☆☆	★★★★★	★★★★★	★☆☆☆☆
Accuracy	★★★★☆	★★★☆☆	★★★☆☆	★★★★☆
Speed	★★★★☆	★★★★☆	★★★★☆	★★★★☆
User interface	★★★★★	★★★★☆	★★★☆☆	★★★☆☆
Language Model	OpenAI's GPT-3.5-turbo	OpenAI's GPT-4	OpenAI's GPT-4	OpenAI's GPT-4

Table 14 - Final AI comparison

Chapter 6: Hardware Design

6.1 General Design

During the process of creating the design for our project we have had multiple iterations. There were many things taken into consideration when thinking of the project design. Some of which being weight, size, comfort, and part availability. When originally thinking of the design we wanted to have the screen curve as stated above. We quickly realized that this would not be feasible for us. In order to get a curved screen we would need to have it specially made for us and this proved to be too expensive. We then started to think of the alternate methods stated above.

6.1.1 Shape and size

We then shifted our focus to how large this device would be. We had to take into account fitting all of the electrical components inside such as the PCB, MCU, and wires. We also need room to have the speaker and analog stick while leaving space for a haptic feedback device in the base of the controller. We were not entirely sure at the moment what size the device will be in order to incorporate all of the previously mentioned specifications, however we had seen many controllers that contain all of these components plus more that are about 90mm x 120mm. We hoped to have our final design in this range with the expectation that our controller may be slightly larger because of our lack of professional equipment.

Our goal was to 3D print the casing for the controller in order to cut costs as much as possible while also providing a lightweight shell. This would allow us to test out different designs so we can test different comfort levels as well as allowing us to mold the controller to the exact dimensions we need to provide enough room for all the components previously stated. This would also allow us to test completely different shapes for the controllers.

Our team had discussed doing a single handle for the device with the display being on top of it. We have also discussed a single controller with the display to the side of the device and the last idea is having Two controllers for the device, one on each side of the device allowing for both hands on the device to maximize control. Each iteration would have its pros and cons of course. For the single handle with the screen on top, this would allow for a sleek one handed design with the display directly in front of the user. However, we would have to consider counter balancing the device so that way it is not top heavy. For the design where the screen is to the side of the screen this would be a more traditional look when compared to other video game devices today, however we would then need to worry about the weight of the screen causing a rotational force to the direction of the screen which may be difficult for the user to control depending on the weight of the screen. Finally the design where the device has two handles on each side of the display would allow for the best balance of the device, however currently, only one side of the device would have any controls so this may cause some confusion with a user.

6.1.2 Weight

As for the weight of the device this depended heavily on the screen choice that we ended up going with. We estimated that most of the weight will come from the weight of the screen since the other components are fairly lightweight. We had to make sure that the device was not top heavy so there were considerations to add some counterweights to the controller in order for the device to feel correct in the user's hand.

6.2 Case

The case design went through multiple iterations as we advanced the idea of our project. These iterations are showcased in Figure 7: ‘A’ is the single handled design, ‘B’ is the two handled design, and ‘C’ is the portable gaming console design.

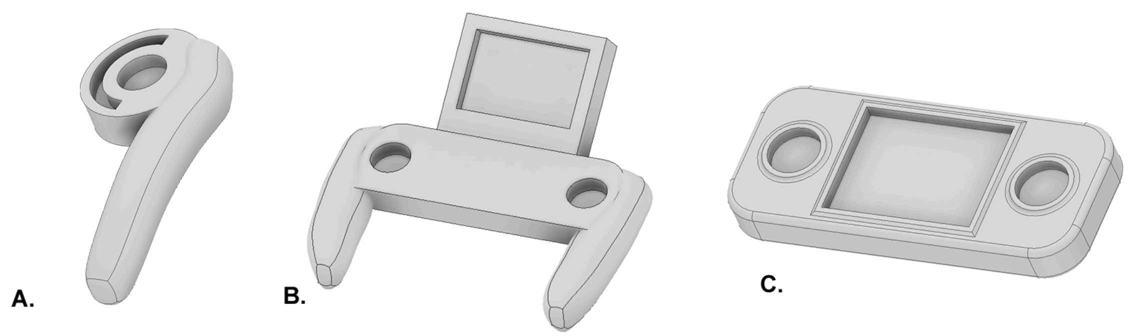

Figure 7 - Case Design Iteration Mockups

6.2.1 Single Handle

The single handle design is the first idea we had for the design of the case. Inspired by the feel of the Valve index controller, the device would fit in one hand. The user would interact with one toggle in the middle, and a screen with the colors would wrap around the area designated for the toggle. The perks of this design is that it is light, ergonomic, and simplistic. The downside is that an ergonomic handle is a very complex thing to both design and manufacture.

6.2.2 Double Handle

The double handle design was conjured when we realized that not only would it be unnecessarily difficult to create a crescent shaped screen that can wrap around the toggle in the single handled design, but we also enjoyed the idea of playing the game with both thumbs instead of just one. The design is made such that the user holds the device with both hands, and there is room for a standard screen, which can display additional information about the game. This design however, still comes with the same manufacturing difficulties as the single handle design.

6.2.3 Portable Console

The last design iteration takes inspiration from gaming devices like the Nintendo Switch™, the Steam Deck™, and the PlayStation Portable™. In order to avoid the complexities of manufacturing an ergonomic handle, and the awkward shape of a screen above the main body of the controller, we decided to consider a shape that more resembled devices with similar peripherals. Lacking in handles and having only one main encased area, this design simplifies hooking up the peripherals such as the toggles and haptic feedback as well.

6.2.4 Final Decision

The initial designs A and B were deemed too difficult due to the strange shape of the screen in A and the unnecessarily hard task of 3D printing either design. Designing aside, the difficulty in manufacturing an ergonomic handle stems from our 3D printing abilities. While it is possible to print unusually shaped objects, our group had no experience 3D printing with any amount of scaffolding when planning our project. Due to this, we prefer to stick with a design where parts have at least one large flat side in which to use as a structural base when printing. We decided on design C because it fit our final idea of how the game would play and holds all the components necessary while also being relatively simple to print and put together.

6.2.5 Designing for System Construction

Beyond the overall design that the case will take on for user experience, the case also needed to be designed with construction of the entire system in mind. The first piece of this puzzle is that the main shell of the case is three pieces, a front, center, and a back. These three main pieces will be secured using the threaded inserts and screws discussed in the System Fabrication Chapter. Designing for these inserts involves two things: the piece that the screws are passing through need a hole and chamfer, and the piece that the threading is inserted into needs an indent which size is specified by the threads manufacturer. The location of any construction hardware must also have not interfered with the ergonomics of the case, such as having the user's hand rest on a screw.

In addition to connecting the front, center, and back, the case needed to be designed to accommodate all of the components involved in the design. Many of these design decisions were made as we moved along in testing the system, the considerations we had before making the case are as follows:

- “The display will most likely need its own piece to keep it in place. The front of the case may likely have an indent that the display will fit into, with a separate frame piece that fits around it and secures to the front of the case locking the display in place.”
- “The two main boards will both need to fit into the case without encroaching on each other. To secure the main PCB with the MCU and toggles, raised holes will be designed into the inside of the back panel of the case. The display driver board will have to be fit between the display and the main board.”
- “The toggles will be soldered onto the main PCB, as per the Senior Design class requirements. To make sure that the toggles are stable, extra PCB mounting will be required at each toggle; this is to prevent the PCB from bending and the user from feeling any give in the construction. In addition to the PCB mounting holes, the front of the case needs to have holes large enough for the interactable portion of the toggle module to stick through. Since the toggles have a rounded base, the diameter of the exit hole depends on how high we decide we want the toggle to be relative to the front of the device.”
- “The speakers have one main requirement of the case: that the sound can escape and reach the user. This can be achieved by designing a speaker grille into the front of the case which sound can escape through. The speakers must also be mounted close to the grille, which can most likely be achieved using something like polyurethane adhesives.”
- “The haptic feedback modules have the highest variety of possible locations inside the case. Wherever we decide they feel the best, they will most likely be secured the same way as the speakers using polyurethane adhesive. One possible design choice is that the module may need to have more contact with the area of the case that the user holds, which would manifest as a cubby shaped into the case that the haptic feedback module snuggly fits into.”
- “The battery will need a solid spot to sit that also doesn’t throw off the balance of the system. Given that the rest of the design is symmetrical, it may become necessary to have the center back of the case thicker than the handles. This way the battery may sit in the center and not mess up the ergonomics by making one side heavier than the other. Once the overall design is finalized, we can experiment with adding a user-friendly way to replace the battery to the back of the case. Most likely this will be a hatch that is secured by a screw on one side and two tabs on the other side, and behind that hatch there is a secluded compartment where the user can reach only the battery to easily swap it.”

6.3 Schematic

The design for our project is based around the ATmega328P as the central coordinator for all components. This section will detail how each component is organized in the schematic. Much of the schematic design is borrowed from the Arduino Uno, since it is open source. We removed the elements that we didn't deem necessary for our project, such as the USB, the female headers, and the reset button. We've also added elements that were required in order to use the toggles and haptic feedback. In total, our design requires fifteen pins: four analog pins, six digital pins, and five display pins. In figure 8 we showcase the overall schematic, and in the following sections we will explain the individual areas of the design.

