Smart Chess Board

Senior Design II

Spring 2020

Group 10

Diego Garcia: Electrical Engineer

Noel Membribe: Electrical Engineer

My Ly Phan: Electrical Engineer

Damani Sinclair: Electrical Engineer

Table of Contents

List of Figures

List of Tables

1. Executive Summary

Chess can be a very fun and mind-expanding game, created centuries ago. Though it's beneficial to the mind and played by millions of people around the world, chess isn't enjoyable to everyone. To some, this classic game can be boring, frustrating and just too time consuming. This was where the motivation of the Smart Chess Board came to mind. To mix it up a bit, the idea of Smart Chess Board was made to be hands-free and voice activated so that will give the game of chess a twist of livelihood, entertainment, and for those whom may like a more innovative recreation of the game. To accomplish this, the chess board would be voice-activated and functioned using a voice recognition software, movements made with the aid of an electromagnet, and some added extra features being LEDs to make it more fun and aesthetically appealing.

The physical appearance of the Smart Chess Board looks like a typical chess board, but bigger and boxier to accommodate the hardware, electrical, and software components that aided in making this project successful. With the base and perimeter surrounding the rest of the board of wooden material and the top surface of the board of Plexiglass, the main components within and outside the chess board consists of the electromagnet, magnets, the piece movement system, the LCD screen, LEDs, power source, microcontroller, PCB, and much more to aid in the functioning of the board. Each game piece has a magnet implanted inside of them for when given the go, the piece movement system acts upon command and relocate the chess piece. The board also includes a "graveyard" located on the surface of the board, off to the side, for when a piece is to be killed off or when a pawn is to be promoted.

The Smart Chess Board is like any other chess game where it's two people challenging one another. The board is to be plugged in for power, and with the use of a laptop to run the voice recognition software and the microphone, the game can begin. The first player can vocally state which piece and to what location to move to by stating the grid's location of A through H and 1 through 8. This is done by using the voice-activating software. While this is happening, LED strips within the box, along the edges of the chess board is illuminated. The piece movement system and electromagnet controller will then come into play and move the piece to the desired location. Movement of the chess pieces is from the electromagnet controller magnet attracting and attaching itself to the magnet that's embedded inside the chess pieces, the pieces move along the lines of the chess board to get to its assigned destination. The players are required to alternate turns and state what their moves is. The player must state whether a piece is to be moved or captured and that will determine if a piece is going to be taken to the graveyard. This iteration will continue until the King is annihilated and the game is over.

2. Project Description

In this day and age, the word “smart” is added to almost anything and everything and it’s capable of being something extraordinarily amazing or ridiculous, or something that’s so ridiculous, it’s extraordinarily amazing. Chess is one of the oldest games in the world, tracing back to about 1500 years ago, to the 6th century. Due to the fact that the team is a group of engineering students that enjoy the game from time to time, the thought of a “Smart Chess Board” came to mind.

2.1 Motivation and Goals

Chess can be very beneficial to exercising the mind. It's a thirty-two-piece puzzle that can stimulate the brain, with the first move having twenty possible moves and after three moves, millions of possible variations. The reason behind the Smart Chess Board was to allow users to not be required to physically move chess pieces while in play, possibly make the game more interactive and fun, and considering it’s such a classic, more popular. Since the chess set was to be voice-activated, anyone who can give voice commands could play.

There were no sponsors for this project. As a group, there was, and always will be, a want for something challenging, fun and entertaining altogether. With a team of all Electrical Engineering Students, the Smart Chess Board was a great opportunity to learn and really put our skills and experience to the test. The project wasn’t expected to be too complex or programming intensive but more mechanically and electrically concentrated, and boy, were we wrong!

2.2 Objectives

The project is very much like any basic chess board but bigger and under the surface of the board, there are electrical and mechanical components that will aid in moving the chess pieces to specific locations specified by the player, making it boxier than the average board game. The use of a laptop was used to run the voice recognition software, as well as for the microphone so the player can speak commands for relocating chess pieces. The base and surroundings of the board was made of plywood, whereas the top surface was partially wood, working as a border, and plexiglass for the playing field. Within the chess board, the electromagnet was attached to the X-Y plotter, that’s programmed to slide the specified chess pieces to the locations stated. The LCD was originally included as another feature in the design to visually see the correct movements and placement of the chess pieces because there would be a chance that the relocation of the pieces won’t be as precise as predicted, as well as commands and instructions by the players and the voice software. After testing and involvement of the LCD with the entirety of the project, more complications arose. Therefore, the LCD just displays simple greetings and remarks. The primary power source of the chess board was a typical power outlet for ease of use. The core features of this project are stated below in Table 1: Core Features.

Table 1: Core Features.

Core Features	
Movable magnetics	Relocate chess pieces.
Voice control	Communicate with the board.
LEDs	Livelier board.
LCD	Displays comments.

A few ideas that were under consideration that were more advanced, also stated in Table 2: Advanced Features, were solar panels, LED lights, and a magnetic generator. Including solar panels at the top edges of the chess board and possibly a power bank could help with portability and serve as an extra power source so that the game can be played anywhere. If this were to be done, size and weight will really come into effect. Another idea to generate power to the board was to use magnets and a DC motor.

Table 2: Advanced Features.

Advanced Features	
Solar Panel/Power Bank	Extra power source.
LEDs	Lights up board to teach players how to play.
Magnetic Generator	Uses Magnets and dc motor to generate power.

One of the ideas that could've been incorporated into the chess board were LED lights within all of the chess pieces, and to advance this portion of the project, the concept of having every square on the board being able to light up was also an option. Lighting up the chess piece would help the player visibly see which piece was moving and where to, and the board lighting up would help with the possibility of lighting up possible locations of where the piece selected can go. This would benefit in helping beginner players on how to play and learn the different moves and patterns every piece can make. This would make the chess board not only livelier and more fun to play but be a great learning experience for beginners or anybody that's interested in playing.

As more brainstorming was being done, further suggestions of what could have been included were proposed. Stated in Table 3: Stretch Goals, there were ideas of having the option to play in a one player mode, where the player can play against a computer. There were further discussions on the possibility of developing a phone application to play with someone through an app or even creating an interactive website to play with anyone through a network. These features would make the game more enjoyable for people that do not have a companion to play with, but because the team were all Electrical Engineering students and none of Computer Engineering, with limited knowledge and restricted time and money, as well, these goals were more farfetched and challenging.

Table 3: Stretch Goals.

Stretch Goals	
One Player Mode	Play against a computer.
Phone App	Making an app to play against someone.
Interactive Website	Making a website to play against someone.

2.3 House of Quality Analysis

When beginning a project of any kind, it was crucial to monitor every aspect of the advantages and disadvantages and how it can affect the different components within the design. The House of Quality was part of a product development process known as quality functional deployment (QFD) that is widely used in industry. QFD is a series of processes that incorporate the customer’s needs throughout the system life cycle. QFD encompasses design, manufacturing, sales, and marketing and is characterized by a series of matrices that visually look like a house. These matrices relate the engineering requirements with the marketing requirements and were used to communicate between different units in an organization. This helped in keeping track of important tradeoffs in constructing the project. This house of quality analysis was created early on during the research process that was performed in this project. It was specifically focused on the requirements specification and is shown below in Figure 1.

Marketing Requirements		Dimensions (-)	Processing Time (-)	Weight (-)	Power Consumption (-)	Production Cost (-)
Durable	(+)	↓				
Easy to Use	(+)	↑	↑	↑		
Portable	(+)	↑		↑	↑	
Easy to Maintain	(+)	↑		↑		
Reliable	(+)				↑	
Cost	(-)	↑	↓	↓		↑↑
		< 62x62x20 cm	< 5 seconds	< 30 pounds	< 1000 Watts	< $800

Strong positive correlation	↑↑
Positive correlation	↑
Negative Correlation	↓
Strong negative correlation	↓↓
Positive polarity	(+)
Negative polarity	(-)

Figure 1: House of Quality.

2.4 Project Management

To successfully complete this project in a timely manner, careful planning and reviewing was required. The project was divided into sections called project blocks and every member was assigned to certain topics determined by each member's strength and interest in the matter. With approvals and discussions within the team, the distribution of the project was as follows in Table 4: Project Management. Responsibilities changed very often throughout the project due to availability of parts, workload and difficulty. One of the main portions of the project was software and programming and none of the group members are extensively concentrated in that, so this portion of the project had tasks that was demanding, and it was necessary for all members to explore and learn more about it. The following table lists which member was administratively responsible for ensuring that each project block was completed in a timely manner. This does not mean that a group member wouldn't be able to work on a project block which was not their responsibility; working together to integrate all of the different parts was essential to ensuring the whole project functions correctly.

Table 4: Project Management.

Group Member	Project Block
Damani Sinclair	Motor Control
Damani Sinclair	PCB Design
Damani Sinclair & Noel Membribe	Power System
Damani Sinclair & My Ly Phan	Magnets
Diego Garcia	Chess Board Housing
Diego Garcia	Chess Piece Set
My Ly Phan	LEDs/LCD
My Ly Phan & Damani Sinclair	Microcontroller
Noel Membribe & My Ly Phan	Voice Recognition Software
Noel Membribe & My Ly Phan	Piece Movement System
Noel Membribe	Piece Detection

3. Requirements Specifications

The requirements specification is a collection of engineering and marketing requirements that a system must satisfy in order for it to meet the needs of the customer or end user. According to the Senior Design textbook, there are three stakeholder groups in the process for developing a requirements specification: the customer, the technical community, and the environment. The input from the customer includes the marketing requirements. The input from the technical community is based upon the knowledge of engineers who are primarily responsible for design, implementation, testing, manufacturing, and maintenance of the system. These two sets of requirements make up the house of quality, shown in Figure 1, that was developed for this project. The environment introduces requirements in the form of constraints and standards that impact or limit the design. These are elaborated on the in the following sections.

3.1 Standards

Engineering standards can be defined as a standard or established way of doing things that ensure interoperability. Standards ensure that products work together and ensure the health and safety of products that people use every day. Identifying and following standards and designing the project around them is an expected part of good engineering practice.

3.1.1 Chess Standards

While the basic geometry of a chess board is fairly easy to grasp, there is a surprising amount of complexity involved in the chess board and chess piece standards. Every chessboard is made up of an 8x8 grid, resulting in 64 total squares. The squares are colored in an alternating pattern, so that there are 32 dark squares and 32 light squares. All of the squares are the same size, and each side of the chess board is the same length, forming a square when viewed from above. There are 32 total chess pieces, 16 for each player. At the start of a game, the chess pieces are set up with 16 on each side, leaving the middle 32 squares open for play.

The horizontal rows (from the perspective of the players) are called ranks and are numbered from 1 to 8. The vertical rows are called files and are named from A to H. Each of the 64 squares can be named by referring to their rank and file number, e.g. A1 or D6. White pieces should always be set up on the 1st and 2nd ranks, while black pieces should be set up on the 7th and 8th ranks. The 2nd and 7th ranks contain only pawns. The 1st and 8th ranks contain the other pieces in the following orientation: Rooks on files A and H, Knights on files B and G, Bishops on files C and F, Queens on file D, and Kings on file E. Every game of chess must begin with the pieces set up in this way.

The specific sizes of the chess board and chess pieces can vary by country or by which chess standards body is referenced. According to the United States Chess Federation (USCF), the square size should be anywhere from 2 to 2.5 inches, while the king's height should be 3.375 inches to 4.5 inches. Other chess standards bodies have slightly different sizes; the possible combinations for boards and sets are almost limitless.

More important than the actual sizes of the chess boards' squares and the chess pieces was the ratio between them. The king is the piece with the widest diameter, and so the king's base diameter is used as the basis to determine the size of the rest of the pieces and the squares. The USCF recommends that the king's base diameter should be between 40-50% of the king's height. As a general guideline, the king's base diameter should be roughly 75-80% of the size of the square to ensure that the board and set combination has the proper piece spacing. If the ratio falls outside of this range, then the board could either be overcrowded, causing pieces to be accidentally knocked over easily, or it could have too much space between pieces, making it harder to tell a glance which pieces are which.

For the purposes of this project, it was not possible to follow these chess standards exactly because if the ratio of 75-80% was used, then the pieces would bump into each other while moving. This was because the pieces will be moving across as 2D plane, as opposed to being lifted off of the board by a human player. Because of this, the largest piece (the king) cannot have a base diameter larger than 50% of the size of the squares.

3.1.2 Communication Standards

There are several different types of connections that communication between the different hardware and software components of the Smart Chess Board. Depending on the nature of the data being transmitted, as well as the nature of the hardware or software that needs to communicate with each other, different communications standards are used. They are outlined below.

3.1.2.1 USART

The ATmega2560 microcontroller was connected to a team member’s PC in order to communicate with the voice recognition software, BitVoicer. They are connected via serial communication using USART. Since the PC doesn’t have a serial input, a USB-to-Serial converter was used to connect the devices. USART stands for Universal Synchronous/Asynchronous Receiver/Transmitter and was a microchip that facilities communication through a computer’s serial port using the RS-232C protocol. The ATmega2560 microcontroller chip has 4 USART ports, labelled Tx0, Rx0, Tx1, Rx1, Tx2, Rx2, Tx3, Rx3. BitVoicer is to be communicated with the Atmega2560 using the Tx0 and Rx0 pins, which corresponds to one of the USART ports.

USART provides the computer with the interface necessary for communications with modems and other serial devices. Unlike UART, however, USART allows for the possibility of a synchronous mode, hence the name. What synchronous mode means in program-to-program communication was that each end of an exchange responds in turn without initiation a new communication. Asynchronous operation means that a process can operate independently of other processes. Thus, USART can provide all of the capabilities of UART with the extra possibility of synchronous operation.

3.1.2.2 USB Hardware

USB, or Universal Serial Bus, is an industry standard that establishes specifications for cables and connectors and protocols for connection, communication, and power supply between computers, peripheral devices, and other computers. The type of USB port used for this project was the USB3.0 A-type; the team member's PC was a Lenovo Yoga 710, which contained two of these types of USB ports.