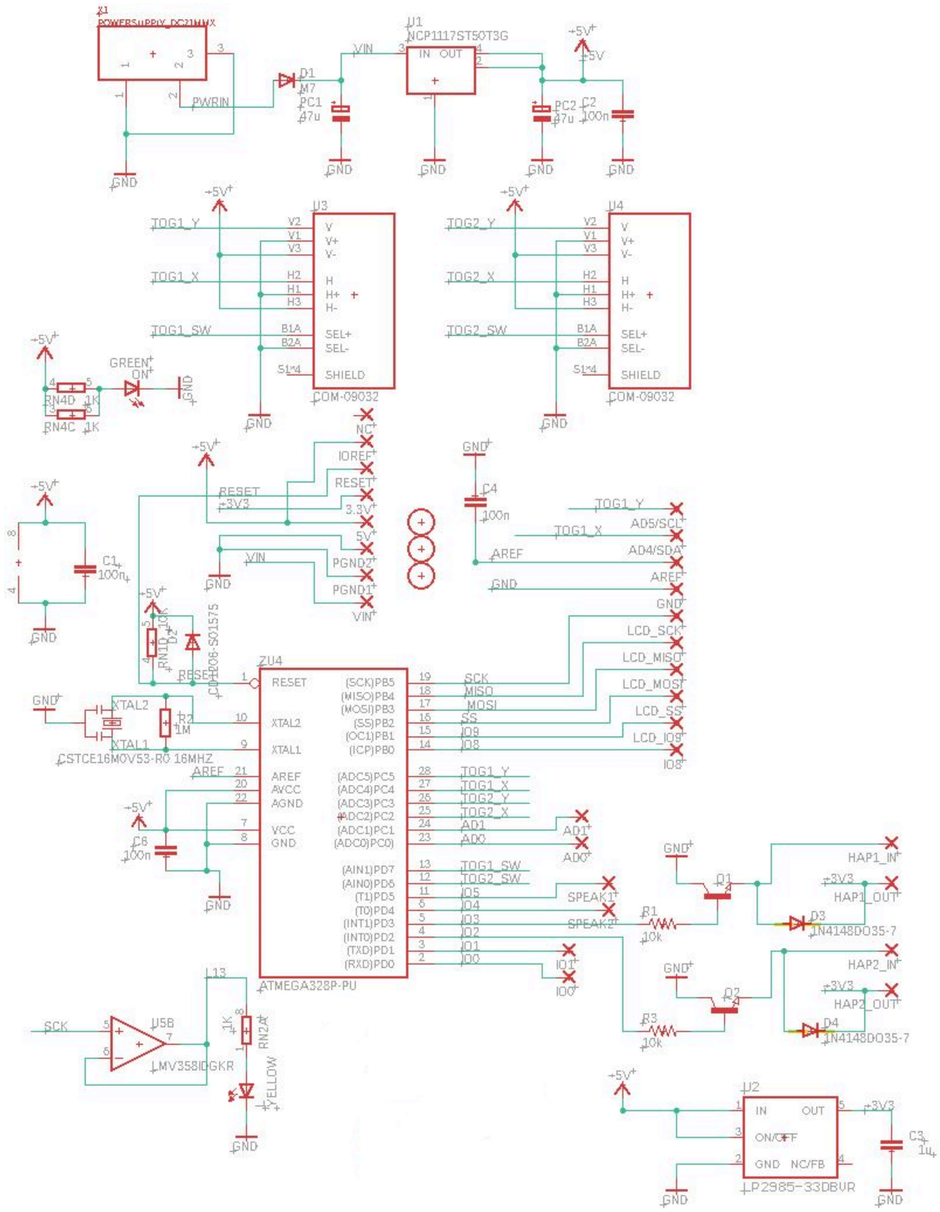

Figure 8 - Tentative Overall Schematic

6.3.1 Power Delivery

To provide the correct levels of voltage for the system, the power delivery portion of the schematic is designed to provide the system with 5 volts and 3.3 volts. The power delivery schematic is illustrated in figure 9 below. The VIN is delivered to the board via X1, the DC jack, passing through the M7 diode for reverse polarity protection. VIN first enters the buck converter U6, then the 3.3 volt regulator U2.

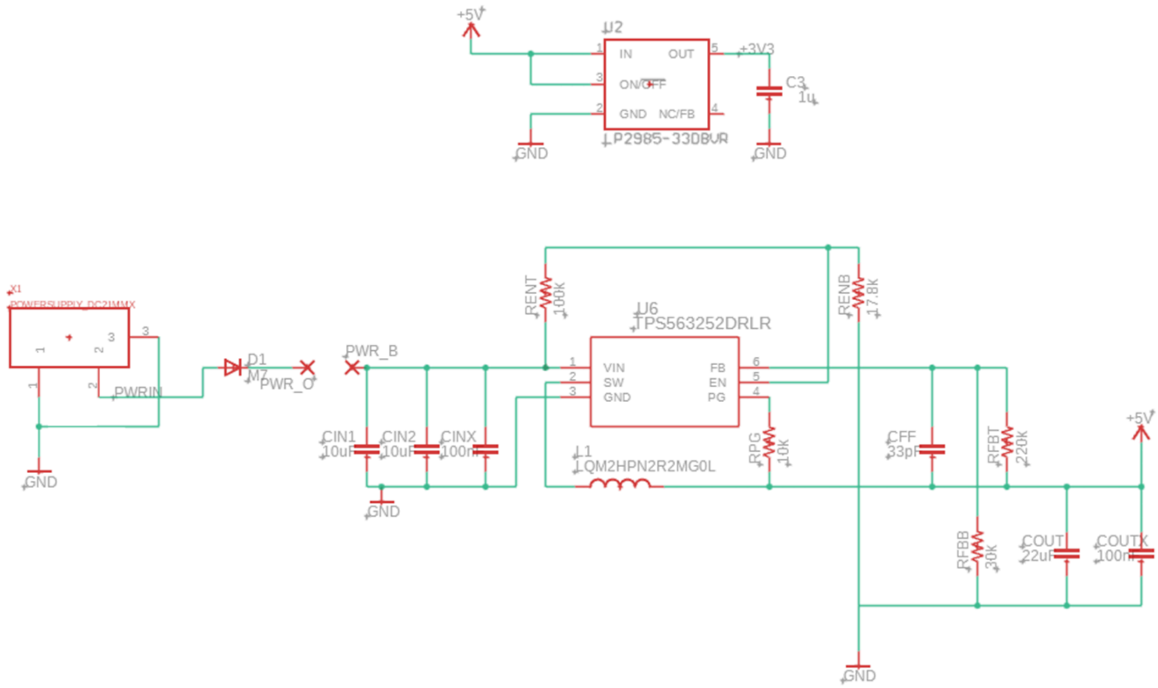

Figure 9 - Power Schematic

6.3.2 Toggles

The toggles are the main element on our board that requires analog pins. The toggle portion of the schematic can be seen in figure 10 below. Each toggle requires two analog inputs, one for the x axis and one for the y axis. The reason each axis needs an analog pin is because the axis measurements are made using mechanical potentiometers. In addition to the two analog pins, each toggle requires one digital pin for the button.

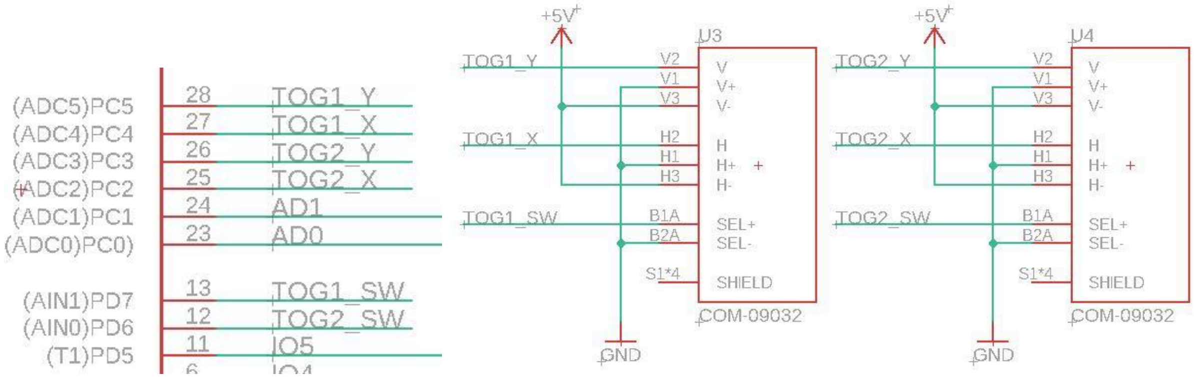

Figure 10 - Toggle Schematic

6.3.3 Haptic Feedback and Speakers

The haptic feedback modules each use one digital pin, as seen in figure 11. One of the schematic requirements of the haptic feedback modules is that in order to control them with the digital I/O pins an NPN transistor had to be used. This is because the haptic feedback modules are motors, and require a higher current than is recommended to be sent through the ATmega328P’s I/O pins. So rather than connecting straight to the motor, the pin that controls each module is connected to a 10kOhm resistor and the base of the transistor. Another requirement of the haptic feedback modules is the diodes D3 and D4. Motors are inductive loads, and so a diode is used to safely let the electromagnetic field of the motor collapse. The speakers are simple, one wire connects to a digital I/O pin and the other wire connects to ground.