The XY-plotter was connected to the team's member's PC via a USB3.0-A to Micro-B cable. Micro-B is a type of Micro plug, which are the most common connections used for smartphone charging and are also commonly used in embedded devices, such the boards used in this project. The PC was running the software which controls the stepper motor drivers. The ATmega2560 microcontroller was performing the analysis of the chess commands and converting them into usable code that the XY-plotter can use to move the pieces. The ATmega2560 also runs the algorithm which chooses the ideal pathing for the XY-plotter based on the known piece positions and other variables.

Because of this, the XY-plotter was not connected directly to the ATmega2560 microcontroller; they are connected indirectly with the PC acting as an intermediate device. The microcontroller was connected to the PC using the same type of cable as the XY-plotter uses to connect: USB3.0-A to Micro-B. The XY-Plotter uses a Makeblock Orion microcontroller (Arduino UNO compatible) which uses an ATmeg328p control chip.

3.1.2.3 Design Impact of Relevant Standards

The chess standards listed above are uniform throughout the entire world of chess. These standards were implemented and used in the design of this hands-free chess board in all aspects that allow those standards to work with the functions of the chess board. That being stated, not all standards stated above are possible within the design limits of the products. These standards were met to the highest possible level while allowing for some changes to fit with the design of this product.

One of the standards that must be adjusted to fit the needs of the set design was that of the chess piece height and diameter ratio. With the chess boards playing surface being a much smaller area than the entirety of the chess boards outer skeleton, due to constraints in motion set by the XY-plotter, the diameter of the largest piece (the king) would already be less than most playing pieces in chess board sets offered in stores. In order to keep the chess pieces stable during movement, the diameter of each chess piece must be set to the maximum allowed value per the design. This was determined by the size of each playing square. Since the chess pieces must maneuver between each other on the lines of the grid, the max diameter was half that of each square. To compromise with the chess standards, though the chess pieces would all share the same base diameter. The height of each piece would be based on the height of standard chess pieces.

3.2 Constraints

Some of the constraints of this project are size, removal of the pieces, disturbance with relocating the pieces, and LED visibility. The board needs to be big enough to accommodate thirty-two chess pieces and all the pieces needs the option to be magnetically operated. The idea to move one chess piece to another location was to slide along the lines of the squares of the chess board. With that, the board must be big enough or the chess pieces small enough so that the other pieces won't be interrupted during play.

Another sizing issue to be investigated was the Plexiglass. It can't be too thick for the magnet on the pieces to be able to be attached to the moving mechanism under the board. Another challenging constraint we had was trying to show the LED usage without there being a disturbance to the magnets or the track they must move on. The magnet and LED must also be small enough to fit into the chess piece together. An additional constraint that was known of was looking for the right materials for the chess pieces to have the ability to slide easily across the board.

The maximum power that could be used was 1440W. This was calculated based on a 120V, 15A outlet under a continuous load. To be safe, we determined that we should use the least power possible to avoid tripping the breaker.

3.2.1 Realistic Design Constraints

Every engineering project needs to consider a set of realistic design constraints, which guide the decisions that are made throughout the engineering design process. The engineers working on a project need to choose a design which balances cost, functionality, and safety. Outlining the realistic design constraints allows an engineer to design a product which fits under budget while functioning adequately and meeting all health and safety standards.

3.2.2 Economic and Time Constraints

The budget for this project was estimated to be $800.00 for all of the parts required to construct a working prototype. The project was funded by the team members directly so minimizing costs was an important design consideration for all aspects of the project. It would be considered a success to maintain costs under $800.00, but that can be sacrificed if needed in order to maintain adequate functionality. The most expensive subsystem involved in this project was the XY-Plotter, which was estimated to cost about half of the overall budget, or $400.00, based on other similar projects and price estimates of different parts online. The plywood and plexiglass case and electronics hardware make up most of the rest of the budget. Whenever possible, free resources such as the PocketSphinx voice recognition software were used.

Time constraints represent a significant factor in informing design decisions for this project. The full project timeline from when a project was decided on to when the final documentation, prototype, and presentation were due was approximately six to seven months. Since the first three months need to be dedicated to writing the first version of the final documentation, there are only about three months left to construct and test a working prototype. For a prototype of the proposed project to be adequately functional, several different large subsystems need to be working properly. If one subsystem fails, the entire project won't work as desired and would likely lose many points on an evaluation.

For these reasons, actions were taken whenever possible to minimize the amount of time spent to construct a working product. For example, it was a strong consideration to purchase an XY-plotter kit rather than to build one from scratch since it was known from researching previous similar projects that building one from scratch was a big-time sink. Purchasing a set of chess pieces and adapting the magnet mechanisms around them was considered because it took much less time than making our own custom chess pieces adapted around an existing magnetic system through 3D printing or some other method. A free voice recognition software was downloaded and adapted to the project's specific needs rather than creating one from scratch. Much care was taken to read through all of the research that was done by other senior design groups on similar projects in order to learn from their mistakes and avoid repeating any similar ones which would waste time.

3.2.3 Environmental, Social, and Political Constraints

The environmental constraints involve designing a product such that the resources consumed in making and maintaining it are taken into account and can be justified by the intended benefit of the product. The product designed in this project uses a large amount of resources for a relatively simple concept; a normal chessboard is much less costly to build than a voice-activated one. However, this product was meant to target a niche demographic and was not meant to be mass produced. Thus, the environmental impacts are minimized in the sense that only one prototype would likely be constructed.

Recycled materials would be used for the plywood and plexiglass outer case of the chess board. The most environmentally harmful components used are all the electronics, which would be difficult or impossible to reuse once the prototype serves its purpose, and the thirty-two permanent magnets which in the long term need to be disposed of somewhere.

3.2.4 Ethical, Health, and Safety Constraints

The safety constraints for this project are important in ensuring that the product was completely safe for every type of user, including children and people with disabilities. The entirety of the moving parts of the product need to be enclosed in the plywood and plexiglass glass so that the user was not exposed to the moving parts of the XY-plotter mechanisms, which can be a minor pinch hazard. The small magnets that are attached to the chess pieces must be strongly attached so that they do not separate and become a choking hazard for children. The chess pieces themselves must be large enough that they do not pose a choking hazard.

An additional safety constraint was the possibility of a fire hazard caused by the sizeable amount of electronics and other flammable materials used in the project. This can be mitigated by minimizing power consumption to reduce overheating, checking for loose connections, properly soldering components together, and regulating voltages to ensure there are no sparks. A lot of pre-planning and testing was required when building the prototype to ensure that the risk of fire was as low as possible. Nonetheless, this remains an issue which could limit the marketability of this type of product.

3.2.5 Manufacturability and Sustainability Constraints

The manufacturability constraints limit the types of components that can be used to ones that can be manufactured using the given resources. Often, manufacturing a component or sub-system rather than buying it increases customizability and ease of integration with the rest of the project at the cost of money and time. It was being considered whether to 3D print custom chess piece that can be made to the exact specifications desired or whether to buy a chess piece set that fits a loose set of requirements and design around it. In this case, the added costs of manufacturing the pieces instead of buying them may not be worth it. In the case of the XY-plotter, it was much less costly to buy a kit and modify that if needed than to design and build one from scratch. The outer case consisting of the plexiglass chessboard and plywood base can be manufactured by one of the project's team members in their garage. This obviously saves a lot of time and money and was a big reason in choosing this type of design in the first place.

As far as sustainability constraints, the product needs to be able to be reused for a reasonable period of time and maintain its functionality with as little maintenance as possible. The outer case needs to be stained and finished so that it can keep any moisture out and last as long as possible. If possible, the case should be able to keep out any dust or particles which may accumulate and impair the motors or electronics. The motors need to be able to operate for at least a few hundred hours of playtime or else be easy to replace. The outer case would be designed in such a way that it can be easily opened for maintenance. If this product was marketed to be sold, an instruction booklet was included which gives the user directions on how to properly maintain the product to extend its life as long as possible.

4. Research

A tremendous amount of time went into researching, learning, and analyzing for this project. This was especially important for a team of all Electrical Engineering students working on a design heavier on the programming and mechanics aspect compared to an electronics stance. Although Smart Chess Board was a more challenging and demanding choice for a full team that focuses purely on electronics rather than a mixed team with computer or mechanical background, this provided a great learning experience for all of the team members.

4.1 Similar Projects and Products

In this section, an outline of similar senior design projects or products of chess boards that already exist was researched and documented to help aid in designing Smart Chess Board. Having access to the information from past projects was a great advantage. Some of the reasons being that most of the basic research has already been done and was easily attainable, not much time was wasted by repeating the mistakes made from the other projects, and the different features provided from each project was able to be compared along with their results, which allowed the team to learn and even try different approaches to further progress the project.

4.1.1 Magic Chess

The Magic Chess project was one of the senior design projects done by previous students. This project was very similar to the product being aimed to be built, so it'll be used as one of the main references to help guide in designing and creating the Smart Chess Board. The students did extensive research which saved our group a lot of time and money. They used many similar features in their project that we decided on, but most importantly, the primary elements and main motivation for the design, the movement of the magnetically controlled pieces and the use of voice recognition.

Magic chess was a voice-activated, hands free chess board that has player vs player and computer vs player capacity. The main features of this board were having the ability to communicate via voice commands, the magnetic plotter that moves the pieces, and a chess engine that would allow you to have different playing options. The chess engine used was one of the main features that was a different approach. The board included an LCD screen and upon startup, it would display a few simple options, such as, the number of players and the difficulty level of the computer. When playing the computer, it uses decision-making algorithms such as Minimax and Iterative Deepening for the computer to determine the best moves. The computer would even check for end game conditions after every move, such as, check, checkmate, and stalemate. This project was one of the main references used because it was the most relatable to our project.

4.1.2 Telepresence Chessboard

Another similar project that really assisted in the design of Smart Chess Board was Telepresence Chessboard. Telepresence Chessboard was created so the users could play with online players, while using a physical chess board. There's a communication through the network and commands would be received and the chess pieces would move to the assigned location on the physical board. This project had magnets implanted in the chess pieces and under the board for movement and Radio Frequency Identification tags to distinguish the different chess pieces. Much of the features included in this project was consistent with what was to be designed; piece detections, piece movement, not technically voice, but command in general, and considerably more. The research done in Telepresence was promising and valuable.

4.1.3 Knight Light LED and Deep RGB

Another similar project that assisted and was used as a reference was Knight Light LED. Knight Light LED was created more as a tool to learn the game of chess rather than entertainment, but the components within the chess board and its capabilities was very much like the ones aimed to design Smart Chess Board. This project supports single-player mode and two-player mode, allowing for player-versus-player and player versus computer.

Lastly, another senior design project that aided in the design of Smart Chess Board was Deep RGB. Like the other projects, Deep RGB allows for the option to play between player versus player, computer or anyone through the internet. This project was made for people that enjoys the game and are also on the ropes of learning. Similarly, to Knight Light LED, possible moves or locations of where a piece can move to are illuminated on the board. Chess pieces are manipulated and moved with magnets once the command was received. This project would also help in Smart Chess Board.

Knight Light LED and Deep RGB had some really useful information that was beneficial in creating and designing Smart Chess Board. Some of the research included the use of software sensing and tracking pieces, LEDs, LCD screen, and overall researched hardware materials, such as, the PCB selection, power sources, and microcontroller.

4.2 Piece Movement System

For this voice-activated chess board design, the players would verbally make commands by communicating with the voice software. The player was to state the desired piece to activate, specifically, where the piece was located on the chess grid, and then declare where the destination was to be. After the speech was recognized by the software, the selected chess piece would then be moved, without any physical contact, around the board to the operator's desired location. In order for this to be possible, a mechanism had to be implemented into the design to attach and move each chess piece without coming into contact or disturbing the other stationary pieces on the playing surface, as well as, the internal components of the board. Given the fact that the chess board's field of play was a flat surface with both an X and Y plane, two options for mechanisms, each with their own advantages and disadvantages, were feasible for the implementation of this design.

4.2.1 Robotic Arm

A robotic arm was one of the options considered for moving the chess pieces from one location to another. It would work similarly to a human arm with the exception of it being a mechanical arm that consists of many electrical and mechanical components to function. Depending on how much time and money would be required to be put into the construction of the arm, there was a great chance that it could've been very costly and time consuming. The robotic arm would do what it's programmed to do but if it were cheaply made or bought, there would be a lack of precision and accuracy, along with speed and efficiency. Though it would've been easily implementable, their bulky design would make for hard transportation and lack of portability. Another downfall of the robotic arm would be its constant need to be monitored at all times just in case of mechanical failure, resulting in disruption or stall in the game.

4.2.2 XY-Plotter

An alternative option to the robotic arm was an XY-Plotter. Some of the advantages of an XY-plotter was that it has free range of motion in both the X and Y directions and can travel freely and precisely within the designated area of the plotter's specification. XY-plotters are commonly found in CNC machines and 3D printers as well as kits sold online where users build a plotter that could be programmed to draw any given design with a pen and paper. If the XY-Plotter were to be chosen as the piece movement system, there was also the options of building the plotter from scratch or just buying a kit and putting it together ourselves. Overall, it would make the chess board more aesthetically appealing due to it being hidden under the surface of the board additionally, improving0 portability compared to the robotic arm.

4.2.3 Comparison of Robotic Arm and XY-Plotter

For this project, the two methods for transporting the chess pieces across the playing surface of the board that was found to fit the desired use were to either implement a robotic arm on the outside of the board to grab and lift pieces from place to place, or implement an XY-Plotter beneath the board to attach to and slide the pieces from place to place. Both had its advantages and disadvantages and the most critical factors are displayed below in Table 5: Robotic Arm Vs. XY-Plotter.

Table 5: Robotic Arm Vs. XY-Plotter.

Piece Movement System	Advantages	Disadvantages
Robotic Arm	• Easy implementation. • Less expensive. • Variety of options for purchase.	• Bulky and protruding from the design. • Less portable.
XY-Plotter	• Hidden from view. • Smooth transition of chess pieces. • Improves portability.	• More time consuming. • Harder to implement.