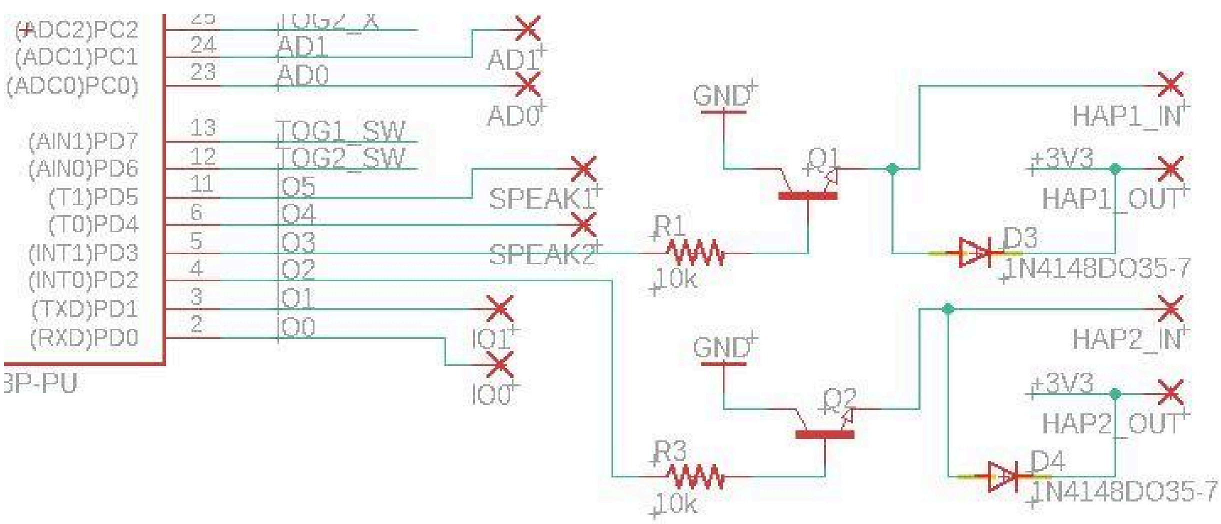

Figure 11 - Haptic Feedback and Toggles Schematic

6.3.4 Display

The schematic for the display is very straightforward. Due to the display having its own driver board, our schematic only needs to have solder points to wires to the correct pins of the ATmega328P. The layout of these pins is illustrated in figure 12 below. The display driver board is the RA8875. The RA8875 communicates with the ATmega328P using SPI and two additional pins. The four pins used for the SPI are pins sixteen through nineteen; these are, respectively, the chip select pin, the serial out pin, the serial in pin, and the SPI clock pin. In addition to this, we have the reset pin of the RA8875 hooked to I/O pin nine. Since we are not using the touch screen functionality of the display, INT, Y+, Y-, X-, and X+ on the RA8875 are not currently hooked up in the schematic. The design currently has four of the six analog pins in use by the toggles so the four coordinate pins that the display driver cannot be hooked up later, should we decide that we do want to use the touch screen functionality after all.

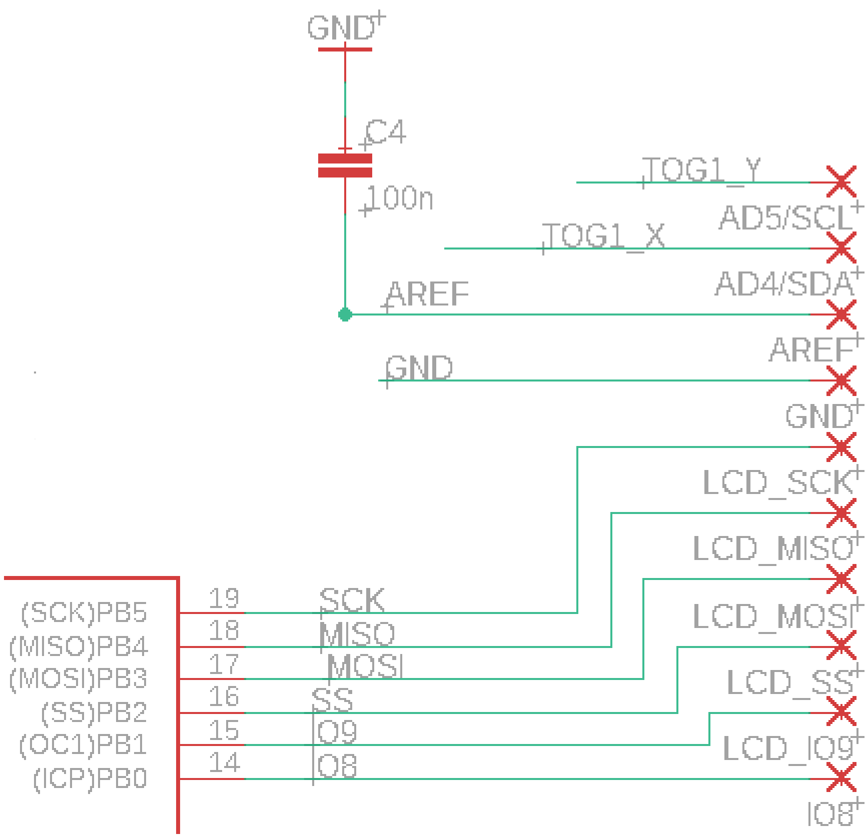

Figure 12 - Display Schematic

Chapter 7: Software Design

In the upcoming sections we aim to bring a greater understanding of the user interface of the game as well as a deeper understanding of the software itself. We will break down the two largest screen states being the main menu, and the in-game screen. Both of these states are key to understand as they will both be hubs for our game.

7.1 Main Menu

When the system is first powered on the user will see the main menu, from here they can change settings, see the leaderboards, and start the game. The menus are navigated using the right toggle and selections are made using L3 (clicking the right toggle.) The leaderboards section will allow the user to scroll through the top three recorded scores and see the initials that the player entered for that score.

The setting options that the user will have is to change the game mode, there is a left, right, or both toggle option. Each toggle option has its own scoreboard in the leaderboards section. The user is also able to clear all the previous highscores from the settings menu.

7.1.1 In-Game

Once in-game the user can only input on the toggles that correspond to what mode they are in. There is no pause as repeatedly pausing and unpausing the game would bypass the difficulty. Once the user runs out of time and enters the game over screen, they will be asked to enter their initials using the toggles for the leaderboards if they achieved a high score. After the user enters their initials they will be sent back to the main menu.

7.2 Software Diagrams

In the following sections we will be diving into our project's software architecture. The following software diagrams will serve as the framework for this project as well as a guide that we can continue to depend on. These diagrams will encompass every aspect of the software that we have used, both from the user's standpoint as well as from the device standpoint. Due to the complexity of this project, not everything will be labeled in the following charts as placing every item into the charts may be too tedious and excessive pulling away from the original intent of these charts which is to allow the reader to have a deeper understanding of our goals.

7.2.1 Flowchart diagram

Below we have a visual representation of the most important software processes that will be happening within our game when a user first begins playing. As the diagram shows, when the user first powers on our game the program within the device will load the class that we assign to hold the main menu. This will include all styling elements that we have implemented as well all the logic that will allow the user to navigate around our main menu screen. We have included a way to track the users inputs through the toggles and then from this menu the user has two options. They can either continue through to the actual gameplay of the device, they can navigate to a screen that will display overall high scores that have been set on the device, or they can access the settings menu that was previously mentioned.

If the user chooses to do the first option to begin game play the device will begin to randomly create a level. We have each level a unique level every time the user plays the game. We do this to create a diverse game that can be played many different times and not feel stale. The idea to do this led us in the decision to randomly generate each level every time. The ability to accomplish this was to have each arrow assigned a number in an array, then when the level is generated it will randomly generate a set of numbers which will correspond to the arrows and colors allowing for randomly set colors for each level. After this is accomplished, a specific arrow that needs to be cleared will be assigned a random number for each level. This allows us to keep the levels fresh each and every time. The next task to get the user to the first level of gameplay was to calculate the location of the toggles. When the user is using the device it will be constantly updating the location of the toggles. This not only allows us to keep track of the toggles during the entire time the device is being used, but also be able to find the toggles quickly every time the device changes screens which is very important to allow for a seamless user experience. Finally, we have a timer set that will continuously loop through during the game, speeding up when there are ten then five seconds left in the game allowing for an audio queue for the user.

In another case where the user selects the option to view the leaderboards, the device is tasked with a few different operations. The first operation is to find the high score values and get them ready to be loaded. Assuming that the high scores are not already stored in high to low order, we must sort it every time they are loaded. After these two tasks are done they can then be displayed onto the display for the user to view. When this occurs the device must also load all of the stylistic features as well, we were thankfully able to reuse a lot of the styling choices from state to state. This was ideal in order to preserve labor time and also to help reduce memory usage.

In the final case where the user selects to access the settings menu they will be then brought to the settings screen. From here they can either change the gamemode to left, right, or normal which just changes a boolean value. This provides a wide range of difficulty widening the range of players who can play our game. After the user selects to go to this screen they will be able to choose one of these options. After the user selects their option they will be taken directly back to the game mode screen where they will be presented with all of the original options. In this settings menu they can also clear the high scores which will set all player high scores and players initial values to null.