4.3 Magnets

One of the main features of this design was playing hands-free, and to successfully execute that implementation, magnets play a crucial role. Magnets can be broken down into different categories and subcategories based on their material or magnetic properties; this made the decision quite difficult. With a lot of research and testing, the right set of magnets was chosen. A few of the different types of magnets that were options to choose from were permanent magnets, temporary magnets, and electromagnets. The material the magnet was made of determines its advantages and disadvantages.

4.3.1 Permanent Magnets

One of the main types of magnets are permanent magnets. These magnets are made of ferromagnet material that produces a magnetic field. Applications for these magnets include generators, electric motors, everyday electronics such as televisions and phones, and so much more. There are four main types of permanent magnets, Neodymium Iron Boron (NdFeB), Samarium Cobalt (SmCo), Alcino, and Ceramic or Ferrite. Below in Table 6, states the advantages and disadvantages of each of them.

Table 6: Permanent Magnets.

Type	Advantages	Disadvantages
Neodymium Iron Boron (NdFeB)	• Composed of rare earth magnetic material. • Highly coercive force. • Have an extremely high energy product range. • Difficulty to demagnetize. • Can usually be manufactured small and compact.	• Low mechanical strength. • Tend to be brittle. • Low corrosion-resistance if not coated.
Samarium Cobalt (SmCo)	• Very strong. • Difficult to demagnetize. • Highly oxidation-resistant and temperature resistant.	• Expensive. • Low-mechanical strength.
Alnico	• Good temperature resistance. • Can be produced to yield different magnetic characteristics.	• Can easily be demagnetized.
Ceramic or Ferrite	• Inexpensive. • Strong. • Not easy to demagnetize.	• Brittle, easy to break.

4.3.2 Temporary Magnets

Temporary magnets are essentially any material that behaves like a permanent magnet when in the presence of a magnetic field. They vary in size and composition. Soft iron metals, such as paper clips, are usually considered temporary magnets. Its property of momentary magnetization was an affective option for Smart Chess Board because it only maintains magnetism when there's an electrical current running through them. Thus, the use of temporary magnets, specifically the electromagnet, was necessary for the project to work.

4.3.2.1 Electromagnets

An electromagnet is a type of temporary magnet that is a coil of electrically conducting wire. Typically, it is created by winding the wire into multiple loops around an iron core. It was possible to create an electromagnet that was wound around an air core; this design is called a solenoid. An electromagnet can technically be created with a single coil of wire, however, using an iron core and a higher number of loops increases the strength of the magnet. An electromagnet is activated when current flows through the coiled wire, then, a magnetic field is generated that has a magnetic flux density proportional to the current. The polarity of the magnet can be determined by using the right-hand rule, wrapping the fingers of your right hand around the coil in the same direction that the current is flowing, and the thumb would be pointing in the direction of the north pole of the electromagnet.

When powered by a DC current, the magnetic field behaves similarly to that of a permanent magnet, maintaining the same strength and polarity. When powered by an AC current, the magnetic flux density fluctuates with the current and reverses polarity every half cycle just like the current signal. DC electromagnets are primarily used to pick up and hold objects, whereas AC electromagnets can be used to demagnetize objects or to hold objects. For the purposes of this project, a DC electromagnet would be much more effective in achieving the desired result of moving the chess pieces by "picking them up" and moving them across the board. The properties of a DC electromagnet allow for it to be configured as a permanent magnet that can be turned on or off at will by controlling the current flowing through the coils.

4.3.3 Magnets with On/Off Switch

Another magnet type that was discovered and considered during the research process was a magnet with a "switch." An example of a well-known product that implements this function is the Magswitch. Magswitch Technology is a super-strong magnet that can be turned on and off with the half-turn of a knob, without using electricity. Each Magswitch contains two diametrically polarized cylindrical rare-earth magnets stacked concentrically on top of each other. Rotating one magnet with respect to the other optimizes or collapses the magnetic field. Thus, two permanent magnets can be used to create one magnet that can be turned on and off at will, a property that was previously reserved only for DC electromagnets until this technology was invented. Magswitch magnets are sold at various generic stores and websites ranging from a price of $15 to hundreds of dollars, depending on size, so they're easily accessible. For the purposes of this project, such magnet would need to be custom-made for the specific application in order to maintain optimal functionality of the chessboard. Theoretically, this type of magnet could be created at home using the right type of magnets and enclosing them in some type of mechanism which allows them to rotate. The downside of this was it would be time consuming to obtain the magnet of the right size to fit into the design.

4.4 Development Board

To start the embedded development, one very important component that was made use of within the design and testing phase was the development board. It was used in conjunction with an Integrated Development Environment (IDE) and aided in linking the microcontroller or microprocessor to all the other various hardware elements within the device. There were a few options to choose from and their advantages and disadvantages will be highlighted in order to select the more optimal choice.

A development board was just an easier way to get started with the design because they are already combined with various elements that was needed. Some of these elements happen to be the processor and the memory. In addition, peripheral components such as the LCD, buttons, and motor drivers were available on these boards. They possess debugging features that really helped the team save valuable time troubleshooting other problems. A few boards are highlighted below.

4.4.1 Breadboard

A breadboard is an inexpensive platform used to build and test electronic circuits. Components such as resistors, diodes, and capacitors can be plugged in the board for testing. This is usually done without applying having to solder anything so it's not permanent and easily reusable. Breadboards are wired together in a certain manner so that electricity can flow from component to component in orderly rows and columns. Everyone from beginners to experts use these devices to experiment with circuit ideas. Although frowned upon, in some cases a breadboard can be made into a useful device.

Using a breadboard for the project was extremely beneficial in many ways. A few advantages were that the whole team has an extensive amount of experience already using breadboards from classes and it didn't require any soldering, therefore, the circuits can be redesigned quickly and easily, multiple times without the permanent connection to each other until the team obtained the desired response. Although breadboards were the best choice, even for beginners and experts, they possess a few problems. Some of the disadvantages associated with using breadboards were that they are usually rated for approximately 5W. This means that the team would have to test parts in sections in order to minimize wattage use each time. The connection strips introduced added resistance into the circuit, or the results may not be obtained due to breadboard connection error. Breadboards are unsuitable for high-frequency circuits due to the relatively large capacitances between isolation pins. Only chips with Dual in-line packages can access the center division, without this ability the board may become crowded very quickly due to space restrictions. The last disadvantage with using a breadboard tends to be the most common problem where the jumper wire can become easily loose making the debugging process difficult because every connection must be tested in the circuit to make sure that there was continuity.

4.4.2 Stripboard

Stripboards are a piece of material with non-conductive plastic on one side and copper tracks on the other. They also have a lot of holes arranged into rows and columns in order to place components in certain places. Strip boards are used for circuits that are going to be permanently soldered onto the board. Usually, they are used for simple or small circuits with no more than one or two IC chips. They are beneficial in that there are no other preparations to be made to the board other than cutting the size needed. A disadvantage of these stripboards is that it’s extremely easy to make an error. There are so many holes and placed so close together that a small soldering mistake can cause a big mess. Another major downfall with using the strip board is that just by handling it, the sweat from a person's hand could corrode the copper tracks. This board would save the team money, but it would not look as appealing or work as efficiently as the other boards considered.

4.4.3 Printed Circuit Board (PCB)

A Printed Circuit Board, or PCB, is usually a board made out of heat resistant materials that contains complex circuitry made from copper tracks. Various electronic components can be soldered on it to control the flow of electricity. The prototype was controlled by the circuit designed by the team on the board. Many things had to be taken into account to when the team chose the type of PCB. The choice was based on many major aspects, such as, its performance, size, aesthetics, or just overall practicality. There are many types of PCBs made for specific reasons, meant to endure specific types of environments or withstand a fair amount of harsh treatment. Below are the different types of PCB board and why they would’ve or wouldn’t have been beneficial in the design.

4.4.3.1 Single Sided PCB

Single Sided PCBs is the simplest type and naturally costs the least of all boards because the circuit is basically printed on only one side of the printed circuit board. This PCB consists of one single layer of the base substrate, which is usually of fiberglass or silicone, and one single conductive layer, usually made out of copper. The vias, or holes, the elements are usually not plated through and the electronic components are commonly placed on one side of the board while the conductive circuit is placed on the other. Having a single sided PCB could be advantageous because it would really save space and allow a lower height requirement for the design. The downfall would be the cost of having more material.

4.4.3.2 Double Layer PCB

Double layer PCBs have a single layer of the base substrate in between two conductive copper layers and the holes must be plated completely through for electricity to reach every part of the circuit without interference. Both through-hole electronic components and Surface Mount Components (SMD) can be soldered on either side of this type of PCB. Solder mask can be applied on both sides of the board as a means of prevention for any kind of short circuit. Having a double layer PCB could be beneficial because it would have saved the team money on small areas such as shipment. A double-sided PCB would also allow for less surface area, therefore, less material would be used and shipped.

4.4.3.3 Multilayer PCB

A printed circuit board is considered a multilayer board when it consists of more than two layers but must have a minimum of three conductive copper layers. These layers must be paired and alternated with a heat-protected insulation layer that's laminated or glued together in order to operate without any disturbances between them. In addition, there can only be a maximum of 40 even layers. So, when a conductive layer of copper is added, there is another non-conductive layer of silicone. All of the layers are interconnected through the use of copper plating and components are placed on the top and bottom layers of the PCB because the inner-stack layers are used for routing and ground. Even though there are multiple layers, this type of printed circuit board is still compatible with both through hole and surface mount components. This type of PCB is beneficial for extremely complex circuits because it allows each section to be separated by a non-conductive layer. Compared to the single and double layered PCB, it's more durable and reliable considering its quality, and protective layer. It's also more powerful because of its high-density build, being able to achieve greater functionality of capacity, and speed but increasing the surface area. The downfall of this PCB is that it's higher cost for manufacturing and has a higher production time compared to the other PCB designs; availability for this PCB type is also limited.

4.4.3.4 Rigid PCB

A Rigid Printed Circuit Board is a board that does not allow any type of deformity in the material so it cannot be bent or forced out of shape. The material it's made of is called FR4, a fiberglass enforced epoxy laminate that does not allow any type of movement. Once a Rigid Board is manufactured, it cannot be modified or folded into any other form or shape. Its copper path allows for the connectivity of different components. This PCB can be single sided, double sided, or multilayered and these types of PCB boards can be used in everyday applications, such as, our handheld devices, medical equipment, the automotive industry, or even aerospace. A few benefits of a Rigid PCB were that they're typically cheaper and more durable compared to the Flex PCBs, they are high quality and high density, and there's a higher demand and supply of them, making them easily accessible.

4.4.3.5 Flex PCB

Flex PCBs are made of flexible material that can be folded or bent easily, allowing easy handling and transportation without damage. The material the board is made out of is a flexible thin insulating plastic polymer film and traces of conductive copper are within this protective layer of film. A Flex PCB can also be either single sided, double-sided, or multilayered. These types of PCBs are used in several electronic devices due to its properties and they are extremely beneficial in numerous amounts of ways. A few ways this type of PCB would've been beneficial to the project are their low thickness level, reducing the size, space, and weight requirement, its flexibility would allow for folding and bending when necessary, they're ideal for 3D interconnection assembly, which would allow increased design freedom, and increased resistance against shock and vibration. It would improve the chess board aesthetically. Some disadvantages are the high cost and its ease of being damaged by the user with reworking or soldering.

4.4.4 Microcontroller / Microprocessor

The microcontroller (MCU), or microprocessor (MPU), is the main control unit to function and communicate between the different devices within the system. There are many types of microcontrollers and microprocessors that could've been used for this design, but for the chess board being designed and built to be most efficient there were some requirements that needed to be met. Some of the more important conditions considered were cost, memory capacity, the amount of General-Purpose Input Output (GPIO) pins, UART and SPI busses, clock speed, power consumption, and of course, the language support since programming was quite a challenge for a team of all electrical engineering students.

Microcontrollers and microprocessors contain the CPU and external memory that stores and run the program. The main difference between them is that MCUs store and execute their programs using read-only memory, whereas MPUs do not have read-only memory on the chip and they run and execute their programs from the RAM. The many options available have different ranges and an extensive amount of varied specifications.

4.4.4.1 Arduino IDE

Arduinos are very common boards with a wide range of options that has a microcontroller incorporated within its design. They're affordable, easily accessible, and made for the consumer's convenience in the sense that they're simple to use, customizable, their cross-platform capability, and its open source and magnitude of software and hardware availability. Specifically, the Arduino UNO, it is one of the cheapest yet simplest boards that's used on the market.

The microcontroller the board supports is the ATmega328P. The board possesses everything needed to support this microcontroller. The Arduino UNO is widely used amongst beginners and experts, therefore as a team with very little experience in programming, this was a great option for prototyping. The reason for its popularity is that it has an open source IDE with a simple syntax based on ‘C’ language. The team has been exposed to C language; therefore, it would be easier for the members to grasp and comprehend the coding prototype without wasting time learning a new programming language. Other features of the board are listed below.

- 32 KB of Flash memory
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 14 (6 pins with PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA.

Another microcontroller by Arduino that stood out during research and recommended to the team was the Arduino ATmega2560. It contains everything that’s required to support the microcontroller. The major specifications are emphasized below.

- Operating Voltage: 5V
- Input Voltage: 7-12V
- CPU Speed: 16MHz
- Number of I/O Pins: 86
- Embedded Interface: I2C, SPI, USART
- Flash Memory: 256KB (8KB used by bootloader)
- SRAM: 8KB
- Clock Speed 16MHz
- USB Port

4.4.4.2 Raspberry Pi

The Raspberry Pi is typically a general-purpose computer. It’s one of the top-of-the-line development boards because if its credit card size. It’s easily integrated with hardware such as a monitor, CPU, TV, and even a standard mouse and keyboard. It is mostly used with Internet of Things (IOT) related applications due to its Bluetooth and wireless capabilities. This board would have been a great choice if a web application or some type of camera interface were to be integrated into the prototype. One major drawback about this board is that it runs on a customized Debian Linux called Raspbian, an operating system which requires the team to have to learn a programming language such as Python, Java etc. A few details about the Raspberry Pi are highlighted below.