After the level ends the device will decide if the user passed or if they did not. If they did pass then the previous process of generating the level will repeat until the user does not pass. When the user inevitably runs out of time a new set of tasks will begin. First the device will prompt a saved “Game over” screen. After this the device will pull the previously high score and compare it to the score that the user just achieved. If the user achieved a score higher than the previously set high score, then that will be plugged into that designated spot. From there the device will load a congratulatory screen for the user to see saying that they achieved a new high score. The final step was to reuse the previous pattern of loading the high score values, placing the scores in order and then displaying the usual high score screen with all of the previous scores. Finally, if the user does not achieve a high score they will simply be brought back to the main menu screen where they can then decide if they would like to replay, view the high scores or change settings.

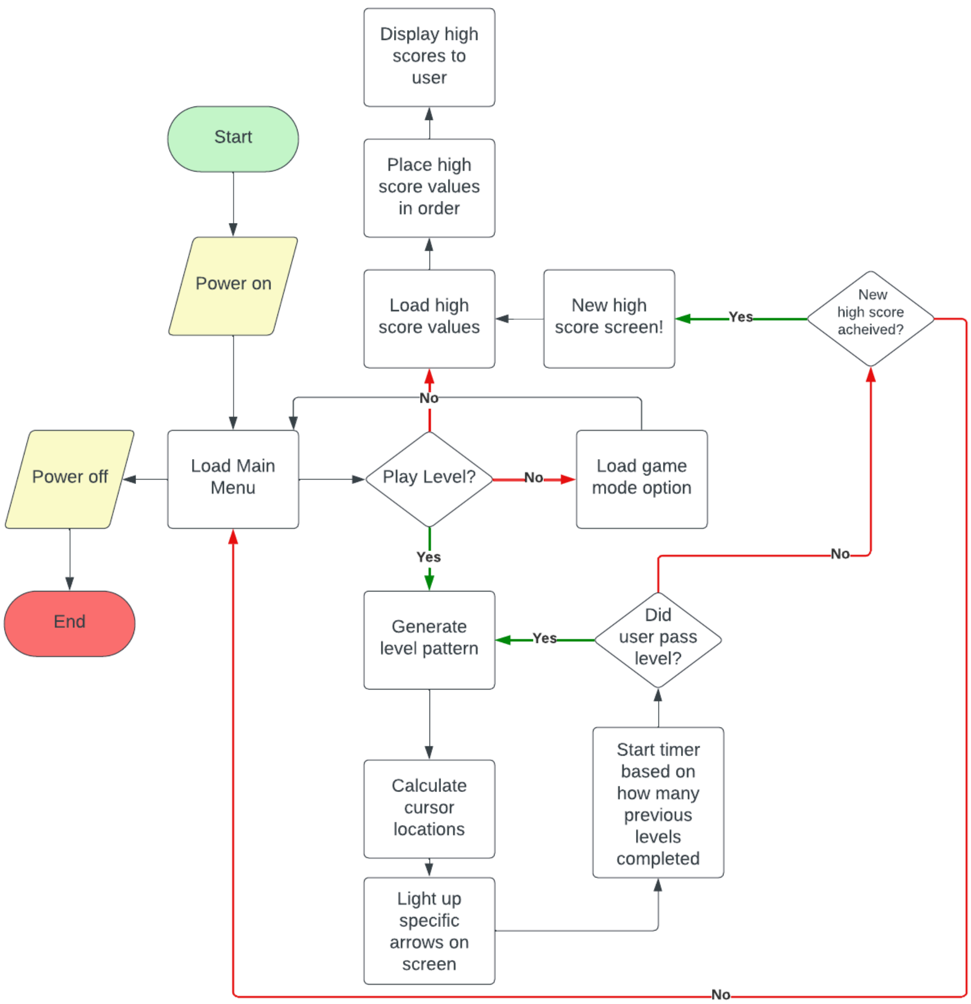

Figure 13 - Flowchart diagram

7.2.2 Use case Diagram

In our use case diagram presented below, we provide an overall visualization of the relationships between the user and the game system. The diagram served as a roadmap, showing the seamless flow of actions between the user and the game. We have color-coded our representation highlighting the specific responsibilities of each party involved. We were able to create a user-centric approach, ensuring that all physical actions within the device are initiated and controlled by the user, thereby excluding any autonomous movements facilitated by motors or other devices.

After further viewing of the diagram, it becomes evident that the user-driven actions encompass a wide variety of actions. These actions range from manipulating the toggles on our device to engaging in gameplay across various levels, accessing the main menu screen, changing the difficulty, and viewing high scores from the main menu interface. Each of these user-initiated actions contributes to the immersive and interactive experience we aim to deliver. Conversely, the device itself undertakes crucial software-related responsibilities. These include the generation and randomization of game levels, meticulous scorekeeping, seamless display of the main menu, and the continuous monitoring of toggle positions. By distinctly setting these roles, our design optimally distributes functions between user engagement and device operation, ensuring a flawless design.

While the use case diagram prominently features these key components, it is essential to acknowledge that the software entails a variety of smaller functions working together behind the scenes. However, due to spatial constraints and a commitment to clarity, the diagram intentionally focuses on the fundamental and overarching functionalities, avoiding unnecessary intricacies that might overwhelm the viewer. This thoughtful balance ensures that the diagram remains a comprehensive yet accessible tool for understanding the fundamental dynamics of user-device interactions in our gaming system.

Figure 14 - Use Case Diagram

7.2.4 State Diagram

In the comprehensive state diagram provided below, we guide you visually through the various key states that our device can assume, offering a detailed insight into the user experience. Upon powering on the device, users are welcomed by the home screen, where they are presented with two distinct options: the opportunity to view other previously achieved high scores on the device, change a few settings of the device or to go straight into gameplay. If the user selects to change the settings they will then be presented with a new settings state, and the same will occur if the user opts to view previous high scores. After the user is finished with both of these states they will be redirected to the main menu screen. The journey through the levels is intricately tied to the user's performance, creating a dynamic and responsive gaming environment. As users navigate through successfully completed levels, they are rewarded with a seamlessly transitioned experience, encountering new randomly generated levels to maintain the flawless gameplay. This will continue until the user runs out of time. The inevitable conclusion of this leads the user to a game over screen.

Post-game over, the device dynamically evaluates the user's performance by comparing the newly achieved score with the high score stored on the device. If the user has surpassed their previous high score, they will be presented with a high score screen, acknowledging their accomplishment. Conversely, if the score falls short, the user seamlessly returns to the main menu, providing them with an opportunity to regroup and commence a fresh gaming experience. This thoughtful progression ensures a captivating and user-centric gaming journey, where each state serves a purpose in enhancing the overall engagement and satisfaction of the gaming experience.

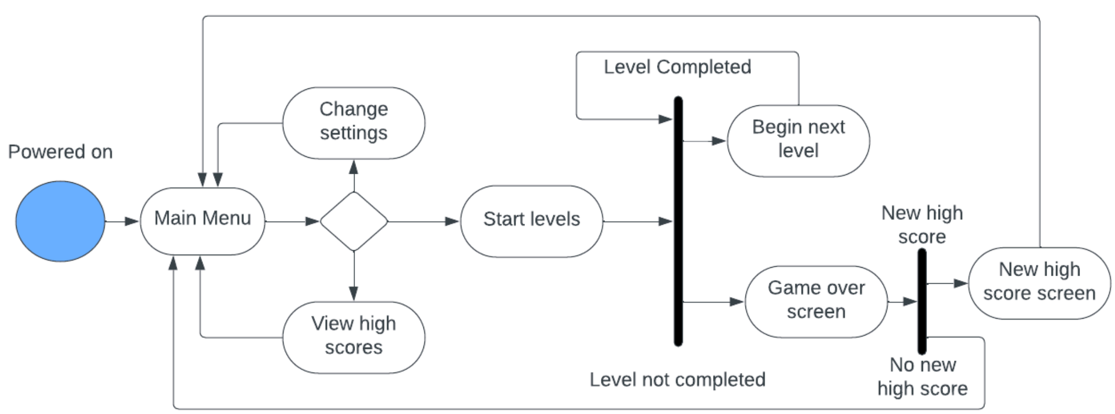

Figure 15 - State Diagram

7.2.5 Class Diagram

In the upcoming class diagram, we aim to explore the fundamental classes essential for the effective operation of our device. Each class will be examined closely, shedding light on its specific purpose and interactions with other classes. This detailed depiction serves as a blueprint, outlining not only individual functionalities but also the collaborative relationships contributing to the smooth execution of our device.

The class diagram, as a visual representation, delves into the device's architecture, showcasing the flow of information and control among identified classes. By scrutinizing key roles and responsibilities, stakeholders and developers gain an understanding of data processing, action triggers, and overall device functionality. This diagram acts as a tool for communication and collaboration, facilitating the identification of dependencies, optimizations, and areas for expansion. In essence, the class diagram encapsulates the relationships and interplay defining the intelligence and efficacy of our device, moving beyond its role as a static representation.