- Processor: 1.2GHz, 64-bit quad-core ARMv8 CPU
- 802.11n Wireless LAN
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)
- 1GB RAM
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot
- VideoCore IV 3D graphics core

4.4.4.3 TI MSP430

The TI MSP430 is designed as an ultra-low power system and low-cost architecture that's very powerful in prototyping consisting of many other different components suitable for various applications. It allows users to increase performance on a low budget and compared to the other microcontrollers that were researched, the TI MSP430 would've also been a great option because of its low power consumption, affordability, and most importantly, it's a more familiar choice for the entire team. From the classes taken, the use of the MSP430 was required and used to program and implement simple codes. Two types of the MSP430 microprocessors were investigated and Table 7 below compares the MSP430G2553 and the MSP430FR6989.

Table 7: TI MSP430 Comparison, G2553 Vs. FR6989.

Component	MSP430G2553	MSP430FR6989
MCU	16MHz	16MHz
Memory	16KB Flash	128KB Nonvolatile
Input/Output Pins	24	83
Analog-to-Digital Converter (ADC)	10-bit	12-bit
Timers	2 16-Bit With 3 Capture/Compare Registers.	5 16-Bit With up to 7 Capture/Compare Registers Each.

4.4.4.4 PIC18F46K80

The PIC18F46K80 microchip was an option for the chess design for its high performance and excellent features. Some of the key characteristics that stood out from this microchip is its alternate run modes, helping to reduce power consumption, multiple idle modes, allowing the CPU to be disabled while all peripherals are still actively operating, permitting even less power consumption, and its low cost and accessibility. Below are some of the features.

- Flash memory
- 64KB Memory
- 16 CPU speed (MIPS/DMIPS)
- 3648KB SRAM
- Digital Communication Peripherals: 2 UART, 1 SPI, 1 12C-SSP(SPI/I2C)
- 2x8-bit, 3x16-bit Timers
- 11 channel, 12-bit ADC input
- 2 comparators
- Operating voltage from 1.8V to 5.5V
- 44 Pins

4.4.4.5 LPC2148

The last microcontroller that was researched and explored for this project was NXP's LPC2148 by Philips Semiconductor. Its design has many built-in features and peripherals making it a very efficient and reliable choice. Its versatile design makes it easy to work with, being applicable in industrial control and medical systems. Their size and minimal power utilization furthermore adds its design to the category of why this microcontroller would have been a good option for the chess board. Some of the major details are listed below.

- 512kB Flash memory
- 32-bit CPU
- 60MHz operating frequency
- 2.0 USB
- Serial communications: 2.0 USB interface, 2 UARTs, 2 timers, 2 I2C, 2 SPIs
- 45 GPIO
- Power capability of 3.3V to 5.5V

4.5 Power System

This section will cover the different types of power supplies that were researched and put under consideration for the project. The different advantages and disadvantages associated with them in conjunction with the project, will be highlighted.

The Smart Chess Board have many components that consume power; therefore, it's necessary to provide a substantial amount of power in order to operate the system. The two main devices that requires the most power is the electromagnet and the motors operated to control and move the chess pieces around the chess board. Trying to keep the power consumption as low as possible, all components were carefully selected based on their voltage and current usage. According to the research that was taken place, the chess board requires at least a power supply of 12 DC voltages. This decision was made because the electromagnet required the lowest amount of power.

The original plan was to make the whole project battery operated, but as more research took place, the team quickly determined that using a standard wall outlet as the power source would be most beneficial. Making the entire project battery operated would be difficult based on the components being used. It would also have to operate efficiently with the batteries and not have them drain fully within a couple of minutes. The decision was made as a team that there was not enough time to incorporate this into the main goals of the project but instead, made it an optional feature.

4.5.1 Power Supply

Depending on the type of computer system chosen to run all of the software for this project, the appropriate power supply was then determined. While researching, it was likely that either multiple power supplies or a single power supply with multiple different outputs was needed. To power the computer system, which was assumed to probably be a Raspberry Pi 4, a 15W USB-C power supply would've been needed. Raspberry Pi makes their own official power supply for the board which can be bought directly from them; however, this power supply is designed for that one board and could not be used to power any other components unless they require the same power. If this was used then different power supply, or power supplies, would be needed to power the XY-plotter, the electromagnet, the RFID scanner, the LEDs, etc.

The most efficient and cheapest way to power all of the components was to buy a power supply that has multiple configurable outputs. This way the computer, and all of the other peripherals could be powered from a single power supply that is connected to a wall outlet. It was preferable to buy a power supply rather than designing and making one ourselves because if mistakes were to be made while constructing a custom power supply, some of the electronics could be damaged and new parts would need to be bought, wasting more time and adding to costs. To minimize that risk, the appropriate power supply was purchased.

4.5.2 AC/DC Converter

A very important component in this design is the AC adapter (wall wart). To power every element on the chess board for the system to function, a standard wall outlet as the power source with an input of AC current and DC output, would make the project more convenient for the team and the user. It made it convenient for the team because the project was able to be tested almost anywhere and does not need a specific power source to test it. It was also a more suitable choice for the users considering there wouldn't be a hassle of having to charge or replace the battery but easily be transported to any setting desired by the user as long as there is a standard power outlet available.

The two main types of AC/DC converters that were considered and researched were the generic standard wall adaptors and the AC/DC adapters with a USB port. The standard outlet is a block with a wire already attached to it so all that has to be done is to plug one end into the wall and the other end, the barrel power jack, into the device. The wall wart with a USB port, which is a block that goes into the wall outlet and the other end is a USB port for many kinds of different cable wires. The USB wall wart could've been a very beneficial option since it's already there for the program inputs.

4.5.3 Voltage Regulator

Every electronic device needs some type of power source to function and voltage is constantly fluctuating. In the chess board, there are many components that requires voltages and currents, and they won't be consistent with one another. With all of the components within it, the inconsistency could really do some damage, and possibly even destroy the components. Therefore, a voltage regulator is required to protect and maintain a fixed output continuously regardless of the input voltage. The two main and different types of voltage regulators are Linear Voltage Regulators and Switch Voltage Regulators.

4.5.3.1 Linear Voltage Regulators

Linear voltage regulators use a transformer to reduce or drop the AC voltage and then converts it to an acceptable DC voltage using rectifiers and filtering. Some of the advantages of this regulator is that they're simple and easy to use, they're affordable and quiet, considering there are no high-frequency switching, and they're a great option when needed for low power outputs. The disadvantages are that they're larger in size, making them heavier compared to the switch voltage regulator, they're step down, meaning their output voltage must be lower than the input voltage, and they're not as efficient. Some of the common applications for linear voltage regulators are for communication, medical, and laboratory devices. There are two different types, series and shunt.

4.5.3.1.1 Series

A series voltage regulator is connected in series with the load in order to stabilize the output voltage. This is done by changing the resistance so that the voltage drop can be varied so the load remains constant. One of the major advantages of the series regulator is that it doesn't draw all the current, making it more efficient than the shunt regulator. A way to improve its performance is to use a feedback network in the voltage regulator so that the error is used to regulate the output voltage.

4.5.3.1.2 Shunt

Alternatively, a shunt voltage regulator is in series with a resistor and the voltage source, and then connected in parallel to the load to maintain the output voltage. The most common form of a shunt regulator is a Zener diode. This circuit setup draws out the maximum current from the source no matter what the load current is. Since the voltage is regulated by adjusting the current, this type of linear regulator is less efficient.

4.5.3.2 Switch Voltage Regulators

Switch voltage regulators use pulse width modification (PMW) to regulate the output voltage efficiently. The transistor works as a switch to stay on, saturation region, so that the output voltage is regulated at a constant voltage, or off, cutoff region. Compared to linear voltage regulators, some of the benefits is its efficiency being higher, it's more flexible in different applications, and it's more compact in size. Some of the drawbacks are the high interference and frequency noise, their design is a bit more complex, and they're more expensive. The PMW generates high frequency noise but it contributes to the higher power efficiency and capacity with the switching capability. The switching mechanism is adaptable to almost any power source, whereas the linear voltage regulators must be adjusted to be suitable for the device it's being used for. A few common applications the switching voltage regulator is used for are manufacturing, high power and current applications, aviation, some communication practices, and much more.

4.5.3.2.1 Step-Up

Step-up switching converters, also known as boost switching regulators, outputs a maintained voltage higher than the input voltage. It's designed to increase or "boost" the DC voltage with the transistor switched on, saturation mode, energy is passed through the transistor and inductor, and back to the supplied voltage, therefore, increasing current flow. Step-up converters are commonly used in applications involving capacitors, such as, battery chargers and photo flashes.

4.5.3.2.2 Step-Down

Step-down switching converters, also called a buck converter, is very much like a linear voltage regulator where its desired output is a lower voltage than the input voltage. Its advantage to the linear regulator is that it's more power efficient by using less power to function. They're the more common option in the switching regulator category.

4.5.3.2.3 Inverter Voltage Regulator

An inverter voltage regulator is a hybrid of both the buck and boost, or step-up and step-down switching regulator in one circuit. The main difference being that it includes a control unit. With an input power at any voltage, the control unit is capable of sensing it and would take the appropriate measures to output the fixed voltage set. This regulator can be bought where the output voltage can be fixed or adjustable, making it convenient and flexible, but it'll be at a higher cost for the extra components required to make this work.

4.5.3.3 Voltage Regulator Comparison

There were quite a few selections for the type of regulator that could have worked for Smart Chess Board. The two main types of regulators that the team had to choose between were linear regulators and switching regulators. Even though the two were discussed before, in this section, each regulator type will be elaborated of its importance to the project. In a linear voltage regulator, a resistive load is used in order to regulate the output. The two different types are series or shunt, depending on the arrangement of the control elements. Table 8 below presents the pros and cons of a linear voltage regulator.

Table 8: Pros and Cons of Linear Voltage Regulator.

Linear Voltage Regulator	
Advantages	**Disadvantages**
• Simple circuit configuration. • Few external parts. • Low noise. • Robust in overcurrent protection and thermal protection. • Cheaper.	• Relatively poor efficiency. • Considerable heat generation. • Only step-down operation. • Efficiency of linear regulators are usually 20% - 60%.

A switching regulator uses a switching element in order to transform the incoming power supply into a pulsed voltage. The pulsed voltage is then smoothed out using capacitors, and inductors. The switching element is usually a MOSFET. When power is supplied from the input, it turns on the MOSFET until the output reaches a predetermined value then the switching element is turned off. The advantages and disadvantages will be shown in Table 9 below.

Table 9: Pros and Cons of Switching Voltage Regulator.

Switching Voltage Regulator	
Advantages	**Disadvantages**
• Higher efficiency, up to 95% can be achieved. • Low heat generation because less power is wasted. • Negative voltage operation.	• More external parts required. • Complicated design. • Increased noise due to frequent switching.

4.5.4 Batteries

The use of batteries was one of the desired features that the team would've liked to add into the design to further portability if time was spared. Batteries are used to power numerous types of electronics constantly, on a daily basis. They are usually divided into two major categories, primary batteries and secondary batteries. The material or chemical elements that the batteries are made out of determine the battery life and the amount of times they can be recharged if rechargeable. Both primary and secondary batteries are useful but they each have their strengths and weaknesses associated with this specific project. Different types of primary and secondary batteries are listed below.

4.5.4.1 Primary Batteries

Primary batteries are batteries that are non-rechargeable. They can only be used once and disposed of afterwards. One of the issues that came about while researching was if primary batteries were to be used, the team would have to take into consideration what to do when the battery dies. How will the chess board be powered back up? Conveniently, the batteries could've just been interchanged with new ones but depending on where they would be located, it could've been a greater hassle. Therefore, more time and possibly an increase in costs would be required to reconstruct the design to comply with the project.

4.5.4.1.1 Alkaline Batteries

Alkaline batteries are the most common types of batteries in the world, most of our everyday electronics use these types of batteries. They are non-rechargeable, disposable, with high energy density, and they have a long-life span without losing its capacity in storage. The main chemical compound in the battery is potassium hydroxide, an electrolyte, where the name alkaline is derived from. Other parts, such as, the cathode is made of manganese dioxide and the anode is made with zinc powder. They offer fine energy density as well as leak resistance and they come in many sizes to accommodate certain designs.

The typical values of that are supplied by a single alkaline battery of 1.5V supplied and 700mA.The supply voltage level of an alkaline cell decreases over time and if the supply voltage is below a certain level, it would not be able to provide enough power to turn the device on, which wouldn't be very proficient in our design.

4.5.4.1.2 Zinc-Carbon Batteries

Zinc-carbon batteries are also known as dry cell batteries. The electrolyte used in this battery is a mixture of ammonium chloride and zinc chloride. The cathode is a mixture of carbon powder and manganese dioxide, manganese helps to increase the conductivity. This whole mixture of the cathode is then packed into a zinc container which will act as the anode. The typical voltage value is usually less than 1.5V. Zinc-Carbon batteries are extremely durable and has a long-life span offered in various sizes. Additionally, these batteries work well in moderate temperatures, but they do not operate well in low temperatures.

Though it has some advantages, the disadvantages would've made them a more troubling choice. Due to their casing design, there is a very high chance of corrosion. They're cheap but if the team were to spend a bit more, it would be well worth the better quality and longer lasting battery.

4.5.4.1.3 Mercury Batteries

These types of batteries are also a part of the primary battery group which means they are non-rechargeable. Mercury batteries are made out of mercuric oxide with manganese dioxide, and also known as an electrochemical battery. These types of chemicals are extremely hazardous to humans. Although the mercury battery is the less popular choice, it is useful for photographic light meters and the real time clock of the CPU. It is less popular because of its lack to provide an output voltage higher than 1.35V. This type of battery would not be a good choice because it would fail to provide an adequate amount of voltage and current.

4.5.4.2 Secondary Batteries

Secondary batteries are batteries that can be recharged numerous times after it is dead. If secondary batteries were to be used, it would be the more beneficial option because replacing the batteries wouldn't be necessary, but on the downside, it would be a bit costlier and the battery would have to be compatible with the power source available to be able to recharge. Therefore, more research would be required for the best selection.

4.5.4.2.1 Lead-Acid Batteries

Lead-acid battery are the first types of rechargeable batteries invented in 1859. The combination of elements that make up this battery are lead, lead dioxide, and sulfuric acid as the electrolyte. Lead acid is common because of its dependability and its cost. These batteries are used for mainly for starting, lighting and ignition systems (SLI). This is so because this type of battery can fulfil the high current requirements of heavy motors. Lead-acid batteries are mainly used in applications such as automobiles, personal computers, telephone exchanges, portable emergency lights, and to store charge for a solar panel that's set up on.

Its ability to be able to recharge is the one of the main advantages. Though it would be very beneficial to be able to recharge the battery to operate our design, the disadvantage of the battery being extremely heavy just outweighs that factor. Just the weight of the battery alone would make the whole portability aspect of the design extremely difficult. A Lead-Acid battery typically weighs on average from about 19lbs to 32lbs, and with the other components required in the design, it'd be better off stationary to avoid obstruction.

4.5.4.2.2 Lithium Batteries

Lithium batteries are also a type of rechargeable battery within the secondary battery group. In fact, lithium ion batteries are some of the most energetic rechargeable batteries available on the market. The chemical compound of the lithium battery can affect its output voltages and can produce anywhere from 1.5V to 3.7V. They are extremely popular in the product market for they provide power to everyday electronics such as laptops, cellphones, and so much more. This type of battery seemed like it would be an optimal choice just based on the type of applications it's used for.

Lithium is also a highly reactive element, meaning that a lot of energy can be stored in them. A typical lithium-ion battery can store 150 watt-hours of electricity in 1 kilogram of battery. This would've made them a far better contestant than the other types of batteries, compared to the lead-acid battery which can only store 25 watt-hours per kilogram. There can be a downfall to the amount of energy the battery can store and over time. The more charges the battery consumes, the less capacity it possesses. Lithium ion batteries have been known to burst into flames occasionally because of the lithium metal being unstable during charging. Although not very common, this can be dangerous if it were to malfunction.

4.5.4.3 Batteries Comparison

If the team were to go above and beyond, a solar powered chess board would've been ideal, making it incredibly innovative. But functioning off of batteries were more realistic at this stage. If the group were able to implement batteries into the chess board design, there are many pros and cons to each type, primary and secondary, and it can be seen in Table 10 below.

Table 10: Primary Batteries Vs. Secondary Batteries.

Battery Type	Advantages	Disadvantages
Primary Batteries	• Typically, cheaper. • High energy density. • Ready to use when bought. • Lower discharge rate.	• Not rechargeable, one-time use. • Accumulates waste. • Not environmentally friendly, if not disposed of properly.
Secondary Batteries	• Rechargeable. • Cheaper in the long run. • Can retain charge for longer time.	• Demands more power. • Initially cost more. • Considered to be hazardous waste.

4.6 LCD/LED Screen Display

Another part of the design that was incorporated to aid in ease of use and understanding for the player is a screen display. It was intended to provide a text display of what the microcontroller wants to say. For instance, stating the start and end of the game, whose turn, and asking what chess piece and where the piece is to go. So, if the player were to miss any statements from the microcontroller, the comment would be shown on the display screen.

4.6.1 Liquid Crystal Display (LCD) Screen

Liquid crystal display, also known as LCD, consist of microscopic molecules between two plates, one of a piece of glass and another of an opaque substrate. Once the molecules align in a certain way from applying an electric charge, light trying to enter is blocked and the desired image would come into view, otherwise, it's just transparent. It was an important component that the team wanted to include within the design for the system output for notifying the user and player of what is happening at that very moment. There are quite a few types of LCDs that exist and what potentially determined the best selection was the cost, efficiency, and compatibility.

4.6.1.1 LCD Technologies

Some types of LCD technologies are Blue Mode STN, Film STN, Colour STN, and Double STN. The technology determines the clarity and color of the display screen. The distinctions are listed below.

- Blue Mode STN (super-twisted nematic) is the simplest and most basic of the LCD screens. It's generally used in a negative application where there's a blue background and white pixels. It's cheap and useful, but depending on viewing angle, the contrast can make it hard to see what is being displayed.
- Film STN is an improvement from Blue Mode STN for the extra layer of film to present an increase in sharpness and contrast for a better visual of the display. This type of LCD is still reasonable in cost and power efficient.
- Colour STN has added layers of colored filters to generate a display of up to 65,000 various colors.
- Double STN provides an enhancement in contrast while eliminating any other colors on the display screen, providing a sharper display.

4.6.1.2 Displayed Data

An additional component that plays a crucial role in the LCD was its ability to display data. After the technology is selected, there are a few different options of what kind of information can be presented on the screen.

- Segment LCDs are old fashioned and can display data such as numbers, letters and fixed symbols.
- Graphical LCDs are pixels that are organized in rows and columns and the characters can be displayed once the set of pixels are energized.
- Color LCD
    - Passive Matrix
    - Active Matrix

4.6.2 OLED

Another option for a display was the organic light-emitting diode, also known as OLED. It's replacing many LCD displays applications, such as, mobile devices, televisions and micro-displays. The main difference between LCDs and OLEDs is that the LCD's are transmissive, where pixels are illuminated with the backlight, whereas the OLED's are emissive, where the pixels produce their own light, emitting light when electricity is applied. The leading downfalls of OLED is the lower lifespan and possible screen burn-in. OLEDs can also commonly have discoloration in certain areas of the screen, a malfunction in the pixels.

4.6.3 Comparison

There are a few pros and cons to both LCDs and OLEDs. Some being that the LCD displays are thicker than OLEDs, making them heavier, and they require more power because of the backlight. Some of the downfall of OLEDs are they lose brightness over time and they're more expensive because of its new technology and developing improvements over time. Table 11 below shows the main specifications comparing LCDs and OLEDs.

Table 11: LCD Vs. OLED Comparison.

Parameter	LCDs	OLEDs
Response Time	Slower (approximately 1ms)	Faster (approximately 0.01ms)
Contrast	Acceptable	Life-like
Pixels	Transmissive	Emissive
Thickness	Thicker than OLED.	Thinner than LCD.
Power Consumption	Higher, must power backlight.	Lower, no backlight power.
Display	Doesn't lose brightness but backlight could fail.	Loses brightness over time.
Cost	Price ranges from affordable to expensive, depending on size. Cheaper than OLED.	Price ranges from affordable to expensive, depending on size. More expensive than LCD.
Viewing Angle	Limited to approximately 50 degrees.	Approximately 80 degrees.

Both the LCD and OLED screens had its benefits, but because of cost restrictions, compatibility, ease of use, and the resources provided, there were a few LCDs that stood out while researching. The comparison of the ones that were looked into are stated below.

Generic TFT LCD Display Touch Screen Replacement Screen from Aliexpress:

- Supports development boards, easily to the Arduino UNO R3 and Mega 2560.
- 3.95-inch color screen
- Effective display area: 83.52mm x 55.68mm
- Module PCB size: 96.52mm x 61.47mm
- 480 x 320-pixel resolution, clear display
- Thin Film Transistor (TFT)
- 8-bit parallel interface
- On-board 5V/3.3V level shifting IC
- Power consumption 5V at 150MA
- Only includes the LCD screen
- Cost: approximately $11

Adafruit RGB LCD Shield Kit

- Supports Arduino
- Dimensions: 2.1'' x 3.2''
- 16 x 2- character RGB backlight LCD
- 3 backlight pins
- 6 tactile switch buttons
- Adjustable contrast
- Instructions and basic code included
- Approximately $24

Elegoo 2.8-inch TFT Touch Screen

- Supports Arduino UNO R3 and Mega 2560
- 240 x 320-pixel resolution
- PCB module display size: 78.22mm x 52.7mm
- LCD area (W x H x T): 50mm x 69.2mm x 2.5mm
- 8-bit parallel interface
- Compatible with 5V or 3V MCU with 5V-3.3V change-over circuit
- Includes micro-SD card slot
- Approximately $15

4.7 Motor Control

In order to move the chess pieces to the correct position as accurately and as precise as possible, certain types of motors was considered. The motors researched usually have a shaft that spin in a circular motion but would have to be converted into linear motion for the necessities of this project. There were a few other properties that had to be considered in order for the motor to work in harmony with the other components. Some of these features included cost, shape, size, speed, acceleration, and overall compatibility with the magnetic system.

4.7.1 DC Motors

DC motors collaborates magnetic fields and conductors converting electrical energy to mechanical energy. There are a wide variety of DC motors, but the two common ones are brushed and brushless. Brushed motors contain coils, magnets, or a stator, that encloses the coils in an electric field. They are used in industrial applications and other common applications such as mobile phones, toys, cordless drills, and car windows. Brushless motors are simpler. The only moving component being the rotor, eliminating the need for brushes. Polarity is the main concern for this motor since it's more difficult to control. Because of its efficiency and durability, they have a wide range of applications. A few utilizations for these brushless motors are washing machines, air conditioning units, and basic electronics, such as, computer fans. Table 12 below exhibits the main difference of brushed and brushless DC Motors.

Table 12: DC Motors: Brush and Brushless Comparison.

DC Motors	
Brushed	**Brushless**
• Simple to control. • Excellent torque at low speeds. • Efficiency of 75-80% • Affordable. • Noise from rubbing parts. • Can potentially cause interference with other parts. • Brushes get worn out, needs constant maintenance.	• Difficult to control. • Requires special regulator. • Generates less noise. • More efficient, able to continuously achieve maximum rotational force/torque. • High durability. • Less maintenance.

4.7.2 Stepper Motors

A stepper motor is an electromechanical device that has both electrical and mechanical components that is converted from electrical power to mechanical power. Stepper motors are specialized types of motors that are made of electromagnets in order to obtain precision rotation in a clockwise or a counterclockwise direction. The motor is made from a permanent magnet rotating shaft, called a rotor, and stationary electromagnetics arranged in cyclical patterns, called the stator. When these magnets energized on and off in a particular order, it will then effectively turn the shaft a certain amount of degrees, which is referred to as a "step."

A rotor may consist of 50-100 different north (N) and south (S) poles. The accuracy usually depends on the number of poles, the more N and S poles there are, the fewer the degree of rotation gets, therefore, causing smaller increments in steps. Stepper motors may have up to 200-400 steps per revolution. An important aspect about stepper motors that should be known is that it uses open loop control, hence, it does not provide or rely on feedback. They can generate high torque for quick acceleration and response, have exceptional speed control, and excellent precision. All these features made the stepper motors a great candidate for this application. Table 13 below shows the advantages and disadvantages of stepper motors.

Table 13: Pros and Cons of Stepper Motors.

Stepper Motors	
Advantages	**Disadvantages**
• High pole count from 50 – 100, making precise steps and excel in applications that require precise positioning. • Precise increments in movement, enabling excellent speed control. • Maximum torque at low speeds, suitable for applications that need low speed with high precision. • Can easily be controlled by microcontrollers i.e. ATmega chips.	• Generates noise while operating. • Less torque at high speed. • Low efficiency, current consumption is independent of load, constantly drawing maximum current. • May skip steps at high load.

4.7.3 Servo Motors

Servo motors are usually small in size but are very energy efficient. The motor is attached by gears to the control wheel and as the motor rotates, the potentiometer's resistance changes, so the control circuit can precisely regulate the movement and the direction. The small DC motor runs from a power source and spins at a high RPM and discharges low torque and the gear arrangement will do the opposite. The gears can slow down the servo’s motor shaft. Inside a servo motor contains a small DC motor, potentiometer, and a control circuit. This feature allows them to be operated in toy cars, robots, and airplanes. They can also be used in industrial applications that deal with a manufacturing line such as pharmaceutics and food services.

One of the biggest differences between a stepper motor and a servo motor is that a servo uses a closed-loop feedback system and a stepper motor uses an open-loop feedback system. This means that the motor uses a feedback sensor to precisely control its angular position. Using Pulse width modulation (PWM), servo motors provide a high-performance alternative to stepper motors. When an electrical pulse is sent, the motor speed will be proportional to the difference between the actual and the desired position. Therefore, the closer the motor shift is to its desired location, the slower it will move. This saves a great amount of energy because it uses just enough energy to get the job done, which was more practical for this design. The pros and cons of servo motors are presented below in Table 14.

Table 14: Pros and Cons of Servo Motors.

Servo Motors	
Advantages	**Disadvantages**
• Have high torque and best suited for applications with high speeds and torque, at speeds greater than 2000 rpm. • Operate under constant closed-loop feedback allowing higher speed and peak torque. • Comes in many sizes and torque ratings. • Affordable.	• Limited range of motion, positional rotation limited to 180 degrees of motion. • Feedback mechanism constantly adjusts to correct drift resulting in twitching.

4.8 Piece Detection System

There were three major types of piece detection systems that could've been used for Smart Chess Board. The first was a memory-based system where the computer has a model of the chess board at the beginning of the game and updates it at the end of every move with the updated positions, essentially "remembering" each move that has been made since the start of the game. The second type of system consists of using RFID tags fixed to each of the pieces which gives them all a unique identifier, and an RFID scanner that would be presumably mounted on the piece movement system. The third and last that was researched consists of a set of Hall effect sensors embedded into the board that can check if there is a piece on any given tile at any given time, by detecting the magnetic fields emitted by the magnets on the pieces themselves.

Research was done on different technologies that could be used to create a piece detection system, and their properties were compared to decide on which one would be the most effective for this project. The major technologies that could be used include RFID and Hall effect sensors.