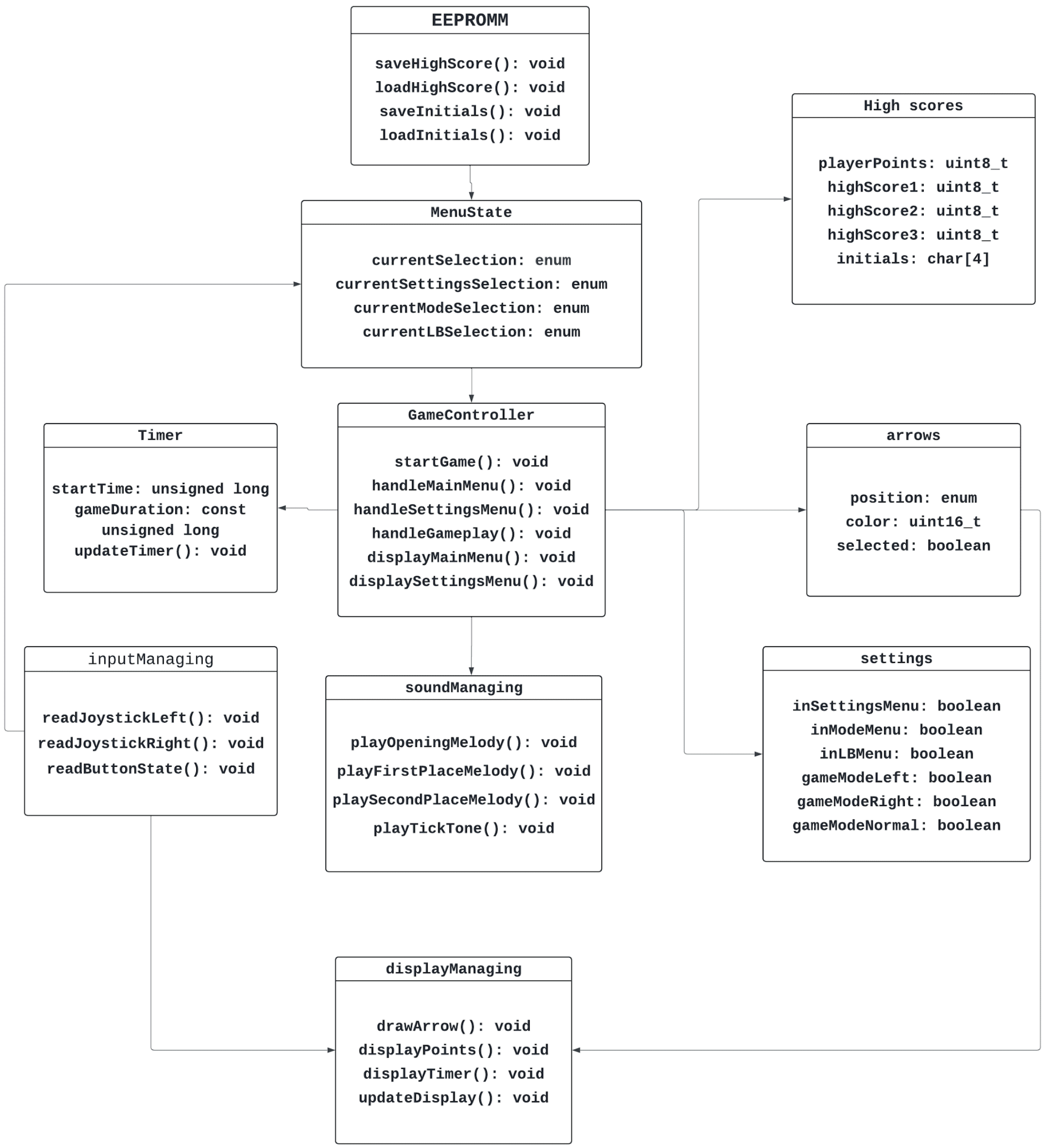

Figure 16 - Class Diagram

In this class diagram we show a basic breakdown of most of the classes that we needed. We used the menu state as a hub where the user begins every session on the device. This main menu includes styling and also different choices for the user to select. On this page we had to use coordinates for the toggle so that way the user is able to select the choice they desire. This was done with a toggle x variable and toggle y variable that we call in every instance that we need the toggle coordinates to be present. As stated in the previous sections the user can select to go to a high score screen, change settings, or go to a new level when they first begin the game. If they go to a new level we will call the class that generates a new level. This class will use the arrow class that creates the individual arrow and it generates multiple of those arrows as well as calling the count down timer class which will calculate the length of the timer that is needed for this instance of the level based on previously played levels. There is more smaller classes that are needed to be called in order to have the level displayed as desired but including every small class into the diagram would be excessive. If the user opts to choose the high score screen the high score class will be called which includes all of the appropriate information regarding the history of the high score as well as continued styling that is similar to the main menu.

Chapter 8: System Fabrication

8.1 Case Construction

Every step of the case construction was done by our members with resources that they already have access to.

8.1.1 3D Modeling and GCODE Translation

The first step in constructing the case was modeling the case in a CAD program. For this we used Autodesk Inventor, which one of our members has access to. Once each iteration of a case piece is good enough to be tested, the model is saved as a .stl and taken to a program that can translate the model into specific instructions for the 3D printer. The program used to do this is CURA Lulzbot Edition. Once the model is imported into CURA, oriented correctly, and material chosen it can then be exported as a .gcode and is ready for the printer.

8.1.2 3D Printing and Fastening

The 3D printer that our team used for the project is a modified Lulzbot Taz 6. It has a BLtouch installed for precise bed leveling and is running Drunken Octopus as its firmware. The print volume is 11.02” by 11.02” by 9.80”, which is more than large enough to print any of the pieces we need.

Each of the three large pieces of the case, the front, center, and the back, were printed one at a time. The prints took about four to seven hours each. Any smaller pieces that were needed, such as the display scaffold, could be printed in batches based on necessity. The pieces that were done in PLA needed to be printed at 185°C.

The large pieces were fastened together using threaded brass inserts and screws. The inserts are small tubes that are threaded on both sides. The inner threading is a standard screw size and the outer threading is a crossing pattern that helps secure the insert in place once the plastic has cooled. The way the threads are inserted is by placing them on to the tip of a soldering iron until they have heated up enough to melt the plastic, then they are pressed into the designated spots on the plastic. Once the soldering iron is removed, the insert and plastic will cool down, securing the insert in place.

8.2 PCB Layout

The biggest consideration in the fabrication of the PCB stems from the Senior Design PCB requirements. In order to meet these requirements our toggles must be attached to the PCB that we design, as opposed to being on a breakout board and wired. This has a significant effect on the size requirement of our PCB, as we have one toggle on each side of the display. This means that our PCB must be about six or more inches long to reach. Another choice we've made is to use the ATmega328P-PU variant of the microcontroller. This allows us to take the chip out of its socket and program it by plugging it into an Arduino Uno. This simplifies the schematic and PCB by allowing us to not worry about any sort of USB to program through. Should we end up needing to program directly to the board anyway, the pins required all have solder points available on the PCB. Another consideration that has to do with the toggles is ruggedness. For this we have included four mounting holes for each toggle, so that there is no give when the user is interacting with the toggles.

The final version of our PCB design can be seen in figure 17. The bottom left and right have the solder points for their respective haptic feedback module and speaker. The top middle has the solder points for the pins associated with the display driver. There is one toggle on each side of the board. The power jack is located in the top center so that the plug does not overlap with any of the parts.

The final version of our PCB took three tries to get correct, our initial PCB had the transistor for the haptic feedback modules hooked up incorrectly, and the second iteration had one trace in the 5V power circuit hooked up incorrectly.

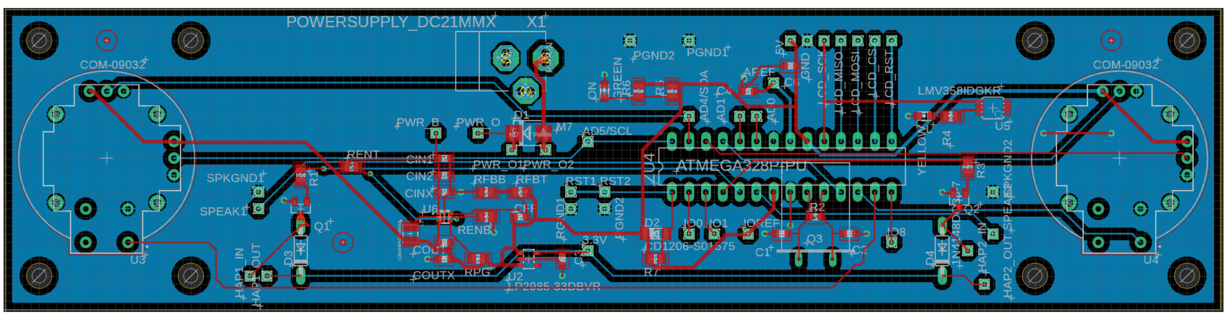

Figure 17 - Final PCB Design

Chapter 9: System Testing

9.1 Hardware Components

Most of the components we have chosen can be tested quickly by simply plugging them in and playing with them for a few minutes. There are, however, a couple components that needed to be tested with a system that was physically very similar to the finished product. This is due to the nature of being a game; for the components to truly fulfill our requirements they must have made the game feel responsive and satisfying to the player. The following figure is a capture of the group testing the components.

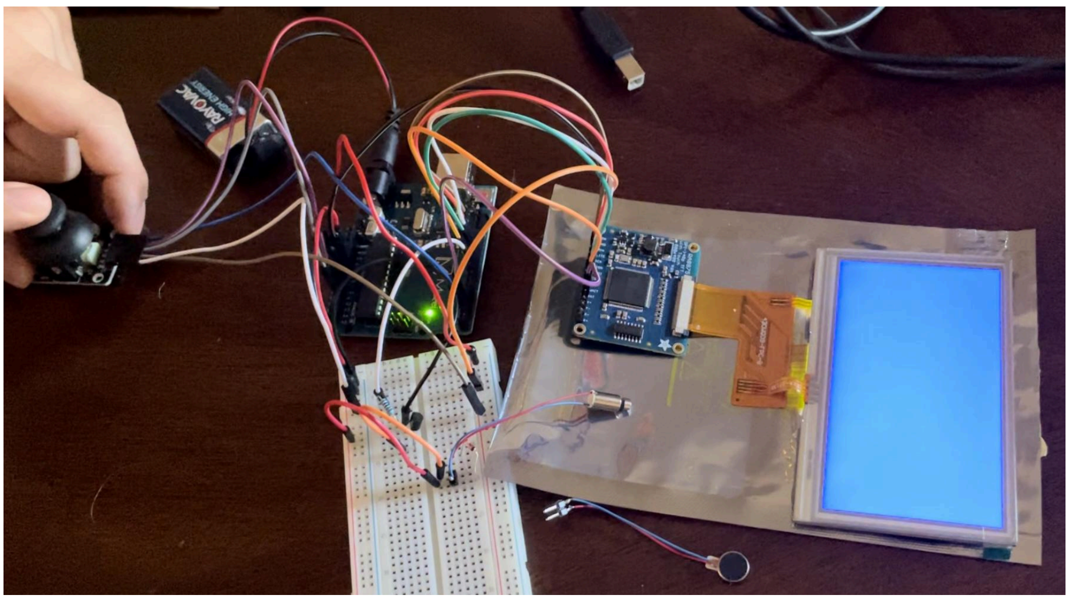

Figure 18 - Testing the Toggle, Display, and Haptic Feedback Simultaneously

9.1.1 Display

The display is the most integral part to the user's experience when playing the game. Despite its importance, testing the display is a rather simple feat. We compiled code on the Arduino IDE specifically meant to ensure the display and touchscreen work. When the code is run, the screen starts up and displays various colors and patterns to showcase that the display is functioning as intended.

9.1.2 Single Board Computer

The single board computer was initially tested just by turning the device on and off to ensure that it has power and can work. More further testing was then connecting components to the board to check if the code can run and other components be controlled. Beyond that, the testing would be to see if it can run every component running in tandem.

The final test of the microcontroller unit was when it was running the actual game itself and seeing if it can maintain all the user inputs in a way that feels satisfying to the user, as well as keeping up the frame rate so the game is visually appealing.

9.1.3 Toggles

To test the toggles, we connected the toggle to the Arduino board by connecting the ground pin of the toggle to the ground of the board, then the 5V pin to the 5V power to power on the toggle. Then the VRX pin and VRY pin to the analog pins in order to read the output corresponding to the X and Y coordinates that measure the sticks horizontal and vertical position on the axis. The final pin connects the SW pin to a digital pin, and this pin is meant to read the thumbstick pushdown button input. Once all the pins are connected, we then connect our LCD display to the Arduino and link the toggle to the display through the Arduino IDE. When we push the toggle up, down, left, or right, the LCD display would change color depending on the direction the toggle was pushed. And when the thumbstick is pushed down, it would also change the display to a different color.

9.1.4 Sound Device

Testing the sound device component emerged as the initial and relatively straightforward phase in our project development. This process can be done with confidence, as it can be done without developing much of the rest of the system. All that was needed was the Arduino Uno, the speaker to test, and two wires to connect the speaker to the power and ground of the Arduino Uno. Employing the Arduino IDE, we loaded an example sketch that is provided by the IDE to test the audio using the board. Once loaded, we adjusted the code to match the required digital pin we intended to use for the speaker, and compiled it onto the board. We reset the board in order to get the code running and tested the speakers one by one, to see which one we preferred more and also ensure that they were functioning as intended. We compared the audio quality of them using the same sounds, and tested for audio output, whether the speaker needed audio amplification, as well as noise vibration, since some speakers can be loud causing vibrations. The speakers worked as intended and played the IDE example code audio pretty well. The issue that we ran into was when we tried to play a custom sound, and converted the file to match the rate and bit size playable of the Arduino. Once we ran and tested the converted audio, it was much lower and slower than the original audio. Due to this, we considered that we might need to incorporate a microSD reader into our design in order to improve the sound quality of any audio we decide to throw at it, but ultimately decided it to be unnecessary.

9.1.5 Haptic Feedback

The haptic feedback modules were the most difficult piece of hardware to test. Initially we tested the haptic feedback by connecting it to a transistor and resistor and then powering it up using the Arduino in order to confirm basic functionality. We then compiled code for it so when the toggle is pushed down, the haptic feedback will turn on and cause vibration. This showcased to us that it was working, and functioning as intended. The role of the haptic feedback modules is to provide the user with satisfying feedback based on what they do in the game. The tricky part of this is that to know how they really feel when integrated into the system we needed to have them inside a case that roughly resembles the finished product. The following figure is a picture of this being completed

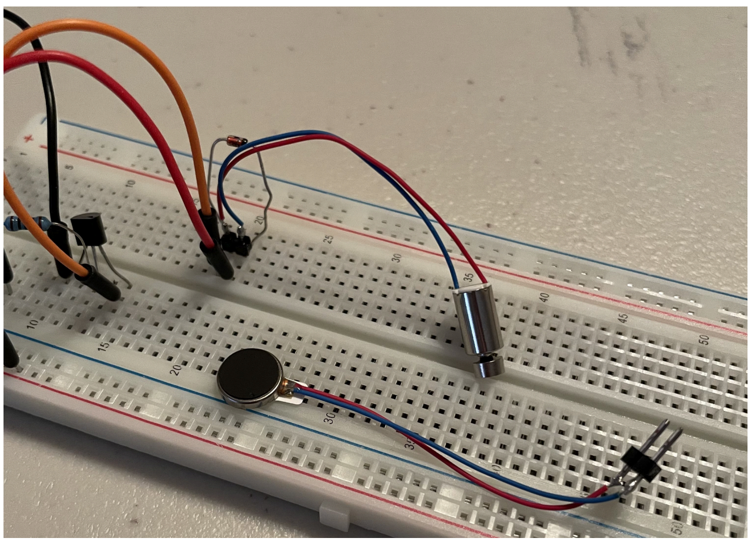

Figure 19 - Testing the Haptic Feedback Modules

9.1.6 Battery

Initial testing of the battery was simply using it to power the Arduino Uno that we've used to test the rest of our components. However for more thorough testing we needed all of the components running at their expected intensity to see if it can handle the load required. At first glance it might seem like we would need to have the game fully running with all of its graphical bells and whistles to test this, but in order to get the testing done sooner we simply used a graphical benchmark. The main draw of power from the battery was expected to be from the graphics processing and screen. This was actually false, the haptic drew 43 more mA than the display. To test this we used a benchmark that is roughly the same level of graphical intensity as our finished game. Doing this we were able to fully test the battery without having to finish the program. The next expected main draw of power was the haptic feedback modules which described before was actually the leading power consumer. In order to test we had them run at the same time as the graphical benchmark.

9.1.7 Case

Initial testing of the case was rather simple, it just needed to hold all of the components. Further testing required that the case felt good for the user to hold. The case needed to not be rough where the user's hands rest, but not so smooth that it's hard to keep a grip. Additionally, the case needed to balance roughly in the center or lower half (relative to facing the screen.) If the device was top heavy it would make it awkward for the user to hold. The case also had to be fairly rugged and able to handle the physical stress of the user quickly moving both toggles. We initially tested the ergonomics of the case by pretending that we are playing the game, but the true test required us to see if any discomfort caused by the case distracts the user from the game they are trying to pay attention to. For the ruggedness, we tested the case by shaking it a reasonable amount and randomly jerking the toggles around a little bit more roughly than we think would actually be done by the user.

9.2 Software

There were many different tests for the software of this device. We went through many iterations of the final program. This was due to us creating and coming up with new ideas as we go along since this was an ever evolving idea. There were a few main tests we did in order to ensure the final product is up to our standards. The first type of tests we ran were unit tests. Unit tests were designed to evaluate the functionality of individual units or components of a software application in isolation. Their purpose is to ensure that each unit of code, such as a function or module, performs as expected and continues to do so even as the software evolves, providing early detection of bugs and facilitating easier maintenance. We used unit tests to ensure that our program was working to a high quality and as efficiently as possible.

Another way we tested the software was through front-end testing and involving third-party individuals. We had outside testers try our game, deliberately executing numerous random inputs. This approach ensured thorough testing of all potential edge cases that could occur on our device.

9.3 Overall Integration

The overall integration of the system was tested two times: once before the PCB and once after. Before the PCB we connected all of the parts together on a breadboard and tested that the pins would communicate with what we wanted. After we received the PCB, we did essentially the same thing except with a focus on making sure the power and battery was working correctly.