4.8.1 RFID

Radio Frequency Identification (RFID) is the wireless non-contact use of radio frequency waves to transfer data. RFID tags can be embedded in different types of products to allow the user to uniquely identify and track them using an RFID reader. RFID Tags can be read without line-of-sight and can have a range between a few centimeters to over 20 meters depending on the type of RFID system chosen.

RFID systems can be classified into three groups depending on the frequency range they operate in. Low frequency devices generally operate in the range of 30 to 300kHz, High frequency devices operate at 13.56MHz, and Ultra-High frequency devices operating at 300 to 3000MHz. Usually, higher frequency devices have a longer read range but may experience more interference from liquids and metals than lower frequency devices. Any of these types of systems could potentially be used for a piece detection system. Below in Table 15 is a more in-depth description of some of the more common RFID systems.

Table 15: RFID Frequency Bands.

Band	Regulations	Range	Data speed	ISO/IEC 18000 section	Remarks	Approximate tag cost in volume (2006) US $
120–150 kHz (LF)	Unregulated	10 cm	Low	Part 2	Animal identification, factory data collection	$1
13.56 MHz (HF)	ISM band worldwide	10 cm–1 m	Low to moderate	Part 3	Smart cards (ISO/IEC 15693, ISO/IEC 14443 A, B). ISO-non-compliant memory cards (Mifare Classic, iCLASS, Legic, Felica ...). ISO-compatible microprocessor cards (Desfire EV1, Seos)	$0.50 to $5
433 MHz (UHF)	Short range devices	1–100 m	Moderate	Part 7	Defense applications, with active tags	$5
865–868 MHz (Europe) 902–928 MHz (North America) UHF	ISM band	1–12 m	Moderate to high	Part 6	EAN, various standards; used by railroads[15]	$0.15 (passive tags)
2450–5800 MHz (microwave)	ISM band	1–2 m	High	Part 4	802.11 WLAN, Bluetooth standards	$25 (active tags)
3.1–10 GHz (microwave)	Ultra wide band	Up to 200 m	High	Not defined	Requires semi-active or active tags	$5 projected

Image Credit: https://en.wikipedia.org/wiki/Radio-frequency_identification#Frequencies

Besides their operational frequency ranges, RFID systems can also be classified by the type of tag and reader. In a Passive Reader Active Tag (PRAT) system, the tag periodically sends radio signals that are received by the passive reader. In an Active Reader Passive Tag (ARPT), the reader periodically sends interrogator radio signals, receiving authentication signal replies from passive tags. Active Reader Active Tag (ARAT) systems have an active reader that interacts with battery-assisted passive tags. For the purposes of the Piece Detection System developed for this project, the tags would be required to be passive since they need to be placed in the chess pieces and there would not be enough space to house the battery of an active tag. Thus, an Active Reader Passive Tag (ARPT) would've been the best choice for this project.

4.8.2 Hall Effect Sensors

Hall effect sensors are devices that are activated by an external magnetic field. The output signal from a Hall effect sensor is the function of the magnetic field density around the device. When the magnetic flux density around the sensor exceeds a certain pre-set threshold, the sensor detects it and generates an output voltage. This allows Hall effect sensors to detect whether a magnetic field, such as a permanent magnet fixed to a chess piece, is within close proximity.

Hall effect sensors can be made to produce either linear or digital outputs. Linear, or analog, sensors produce an output voltage directly proportional to the magnetic field passing through the Hall sensor. Digital sensors have only two possible outputs: an “OFF” condition and an “ON” condition which is triggered once the magnetic flux passing through the sensor exceeds a pre-set threshold. More specifically, there are two basic types of digital Hall effect sensors: bipolar and unipolar. Bipolar sensors require a positive magnetic field (south pole) to operate them and a negative magnetic field (north pole) to release them. Unipolar sensors require only a single magnetic south pole to both operate and release them as they move in and out of the magnetic field.

In many cases, Hall effect sensors can be activated by a permanent magnet attached to a moving object which moves close to the sensor. There are different ways in which the magnet can move in relation to the sensor, and the type of magnet movement has implications on the operation and applications of the sensor. No matter what type of magnet movement occurs, one constant requirement is that the magnetic flux lines of the permanent magnet must always be perpendicular to the sensing area of the device and must be of the correct polarity. Two of the most common types of sensing configurations used include head-on detection and sideways detection.

Head-on detection implies that the magnetic field is perpendicular to the Hall effect sensing device and that it approaches the sensor straight on from the active face for detection. This approach generates an output signal which represents the magnetic flux density as a function of the distance away from the sensor, assuming a linear device is used. The nearer to the sensor the magnet gets, and therefore the stronger the magnetic field, the greater the output voltage of the Hall effect sensor and vice versa. This can also be used to differentiate the polarity (positive or negative) of magnetic fields. This configuration can also be used to create a positional detector by setting up the non-linear device to trigger the output “ON” at a pre-set air gap distance away from the magnet.

Sideways detection consists of moving the magnet across the face of the Hall effect sensor in a sideways motion. This is useful for detecting the presence of a magnetic field as it moves across the face of the Hall element within a fixed air gap distance. For example, counting rotational magnets or the speed of rotation of motors are some applications of the sideways detection configuration.

4.8.3 Memory-Based

It’s possible to create a memory-based piece detection system that utilizes the chess ruleset from a chess engine. This would work by writing a program in the Arduino IDE which is run on the microcontroller. This program first translates the signals from the voice recognition program (BitVoicer) into a usable format. Then it checks whether the move is possible according to the rules of chess, and whether a piece is to be captured during the move. If the move is not possible, a message will be sent to the player alerting them of this fact and prompting them to make another move. If it’s possible the program will check if an enemy player’s piece is occupying the tile to be moved to. If it is, then the piece to be captured will be moved to the graveyard and then the piece to be moved will take its place. If not, then the piece to be moved directly goes to the location the player commands it to.

4.8.3.1 Mailbox Array (Offset Board Representation)

A “mailbox” array, also called offset board representation, is a square-centric board representation where the encoding of every square resides in a separately addressable memory element, usually an element of an array for random access. It’s called a mailbox because the square number (its file and rank) acts like an address to a post box, which may be empty or may contain one chess piece.

The piece locations are stored in an 8x8 array, which for Tom Kerrigan’s Simple Chess Program (TSCP) is a one-dimensional array containing piece and empty square codes indexed by a square in a 0.63 range which combines rank or file indices in three consecutive bits each. Such a board representation is often used redundantly in bitboard programs to answer the question of which piece (if any) resides on a square efficiently. As the sole board representation, the 8x8 board has some efficiency issues with move generation related to the off the board test. This means that while the piece locations can be stored efficiently in an 8x8 board, it is difficult to determine which moves a given piece can make.

To solve this issue, one of the most common approaches found in chess engines, including the TSCP, is to use a 10x12 board array. The 10x12 board is an expanded mailbox array that consists of the aforementioned 8x8 board array, surrounded by sentinel files and ranks to recognize off the board indices. A sentinel value is a special value in the context of an algorithm which uses its presence as a condition of termination, typically in a loop or recursive algorithm. In the case of the TSCP, the sentinel values are -1. These 10x12 board generates moves using offsets per piece and direction to determine possible move target squares, which is the square that a piece moves to once a successful move is completed.

Thus, the 10x12 board solves the problem of checking each move to see if it is possible for that given piece, while also keeping track of all the piece locations in order to handle piece capturing. This 10x12 board is commonly used in many chess engines, including the TSCP to detect pieces as well as analyze moves for legality. Below in Figure 2 is an example of what a 10x12 board looks like in TSCP with the embedded 8x8 board as well as sentinel values around the edges and how it can be used to figure out which pieces can go where. It shows the values that are stored in each of the array elements. These values represent the indices that the squares would be associated with on an 8x8 chessboard.

```
-1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
-1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
-1,  0,  1,  2,  3,  4,  5,  6,  7, -1,
-1,  8,  9, 10, 11, 12, 13, 14, 15, -1,
-1, 16, 17, 18, 19, 20, 21, 22, 23, -1,
-1, 24, 25, 26, 27, 28, 29, 30, 31, -1,
-1, 32, 33, 34, 35, 36, 37, 38, 39, -1,
-1, 40, 41, 42, 43, 44, 45, 46, 47, -1,
-1, 48, 49, 50, 51, 52, 53, 54, 55, -1,
-1, 56, 57, 58, 59, 60, 61, 62, 63, -1,
-1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
-1, -1, -1, -1, -1, -1, -1, -1, -1, -1
```

Figure 2: 10x12 Board.

Figure 3 below shows the mailbox numbers (indices) of the inner 8x8 board (the actual chessboard) that is embedded onto the 10x12 board. Credit for the arrays goes to Tom Kerrigan.

```
21, 22, 23, 24, 25, 26, 27, 28,
31, 32, 33, 34, 35, 36, 37, 38,
41, 42, 43, 44, 45, 46, 47, 48,
51, 52, 53, 54, 55, 56, 57, 58,
61, 62, 63, 64, 65, 66, 67, 68,
71, 72, 73, 74, 75, 76, 77, 78,
81, 82, 83, 84, 85, 86, 87, 88,
91, 92, 93, 94, 95, 96, 97, 98
```

Figure 3: 8x8 Board.

These arrays are used by the TSCP chess engine to figure out which pieces can go where in the following way. Let's say that there is a rook on square a4 (value of 32) and we want to know if it can move one square to the left. If we subtract 1, we get a value of 31 which corresponds to square h5. We know that a rook can't make this move, but how does the program know this by looking at the arrays? Instead of using the value associated with a4 (31), we use the mailbox number (index) of a4 (61), as shown in Figure 3. Then when we subtract 1 from that, we get the mailbox number 60. Going back to Figure 2, we see that the value stored on the mailbox number 60 is -1. Thus, we know that this move is impossible. A similar process can be repeated for each piece. If the sentinel value of -1 is referenced, then we know the move is out of bounds and is an illegal move.

Each type of piece has a different set of move target squares, and so the 10x12 array is modified for each one so that only the legal moves are known. This is done through offset move generation, where all of the values on this 10x12 array are offset by certain values depending on the type of piece which is being moved which allows for move target squares to be updated accordingly.

4.8.4 Piece Detection Comparison

After comparing the aforementioned possible piece detection systems, it was concluded that using a memory-based system, specifically a mailbox array-based system, would be the most effective solution. This type of system is already implemented by default in most chess engines, including the TSCP chess engine which was determined to be used for this project. The code didn't even need to be modified very much, because the TSCP already uses this piece detection scheme to determine if a move is legal and will automatically return the text "Illegal move" if a move cannot be made according the rules of chess.

Thus, the only extra work which needs to be done when using this system would be to modify the code to send a signal to the microcontroller as well as returning that text whenever an illegal move is triggered. This takes care of the issue of keeping track of where all the pieces are in the most efficient way.

While using Hall effect sensors or RFID were considered, these methods would take much more effort to implement than using the built-in system from the chess engine and likely wouldn't be as effective anyways. Thus, it was a no-brainer to use a memory-based piece detection system in this project.

4.9 Voice Capture and Recognition

For the voice-activated portion of the design to work properly, the right hardware and software needed to be chosen to most effectively capture and interpret voice commands. Capture and interpretation of voice commands was a necessary component in the functionality of the product. A major premise of the game is the ability for the players to use only their voices to move the pieces. This means the game must be able to give feedback to the players and prompt them on when to issue their commands without the players needing to physically interact with any part of the game.

At the start of each player's turn, the team contemplated on a set of speakers that would be connected to the chessboard to indicate which player's turn it is and prompt the player for voice commands, as well as inform the player of any current endgame conditions. A microphone was also required to capture the player's voice commands, which would be analyzed by the voice recognition software and translated into activating the piece movement system to perform the desired move. If the player attempts to make an illegal move or issues a command which is not understood by the software, then the speakers would indicate to the player what the issue was with that move and ask them to issue their command again.

Assuming the use of these hardware components, the speakers and the microphone, they would have to be able to fit comfortably within the design of the chess board. Not only that, both the microphone and the speakers would have to easily connect to the microcontroller. Ideally, the power consumption and required voltage to operate the audio hardware would have to be provided by the microcontroller itself. Otherwise, power would need to be re-routed from the wall outlet and pass through an AC adapter and voltage regulator so that the audio hardware would be properly powered. Besides this, the microcontroller runs the chess engine and that checks whose turn it is, move legality, endgame conditions, etc. The voice recognition software was expected to be run off the microcontroller if memory isn't an issue and having the microphone and the speakers connected directly to the microcontroller would reduce processing time and have a faster response. The other option was to run the software on a desktop or laptop and the data will be sent to the microcontroller to be analyzed by the chess engine, furthering costs and space within the chess board with using the computer's speakers and microphone.

4.9.1 Voice Recognition

A major focus of this project was to create a product which allows users to play a game of chess using only voice commands to control their pieces. This was implemented through the use of a voice recognition software which converts the user's voice commands into a set of text instructions which can be inputted to the piece movement system to produce the appropriate movements. Research was done on the different open-source voice recognition software packages available until one was chosen that appeared easier to use and could've been implemented into our project.

4.9.1.1 PocketSphinx

CMU Sphinx is an open-source voice recognition software created by students at Carnegie Mellon University that was considered to be used for this project. CMU Sphinx has a version called PocketSphinx that is designed to be used in embedded systems and can support Python or C. However, it could be used on a desktop computer as well. This version was chosen because it uses less resources and was easier to integrate with the rest of the project's software. PocketSphinx requires its source code to be downloaded and compiled on the operating system. It may be preferable to run PocketSphinx on a desktop computer rather than a microcontroller because most typical microcontrollers don't have enough memory to run PocketSphinx and all the other software concurrently without a significant computational delay. Running PocketSphinx on a computer would also make it easier to debug and test modifications to the code.