9.4 Senior Design 2

Much of the aforementioned testing was completed in Senior Design 2, especially any that required the game to be remotely functional programmatically. There was a logical order to what we had to follow this semester: first we had to get the software running a barebones version of the game, enough to test the speed but sound effects and fancy graphics can be done later. Next or even parallel to the barebones software we made a roughly accurate case that is modified to fit the Arduino Uno inside of it, so we can test the integration of the parts. Once the Uno sized case was usable, the software became the main focus, so we could test every facet and became confident enough to order the PCB. Once the PCB was ordered, the focus became getting everything inside of the final version of the case in a sufficiently rugged manner.

Chapter 10 Administrative Content

10.1 Budget and Financing

Since our group did not have a sponsor all of our funding was provided by ourselves. With this in mind we heavily considered price when comparing components against each other. The below table is a rough breakdown of what we spent in total. An important thing to note is that this table does not include any devices that we purchase strictly for testing, there are multiple devices on this list that we purchased duplicates of in order to do thorough testing on. We have not included these extra purchases into our total cost because we did not know at the time which devices we will have to purchase multiple of. The table below shows a full breakdown of all the planned purchases at this point.

Item Description	**Quantity**	**Total Estimated cost**
Display	1-2	$45
RA8875 Driver board	1	$40
Toggle	2	$20
Speaker	1	$10
Controller (casing)	1	$35
Printed Circuit Board	1	$50
Microcontroller	1	$20
Haptic feedback	2	$4
Total estimate		~ $224

Table 15 - Estimation of total product cost

The following table is our tentative bill of materials for our pcb. The reason this is tentative is because the pcb requirements may change once we finalize our design.

Tentative Bill of Materials		
Part	Qty	Price Per
ATmega328P-PU	1	$2.89
PU Mount	1	$0.44

0.1μF Capacitor	5	$0.10
10kΩ Resistor	3	$0.10
09032 Toggle	2	$4.50
NPN Transistor	2	$0.10
1N4148 Diode	2	$0.10
3.3 V Regulator	1	$0.50
Diode 1000V 1A	1	$0.10
Power Jack	1	$1.82
Yellow LED	1	$0.30
OPAMP	1	$0.57
1MΩ Resistor	1	$0.10
16MHz Resonator	1	$0.25
47μF Capacitor	2	$0.35
5 V Regulator	1	$0.68
Diode 100V 1A	1	$0.44
1kΩ Resistor	2	$0.10
Green LED	1	$0.25
Total Price		$19.44

Table 16 - Tentative Bill of Materials

10.2 Project Milestone

Displayed below is a comprehensive visual representation outlining the large milestone tasks that we have committed to accomplishing. This table served as a crucial reference point, guiding us in determining the timelines for the completion of specific assignments. Over the course of the semester, we diligently marked off completed tasks, actively engaging with the progress we made. Additionally, this table functioned as a key reference tool for managing the deadlines of our larger-scale projects. While not all smaller tasks are explicitly articulated in this breakdown, it is important to note that our operational strategy acknowledges the existence of these more granular objectives. However, due to the sheer volume and intricacies involved, providing an exhaustive list of every individual task was not practically feasible or useful.

Task	Complete by date	Status
Project discussions	August 25th, 2023	Complete
Project Decision	September 4th, 2023	Complete
10 page Divide-and-Conquer report	September 15, 2023	Complete
Component research and decision	September 28th, 2023	Complete
PCB research	October 12th, 2023	Complete
Bill of Materials	October 20th, 2023	Complete
Table of Contents	October 28th, 2023	Complete
45 Page Documentation	November 3rd, 2023	Complete
Final 90 Page Documentation	December 5th, 2023	Complete
Obtain parts	During winter break starting December 2nd, 2023	Complete
Senior Design 2	-----------------------------------	-------------------------------------
Individual part testing	December 1st, 2023	Complete
Assembly	March 5th, 2024	Complete
Full testing and redesigning	March 20th, 2024	Complete
Final testing	March 1st, 2024	Complete
Presentation	March 18th, 2024	Complete

Table 17 - Project Milestone list

10.3 Work distributions

Presented below is a comprehensive table showing our distributions of tasks that formed the core of our project. In light of the fact that we operate as a relatively small group being only three of us, we adopted an "all hands on deck" methodology. This implies that no single task is exclusively the responsibility of one individual; rather, we fostered a collaborative work environment where each member actively contributes to every task, offering assistance and support as needed. Despite the existence of distinct specializations based on individual backgrounds and selected areas of study, it is important to note that these areas of expertise do not confine or limit any team member to solely their designated tasks. The teamwork we embraced emphasized cooperation, encouraging and empowering team members to ask for help whenever they need it. This willingness to collaborate ensured that the expertise of the entire team was readily available, creating an environment where collective success took precedence over individual accomplishment.

Task	**Primary person**	**Secondary person**
PCB research	Linus Fountain	Eric Espinosa
PCB design	Eric Espinosa	Linus Fountain
Case design	Linus Fountain	Eric Espinosa
Case 3D printing	Linus Fountain	Eric Espinosa
Acquire parts	Josh Bell	Linus Fountain
Test display	Josh Bell	Eric Espinosa
Test display driver	Josh Bell	Linus Fountain
Test haptic feedback	Linus Fountain	Eric Espinosa
Test speakers	Eric Espinosa	Linus Fountain
Write program	Josh Bell	Eric Espinosa
Finalize design	Linus Fountain	Eric Espinosa
Build device	All	All

Table 18 - Task Allocation

Chapter 11: Conclusion

The goal of this project was to design, construct, and demonstrate a handheld game system that has hardware tailored specifically to the rules and feel of the game. The design of the game is such that the user will use a toggle to quickly move a cursor through colored arrows of a circle in the semi random sequence generated by the game. The game will feature the options to play with both toggles at the same time or each individually. A leaderboard will be kept for each toggle option.

All of the components were thoroughly tested throughout the course of this class and confirmed to be functional and integrable overall. Due to the tight nature of a handheld device and the concern for user experience however, the components also had to be tested thoroughly within their use case to ensure that they fit our needs. The two parts that concern this the most are the speakers and the haptic feedback. The haptic feedback is included in the design so that navigating and playing the game has a touchy and responsive feel to it, and thus has the highest standards among all our physical components. In testing and researching the speakers, we found that in order to play higher quality sounds we may need to include an SD card and reader in our design; this is something we will have to test further to decide how high of quality sounds we need to be satisfied with our user experience.

While the software was not overly complex relative to what is possible, it did need to be finely tuned for a satisfying game. The logic of the software had to be able to keep up with the intended speed of the game and user actions. It was preferable that the menus themselves are also smooth to navigate and easy to quickly understand. Much of the non physical user experience comes down to the graphical elements and sound design.

One of the goals of our project was to design a custom PCB. To this end we decided to design one that uses an ATmega328P to run the game and coordinate all of the inputs and outputs. The board itself houses the microcontroller, and the toggles. The board also handles all of the power distribution. The speakers, haptic feedback, and battery are wired to the I/O pins of the ATmega328P on the main PCB, but are not mounted onto it. A second PCB that is not of our design is also included in the system, the display driver; the display driver is connected to the ATmega328P via the I/O pins.

In conclusion, this project required us to research and carefully consider every component in the design and how they would affect the user's experience as well as fit into our vision. We had to determine what could present the user with a visually and tactilely enjoyable experience, to this end we chose a display with a high enough refresh rate and included haptic feedback in the design. After finalizing the decision on all of these components, our next step was in Senior Design 2. This was to integrate the system together step by step. Along with integrating the components we had to expand and fine tune the software. Once we were confident in the components and their integration with the software, we prototyped the PCB and completed the final product.

Appendix A - Sources and References

[1] "3 Options for Playing Audio on Arduino." YouTube, YouTube, 22 May 2022, www.youtube.com/watch?v=UN9XPWHamHw. Accessed 5 Dec. 2023.

[2] "4.3" 40-Pin TFT Display - 480X272 with Touchscreen." Adafruit Industries Blog RSS, www.adafruit.com/product/1591. Accessed 5 Dec. 2023.

[3] "5pcs Micro SD Carte TF Adater Reader Module." 5pcs Micro SD Carte TF Adater Reader Module, Shenzhen HiLetgo Technology Co., Ltd, www.hiletgo.com/ProductDetail/2158021.html. Accessed 5 Dec. 2023.

[4] "7.0" 40-Pin TFT Display - 800X480 with Touchscreen." Adafruit Industries Blog RSS, www.adafruit.com/product/2354. Accessed 5 Dec. 2023.

[5] "AB-004: Understanding ERM Vibration Motor Characteristics." Precision Microdrives, 28 June 2022, www.precisionmicrodrives.com/ab-004.

[6] "Amazon.Com: Rayhome Led Sign for Car, 7''x3''flexible Led Matrix Panel ..." Amazon, www.amazon.com/Rayhome-6-7x2-76Flexible-Bluetooth-Application-Programmable/dp/B0BHHW9JCC. Accessed 5 Dec. 2023.

[7] "ATMEGA328P - Microchip Technology." MICROCHIP, Microchip, www.microchip.com/en-us/product/ATMEGA328P. Accessed 23 Oct. 2023.