One modification that needed to be done was to customize PocketSphinx's word library to only contain the words that are needed to play chess; for example, "A2 to A4" needs to be interpreted as the piece on tile A2 moving to tile A4. If the original word library containing the entire English dictionary was used, the accuracy would be much lower because there were too many possible words to be interpreted. Additionally, two players would be issuing voice commands during the same match, so the software would need to be able to support multiple voices as well as, to be accurate for a voice it has never heard before. Thus, the word library would've had to be modified until there was a high level of accuracy when issuing chess-related commands.

Another modification that was thought to be done to the software was the noise cancellation algorithm. CMU Sphinx provides their own noise cancellation algorithm that can be switched on; however, this would need to be tested for the project's application and modified, if necessary. If the team were to use a microphone, the type used, the default settings may not allow for a sufficient level of accuracy. In that case, a new noise cancellation algorithm would need to be developed.

A new program would need to be written so that it would be able to analyze the text outputted from the PocketSphinx software, listening to player giving commands. This text would be translated into a format that allows this new program to check if the move is legal in terms of the chess rules. If it is a legal move, then this program would send the source and destination tiles to the stepper motors, and the XY-plotter would then move the desired piece from the source tile to the destination tile. This piece of software would be required to be custom-built and serve as the interface between the voice recognition software and the physical parts of the XY-plotter.

4.9.1.2 Amazon's Alexa Skills Kit (ASK)

The Alexa Skills Kit (ASK) SDKs for Node.js, Java, and Python are software development tools and libraries that give developers programmatic access to Alexa features. The ASK SDKs can help make it easier to build Alexa “skills” by enabling the developer to spend more time on implementing features and less time writing boilerplate code. “Skills” are like apps for Alexa, enabling the user to interact with all sorts of everyday products and content using their voice.

The advantage as a developer is that it’s possible to integrate Amazon’s voice recognition API into all sorts of everyday products, including the Smart Chess Board by using the ASK SDKs that Amazon provides. There are many tutorials out there on how to do this, including from Amazon themselves, although not any on how to create a smart chess board in particular.

Some of the drawbacks of Alexa is that it’s only voice-activated by vocally stating “Alexa” before it starts recording. Unless this could be altered to be activated by a pushbutton or some other switch that the player controls, then this could be problematic. Another problem that other developers who attempted to create a chess game using Alexa is the time before it stops listening. Alexa includes a security feature which closes the app after 8 seconds of listening so that a third party could not hack into the device and be constantly listening.

4.9.1.3 BitVoicer

BitVoicer is a speech recognition application that enables simple devices with low processing power to become voice operated. It was created specifically to be used with Arduino devices; however, it can be used with any programmable microcontroller that has a serial or TCP/IP communication interface. BitVoicer works by using the PC’s processing power to analyze audio streams, identify which sentences are present in these streams, and send commands to the microcontroller connected to it. BitVoicer currently can be used in over 25 different languages and dialects, including US English, and can support an unlimited number of commands and sentences. The BitVoicer application consists of an all-in-one graphical interface where the user can design Voice Schemas, set up commands, follow the communication activity, and follow the speech recognition status.

The Voice Schema is the structure upon which the BitVoicer works to bind a specific command to a sentence recognized by the speech recognition engine. These sentences are manually typed in by the developer and can be customized to their specific needs. The main features of the Voice Schema are the creation of all possible permutations, called anagrams, of items for a given sentence and the mapping of each anagram to a corresponding command. A sentence is made up of a sequence of items, or words, which are combined to form the sentence. Table 16 below represents an example of a sentence model. Some items have a single option, while others have multiple.

Table 16: Example of Sentence Model.

Single Item	Option Item	Single Item	Option Item	Single Item
turn	on	the	Red	LED
	off		Blue	
			Green	

Table 17 below shows all of the possible anagrams, also known as permutations, that can be created from the sentence model shown in Table 16. Each of these anagrams are associated with a data type and a command that is to be communicated to the microcontroller interfacing with BitVoicer.

Table 17: Possible Anagrams of Sentence Model.

Anagrams				
turn	on	the	Red	LED
turn	on	the	Blue	LED
turn	on	the	Green	LED
turn	off	the	Red	LED
turn	off	the	Blue	LED
turn	off	the	Green	LED

BitVoicer has the option to use an activation word, which is a word or set of words that must be said before the sentence itself. This option can be turned on or off. If it's turned off, the voice recognition only activates when the "Start" button on the BitVoicer application is pressed and stops when the "Stop" button is pressed. If the option is turned on, the voice recognition will start upon the activation word being said and stop after the activated period expires. The activated period is set by the user and is entered in seconds. The activation word option can be used to personalize a Voice Schema by assigning different activation words to different Voice Schemas or to avoid false positives when operating in a noisy testing environment. Below in Figure 4, a sample of the Bit Voicer application can be seen.

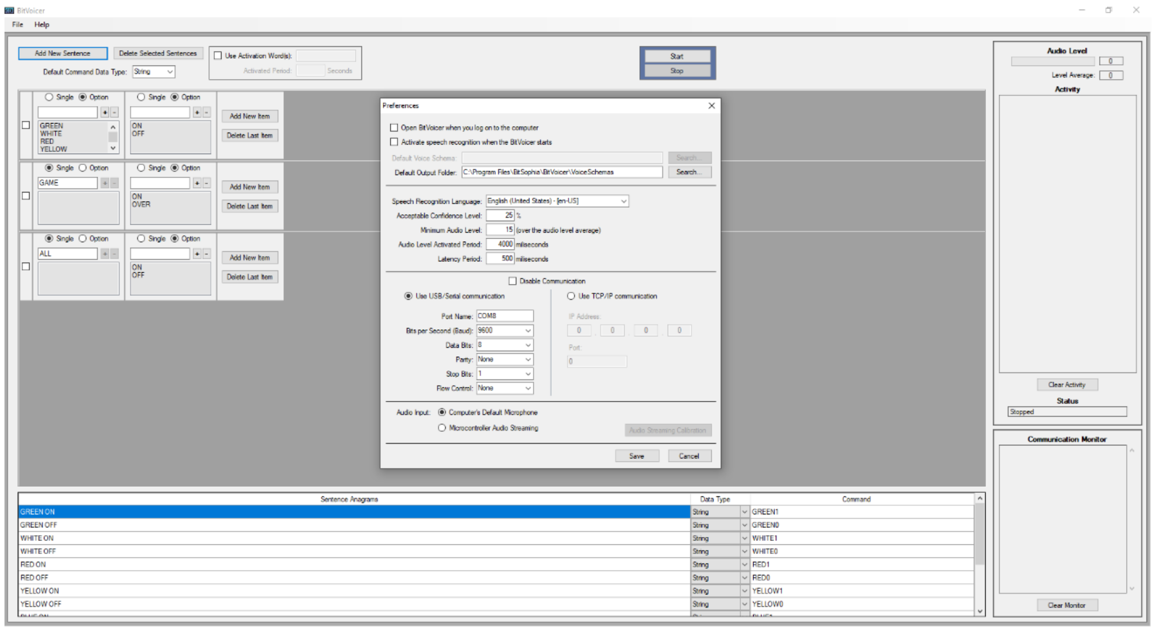


Figure 4: BitVoicer Application Sample.

BitVoicer could communicate with the microcontroller over serial (UART) ports or TCP/IP (Ethernet) ports. Whichever option was used, the data exchanged between BitVoicer and the microcontroller must be wrapped in a specific protocol called the BitVoicer Protocol. This protocol must be used in both communication directions: from BitVoicer to the microcontroller and from the microcontroller to BitVoicer. Essentially, all exchanged data is a sequence of bytes whose length will vary depending on the data type and the amount of data exchanged. The specifics of the BitVoicer protocol are outlined in the BitVoicer User's Manual. The BitVoicer installation packages includes an open source library that can be used as a reference. This library can be copied to the installation folder of the Arduino IDE to easily interface with Arduino devices.

4.9.1.4 Comparison and Justification for Selection

There are different pros and cons to using either PocketSphinx, Alexa Skills Kit, or BitVoicer but in the end BitVoicer was determined to be the most desirable solution. Both PocketSphinx and the Alexa Skills Kit are open source whereas BitVoicer costs approximately $5 to acquire the product key. The benefits of using PocketSphinx would be that the word library/vocabulary can be modified and trimmed down until the only words or phrases it recognizes are the ones that are used to play chess. This would be a much harder process for the Alexa Skills Kit because it has a built-in word library. However, BitVoicer is much more effective than both of them because there is no word library; the developer can add all of the chess-related terminology directly and make it so that no other words are included in the Voice Schema and the benefits of using it make the cost worth it. Each anagram can be assigned to a particular chess move and they can all be hardcoded to be done by a single command or the moves can be performed by a combination of commands. Either way, BitVoicer allows for much greater customizability than other solutions and is therefore the best choice.

A major problem with Alexa Skills Kit is that it is an example of a speaker-dependent voice recognition software, meaning that each user needs to "calibrate" their voice to the software by speaking into it for a period of time until it learns that user's voice and can accurately interpret commands. This would be a problem if it were to be used for the Smart Chess Board in the event that two human players are playing against each other since they obviously have different voices. It may be the case that the accuracy could be higher for one player than it is for the other. Since PocketSphinx is a speaker-independent software so it doesn't need to be calibrated to a user's unique voice. However, PocketSphinx needs to be provided a testing set consisting of common sounds so that it can compare inputted audio streams to them. A custom testing set could be created which represents a sufficient sample of diction, both acoustically and in terms of the language. The overall accuracy may still be lower than using an Alexa skill that has listened to thousands of words from a single user. BitVoicer is once again better than either of the other two options because not only is it a speaker-independent software but it doesn't require a custom testing set to work properly because one is already included in the program. The chess-related terminology is added to the Voice Schema and BitVoicer compares the inputted audio stream to only the collection of items included in the Voice Schema and chooses the one with the highest confidence level.

In addition to this, PocketSphinx is specifically designed to be portable and thus can easily be implemented on an embedded processor, and there are plenty of tutorials available on how to do so. Alexa is also portable, since it is meant to be implemented on an Amazon Echo or similar product, however it's not as easy to implement on a custom-built embedded processor such as the one to be used in this project as PocketSphinx is. BitVoicer gives the developer different options; it can be run on a PC and communicate with a microcontroller using either serial or TCP/IP communications, or it can run off a microcontroller and connect with the BitVoicer server using TCP/IP. It was originally designed to be used specifically with Arduino boards, so it is, by nature, lightweight and portable.

The major problem with Alexa, is the fact that Alexa needs to be voice-activated and only continues listening for a maximum of 16 seconds due to a built-in security feature. This would cause a major annoyance to the players since they would need to say "Alexa…" to activate the microphone before every turn. If they take longer than 16 seconds to issue their command then the microphone would shut off and either force the player to repeat the process or potentially misinterpret the command and make the wrong move on the chessboard, defeating the entire purpose of the product. Neither BitVoicer nor PocketSphinx have this issue. Both software packages have the option to be activated by a button (tactile pushbutton on the embedded device for PocketSphinx and a button on the BitVoicer PC application for BitVoicer) or to be voice-activated similar to Alexa. The advantage of BitVoicer is that the activation word can be set by the developer and the activated period can be customized and made longer than 16 seconds. For all of these reasons, BitVoicer was determined to be the best solution for the Smart Chess Board project.

4.9.2 Speakers

To add on to the voice-activated software and system and to make this game of chess more fun, speakers were to be added to the board. The speakers, initially, was going to be used for the board to state whose turn it is and ask what chess piece to move and where to, as well as, sound effects for when the game is starting, when a piece is killed off or promoted, and when the game is over. There are many types of speakers that could’ve been used for this design, but because the chess board has features that requires the speaker to be loud and clear enough for an output range to be heard by the two players, small enough to fit inside the chess board, cost efficient, and input/output capability adaptable to the hardware and software, the two main types of speakers that are being considered are portable speakers and computer speakers.

The option of speakers could’ve been housed either within the outer case of the chessboard or outside the case, as long as the players can clearly understand any prompts from the speakers. To further improve this feature, a set of speakers with a knob to control volume was desired, as this would allow for comfortable use in different environments. The speakers should be of high enough quality that commands can be clearly and easily understood given an appropriate volume. The cost would need to be balanced with quality to choose speakers which are good enough, but of which are not excessively expensive. For ease and simplicity, there was a great chance that speakers powered by a USB port could’ve been used. Some of the speakers researched and contemplated included computer speakers and portable speakers but because of cost restrictions, the different specifications had to be compared and the options of speakers are compared below.

MakerHawk Mini Speaker from Amazon

- Dimensions: 31mm (L), 28mm (W), 15mm thick
- Resistance: 8 Ohms
- Power: 3 Watts
- Full band/range frequency
- Frequency response: 500-20K (kHz)
- Sensitivity: 86(dB/W)

Adafruit Mini Metal Speaker with Wires

- Dimensions: 28mm/1.1’’(D), 4.55mm (height)
- Resonance frequency: 680 ±20%Hz at 1V
- Rated impedance: ~600-10KHz
- Rated input power: 0.25W
- Maximum input power: 0.5W
- 8 Ohm impedance

Skycraft Full Range Speaker

- Micro speaker mounted on ABS Plastic Cube Case
- Dimensions: 1.22’’(L) x 1.22’’(W) x 0.91’’(D)
- Wire length: 3 inches with a 0.1’’ pitch micro connector
- Single mounting hole 0.05’’ Diameter
- Rubber Edge Aluminized Mylar Cone with Copper Coil
- Frequency Range: 200Hz-20KHz
- Power: 2 watts RMS
- Rated: 8 Ohm Coil

4.9.3 Microphone

A microphone was required to provide an input to the voice recognition software so that the board can listen to the player’s commands and act accordingly. The type of microphone should ideally be optimized to listen to a human voice and provide a clear enough voice sample so that the voice commands can be analyzed with a high degree of accuracy.