[8] "Bard vs. CHATGPT: What's the difference? [2024]." Zapier. https://zapier.com/blog/chatgpt-vs-bard/

[9] "Battery Power Online: The World of Alkaline Batteries." Battery Power Online |, 2 Oct. 2013, www.batterypoweronline.com/markets/batteries/the-world-of-alkaline-batteries/.

[10] "Buy A Raspberry Pi Zero W." Raspberry Pi, Raspberry Pi, www.raspberrypi.com/products/raspberry-pi-zero-w/. Accessed 20 Oct. 2023.

[11] "Chatgpt vs. Jasper Ai: Which AI Writer Is Best for Your Business?" RSS, seo.ai/blog/chatgpt-vs-jasper-ai#:~:text=ChatGPT%20offers%20businesses%20a%20tool,consistent%20and%20targeted%20promotional%20materials. Accessed 5 Dec. 2023.

[12] "Character LCD 2X16 Blue Display Module HD44780 White on Blue." BuyDisplay, www.buydisplay.com/character-lcd-2x16-blue-display-module-hd44780-white-on-blue. Accessed 5 Dec. 2023.

[13] "COM-09032 Sparkfun Electronics | switches | DigiKey." (n.d.). www.digikey.com/en/products/detail/sparkfun-electronics/COM-09032/6823623?utm_ad

group=Switches&utm_source=google&utm_medium=cpc&utm_campaign=Dynamic%20Search_EN_RLSA_Buyers&utm_term=&utm_content=Switches&utm_id=go_cmp-175054755_adg-15264281715_ad-665604605996_aud-505192123430:dsa-45015282538_dev-c_ext-_prd-_sig-CjwKCAiAjrarBhAWEiwA2qWdCNUra8lnokUiNzSfV9e6IcxkstQKOJFDA1qfrGJxzVFYqWc18y3iYRoCQ0wQAvD_BwE&gad_source=1&gclid=CjwKCAiAjrarBhAWEiwA2qWdCNUra8lnokUiNzSfV9e6IcxkstQKOJFDA1qfrGJxzVFYqWc18y3iYRoCQ0wQAvD_BwE

[14] "Digikey, www.digikey.com/en/products/detail/universal-solder-electronics-ltd/LCD-1602-2x16-Blue-White/16821383." Accessed 5 Dec. 2023.

[15] "ESP32-WROOM-32-N4 Espressif Systems | RF and Wireless | DigiKey." DigiKey www.digikey.com/en/products/detail/espressif-systems/ESP32-WROOM-32-N4/8544298. Accessed 3 Nov. 2023.

[16] "Getting Started with the ATMega328P." Instructables, Instructables, 29 Sept. 2017, www.instructables.com/Getting-Started-With-the-ATMega328P/.

[17] "Generic Standard on Printed Board Design - IPC." (n.d.-a). www.ipc.org/TOC/IPC-2221B.pdf

[18] "Home." Arduino, www.arduino.cc/. Accessed 5 Oct. 2023.

[19] "How Does a Linear Resonant Actuator Work?" Analog - Technical Articles - TI E2E Support Forums, e2e.ti.com/blogs_/b/analogwire/posts/how-does-a-linear-resonant-actuator-work#:~:text=A%20linear%20resonant%20actuator%20(LRA,mass%20connected%20to%20a%20spring. Accessed 5 Dec. 2023.

[20] "How lithium-ion batteries work. Energy.gov." (n.d.). www.energy.gov/energysaver/articles/how-lithium-ion-batteries-work#:~:text=The%20movement%20of%20the%20lithium,to%20the%20negative%20current%20collector.

[21] "IEC." IEC 61508:2010 CMV | IEC Webstore, webstore.iec.ch/publication/22273. Accessed 20 Oct. 2023.

[22] "IEC. IEC 61508-1:2010 | IEC Webstore | functional safety, smart city." (n.d.). webstore.iec.ch/publication/5515

[23] "IEC. IEC 60086-1:2021/COR1:2022 | IEC Webstore. (n.d.)." webstore.iec.ch/publication/76127

[24] "IEEE SA - IEEE standard for floating-point arithmetic. IEEE Standards Association." (n.d.). standards.ieee.org/ieee/754/6210/

[25] "IEEE SA - IEEE Standard for Advanced Audio and Video coding. IEEE Standards Association." (n.d.-a). standards.ieee.org/ieee/1857/10644/

[26] "International Electrtechnical Commission Homepage." IEC, International Electrtechnical Commission, Accessed 1 Nov. 2023, iec.ch/.

[27] "JayconSystems, and Instructables. “Getting Started with the ATMega328P.” Instructables, Instructables, 29 Sept. 2017, www.instructables.com/Getting-Started-With-the-ATMega328P/.

[28] "Joystick Information and Circuits, www.epanorama.net/documents/joystick/ataristick.html." Accessed 15 Oct. 2023.

[29] "LCD 1602 2x16 Blue-White Universal-Solder Electronics Ltd ..." Digikey, www.digikey.com/en/products/detail/universal-solder-electronics-ltd/LCD-1602-2x16-Blue-White/16821383. Accessed 5 Dec. 2023.

[30] "Lead Soldering Safety, blink.ucsd.edu/safety/occupational/hazard-control/lead-soldering.html#:~:text=Workers%20can%20be%20exposed%20to,and%20muscle%20and%20joint%20pain. Accessed 29 Oct. 2023.

[31] "Matan. (2023, October 26). Toggle switches: How it works, application & advantages. Electricity." www.electricity-magnetism.org/toggle-switches/

[32] "Meganburroughs. “AB-004: Understanding ERM Vibration Motor Characteristics.” Precision Microdrives, 28 June 2022, www.precisionmicrodrives.com/ab-004."

[33] "MSP430FR69891." MSP430FR69891 Data Sheet, Product Information and Support | TI.Com, Texas Instruments, www.ti.com/product/MSP430FR69891. Accessed 3 Nov. 2023.

[34] "Mini 2-axis analog thumbstick." adafruit industries blog RSS. www.adafruit.com/product/2765

[35] "Mini Oval Speaker with Short Wires - 8 Ohm 1 Watt." Adafruit Industries Blog RSS, www.adafruit.com/product/4227. Accessed 5 Dec. 2023.

[36] “Nichel-Metal hydride B - ease storage." (n.d.-b). ease-storage.eu/wp-content/uploads/2016/03/EASE_TD_NiMH.pdf

[37] "Nickel-metal hydride B - ease storage." (n.d.-b). ease-storage.eu/wp-content/uploads/2016/03/EASE_TD_NiMH.pdf

[38] "Piezo actuators: Types and applications. Types And Applications - Piezo Actuators | APC International." (n.d.). www.americanpiezo.com/piezo-theory/actuators.html

[39] "RA8875 Touch Display Driver Board." Adafruit Learning System, learn.adafruit.com/ra8875-touch-display-driver-board/arduino. Accessed 5 Dec. 2023.

[40] "Rayhome Led Sign for Car, 7''x3''flexible Led Matrix Panel ..." Amazon, www.amazon.com/Rayhome-6-7x2-76Flexible-Bluetooth-Application-Programmable/dp/B0BHHW9JCC. Accessed 5 Dec. 2023.

[41] "SP-2804L - SP Series." Soberton Inc., 12 Feb. 2022, www.soberton.com/sp-2804l/.

[42] "SP-3605 - Speakers." Soberton Inc., 12 Feb. 2022, www.soberton.com/sp-3605/.

[43] "SP-16273 Sparkfun Electronics | switches | DigiKey." (n.d.-a). www.digikey.com/en/products/detail/sparkfun-electronics/COM-16273/13237023 [51]

[44] "Talking Arduino | Playing MP3 Audio with Arduino | Arduino PCM Audio without Audio or Mp3 Module." YouTube, YouTube, 3 May 2018, www.youtube.com/watch?v=F28Znry0qcw&t=127s. Accessed 5 Dec. 2023.

[45] "Things That Go Bzzzzzz (Haptic Feedback Motors)." YouTube, YouTube, 17 Oct. 2021, www.youtube.com/watch?v=mgwWh6gVGCo. Accessed 5 Dec. 2023.

[46] "Toggle switches: How it works, application & advantages. Electricity." www.electricity-magnetism.org/toggle-switches/.

[47] "Toy Industry Association, Inc. "U.S. Sales Data." The Toy Association, www.toyassociation.org/ta/research/data/u-s-sales-data/toys/research-and-data/data/us-sales-data.aspx. Accessed 28 Sept. 2023.

[48] "Utmel. (2021, March 25). Analog Switch: Types and application. https://www.utmel.com/blog/categories/switches/analog-switch-types-and-application"

[49] "USB." Encyclopædia Britannica, Encyclopædia Britannica, inc., 13 Sept. 2023, www.britannica.com/technology/USB.

[50] "What Is Chatgpt? Everything You Need to Know." WhatIs.Com, TechTarget, 17 Nov. 2023, www.techtarget.com/whatis/definition/ChatGPT#:~:text=ChatGPT%20is%20an%20artificial%20intelligence,%2C%20essays%2C%20code%20and%20emails.

[51] "What Is the History of Artificial Intelligence (AI)?" Tableau, www.tableau.com/data-insights/ai/history#:~:text=Birth%20of%20AI%3A%201950%2D1956&text=into%20popular%20use.-,Dates%20of%20note%3A,ever%20learn%20the%20game%20independently. Accessed 5 Dec. 2023.