The typical frequency range for a human voice that is used in telephony ranges from approximately 300Hz to 3400Hz. However, the fundamental frequency, or the lowest frequency present in the signal, can range from 85Hz to 180Hz for an adult male and 165Hz to 255Hz for an adult female. Therefore, the microphone that was selected needed to be carefully selected so it can maximize the volume of signals within at least the typical range for a human voice and attenuates any frequencies outsides of this range. That way the sound of the chess player’s voice commands can be converted to an electrical signal with minimal noise and thus be more accurately interpreted by the voice recognition software. Even though the voice recognition software has its own built-in noise cancellation algorithm, choosing the appropriate microphone and setting it up correctly could improve the functionality of the final product significantly by ensuring that the player’s commands are interpreted quickly and correctly on the first try.

Initially, the microphone option would be housed outside of the plywood and plexiglass casing of the chess board so that the player has access to it. The best type of microphone would be one that the players can wear on their head so that it can be as close as possible to their mouth and can clearly capture voice samples. A more sensitive microphone that would be placed further away could be used, but that would provide a lot more noise in the voice sample than a wearable headset would. Even though a noise cancellation algorithm is included the voice recognition software to help improve accuracy, it would be best to minimize noise from the outset by choosing the appropriate microphone type and setup.

4.10 Chess Engines and Computational Algorithms

A chess engine was required to interpret the commands that the voice recognition software outputs. In computer chess, a chess engine is a program that analyzes chess positions and comes up with a list of possible moves then executes the one it considers to be the strongest. A typical chess engine consists of a back end with a command-line interface. Some chess engines are used with a front end with a graphical user interface that allows human players to interact with them. There are myriad open-source chess engines available that could be used for this project; new ones are being developed constantly. All chess engines are made up of two major components: position evaluation and searching for the next move and choosing the best one. These two major components are implemented by a set of algorithms that are pretty similar across all chess engines.

The first major component of a chess engine, position evaluation, consists of a piece of code that evaluates any position and assigns it a numerical value. A position in this case refers to the positions of all of the chess pieces at any given moment, a “snapshot” of the board at that time. Most of the time, the code will be written so that a positive value corresponds to the white pieces holding an advantageous position, a negative value corresponds to the black pieces holding an advantageous position, and a value of zero means that the position is equal.

This numerical value is calculated by first assigning a value to each piece on the board based on their relative strength. Typically, pawns will be assigned a weight of 10, knights and bishops 30, rooks 50, and queens 90, with positive values for white pieces and negative values for black pieces. Then, a set of heuristic rules are applied which modify the evaluation of the position. These rules are based on basic rules of thumb for chess; for example, having doubled pawns (two pawns on the same file, or column) is usually considered bad, whereas castling is usually considered good. These heuristic rules are usually hard coded into the program and are developed by chess grandmasters or by referencing another chess engine. The more of these rules that are included, the stronger the chess engine will be.

The second major component involved in developing a chess engine is a piece of code which searches for the next move and chooses the best one. According to an estimate calculated by mathematician Claude Shannon, there are about 10^{120} possible chess games and roughly 10^{43} possible positions. For comparison, the number of atoms in the observable universe is estimated to be roughly 10^{80}. Therefore, using brute force calculation, where the computer analyzes every possible move and position before every turn, is not a feasible way to compute the best next move due to the computational power required. All engines generate a tree of moves where the root is the current position, and for each legal move there is a branch from the root to all the new positions resulting from a single move. For each of these branches, new branches are added for all legal moves. This keeps repeating and the number of branches increases exponentially as the number of future moves to be considered increases. This search tree is created using a minimax algorithm and refined using a method called alpha-beta pruning.

4.10.1 Minimax

In game theory, minimax is a decision rule used to *min*imize the loss involved when the opponent selects the strategy that gives *max*imum loss, hence the name. In other words, minimax is a process which aims to minimize the impact of the worst-case scenario. It can be used both when the players move alternatively or when the players move simultaneously. Minimax was originally formulated for two-player zero-sum games. A zero-sum game is a game where each participant's gain or loss of utility is exactly balanced by the losses or gains of the utility of the other participants. An example of a two-player zero-sum game is chess.

The minimax algorithm works by exploring the recursive tree of all possible moves to a given depth and evaluating the ending "leaves" of the tree. After this tree is evaluated, the smallest or largest value of the child to the parent node (depending on where it's white's move or black's move) is returned. Below is a visualization of how this process works.

In the example from Figure 5 below, the best move for white is b2-c3 because it guarantees that the worst possible position will be -50. If white moves b2-c1, then the worst possible position would be -80 (if black makes the best possible move). The minimax algorithm only takes into account the worst possible position for each move because it assumes the other player will always make the best possible move for themselves.

The minimax algorithm becomes more effective as the depth of the search tree increases. However, due to the large number of possible moves and positions in a game like chess, the number of leaves on the search tree increases exponentially and thus there is a limit to how deep the minimax algorithm can go without compromising computational speed.

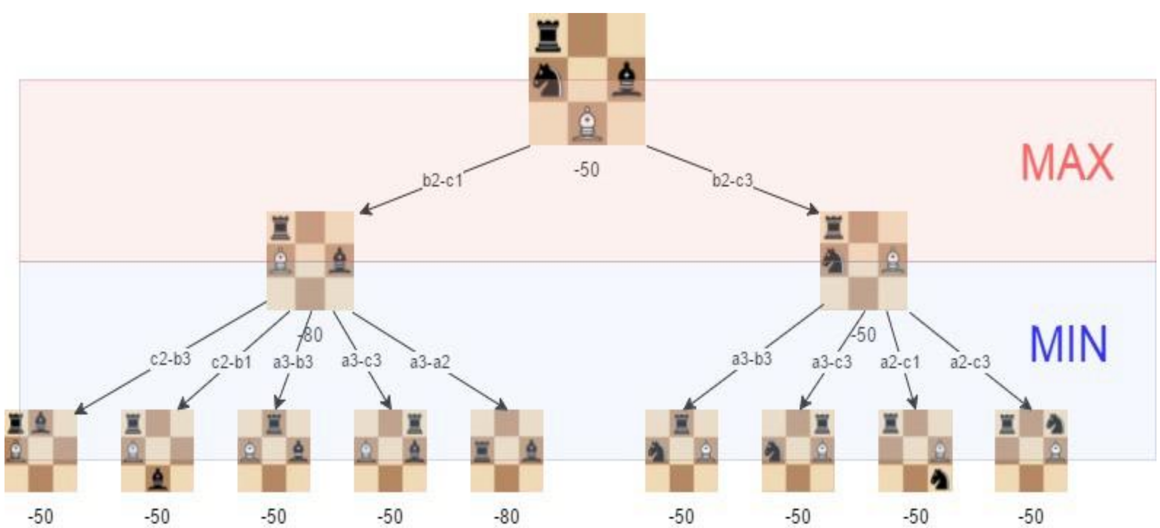


Figure 5: Visual example of Minimax algorithm. Used with permission from Lauri Hartikka.

4.10.2 Alpha-Beta Pruning

Alpha-beta pruning is an optimization method to the minimax algorithm which allows for some branches in the search tree to be disregarded. Since the computer no longer needs to analyze every possibility, it can evaluate the search tree at a much greater depth while using the same resources. The way this works is that if a move leads to a worse situation than a previously discovered move, then that branch of the search tree stops being evaluated. Below is a visual representation of the alpha-beta pruning optimization method applied to the same minimax example in the previous section.

As shown in the example from Figure 6 below, use of alpha-beta pruning does not influence the overall outcome of the minimax algorithm, but it does make the computation much faster. By using alpha-beta pruning to perform a search with a depth of 4, the number of different positions that need to be evaluated can be reduced by an order of magnitude. As the depth increases, the improvement in performance becomes more drastic.

These methods will not generate a perfect chess engine, but it will ensure that a simple one that avoids making stupid mistakes can be made. The average modern chess engine can evaluate potential positions up to a tree depth ranging from 16 to 18. To create such a chess engine would require the use of more sophisticated methods, but for the purposes of this project using this simple framework should provide a good enough chess engine for adequate functionality.

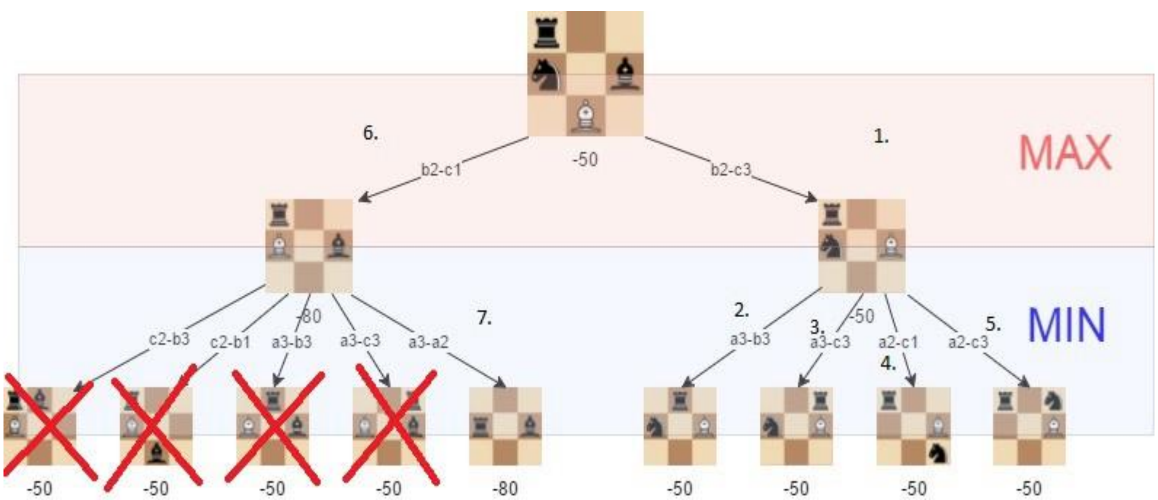


Figure 6: Visual ex. of alpha-beta pruning method. Used with permission from Lauri Hartikka.

4.10.3 Comparison of Different Chess Engines

Given the vast number of open-source engines that are available, different ones had to be compared in order to find the one that provided the most effective solution for the requirements in this project. The chess engines considered were Tom Kerrigan's Simple Chess Program (TSCP), Micro-Max, and Stockfish.

Several key factors were considered in choosing a chess engine to proceed with. The first was robustness, which is the ability for a computer system to cope with errors during execution. This includes accuracy and the ability of the chess engine to deal with mistakes in data entry. A robust system would not experience any consequences if data is improperly entered. The second key factor was resource usage. The chess engine chosen should be able to use the least amount of hardware resources to run. Ideally, it would be able to run completely off of the ATmega2560 microcontroller, but if no chess engine lean enough could've been found, it would run efficiently off of a team member's laptop. The third factor was adaptability. A chess engine must be modifiable so that it could be incorporated into the rest of the software used in the Smart Chess Board so that the features can be added or removed easily as necessary.

4.10.3.1 Tom Kerrigan's Simple Chess Program (TSCP)

TSCP is a small open-source tutorial chess engine, i.e., it's designed to teach people how chess engines work. Among the options considered, TSCP is very robust. To test this, it was run on a laptop and different moves were typed into the command prompt to test failure tolerance. If any data was entered which was not a recognized command or was not in the proper notation, the program simply returns "Illegal move." This would work well for the Smart Chess Board, since it would be able to check if a move is illegal or not. Despite the name, TSCP is relatively resource-intensive compared to other chess engines that were considered. It contains over 2,000 lines of code, uses almost 2MB of memory (the ATmega2560 only has 256KB) and requires a little under 64KB of RAM (the ATmega2560 has 8KB of RAM). However, TSCP is quite adaptable, as the source code is written in C, which the team is very familiar with, and is well commented. It also allows for the game state to be displayed at any point, which is useful for piece detection.

4.10.3.2 Micro-Max

Micro-Max (μ-Max) is a minimalist open-source chess engine. Micro-Max is not as robust as the other options when player-versus-computer is tested, however for the purpose of the Smart Chess Board project only the player-versus-player capability was going to be needed. For this case, it does work accurately, however when an invalid input is entered the program will not explicitly say "illegal move" but rather it will just ignore the move and behave as if the player has forfeited their turn. Micro-Max is a very compact program, only consisting 133 lines of C code and under 2000 characters. Even though there are few lines of code, the way in which memory is utilized is highly inefficient as a result, and so, Micro-Max uses about 200MB of memory. In terms of adaptability, Micro-Max is not as useful as the TSCP because for the Smart Chess Board, having a player forfeit their move was undesirable and it would've been much easier to have the program directly print out "illegal move" than it would be to write extra code to detect that an illegal move has been made based on the way Micro-max behaves. Also, because Micro-

Max is such a compact program, it would be harder to modify since each line is crucial to the program running properly. If anything was to be edited, it could have detrimental effects on the functionality of the program.

4.10.3.3 Stockfish

Stockfish is a powerful open-source chess engine, known for its strength and ability to beat most chess grandmasters easily. However, for this project only player-versus-player capability was needed so playing against the computer was an extra feature that wasn't necessary. Stockfish is also a robust engine comparable to the other two engines that were considered, as well as the most resource intensive. Since it is a rather large and complex program, it uses over 12MB of RAM and over 138MB of memory. This was much more than the other two programs and means that it cannot be run on the ATmega2560 chip at all and must be run on a PC. Even so, the performance would be slower than TSCP or Micro-Max running on a PC, although this difference may not be perceptible compared to the overall latency of the Smart Chess Board. In terms of adaptability, Stockfish is designed to have customizable rule sets which the developer can easily edit using .bin files. This feature is nice, but it was not needed for the Smart Chess Board since only the general chess rules were needed to be followed; specialized strategies and openings are only utilized for player-versus-computer games. When compared to the other two chess engines considered in this section as well as most other chess engines out there, Stockfish has a huge amount of documentation and tutorials available. Due to its strength, Stockfish is one of the most popular chess engines, so the documentation is regularly updated, and many developers use it in their projects compared to other engines.

4.10.3.4 Conclusions

Based on analyzing all of the different chess engines, it was decided that using Tom Kerrigan's Simple Chess Program (TSCP) would be the best solution. Tom Kerrigan was contacted, and he explicitly provided permission to use his software for this project. TSCP was ideal because it returns the phrase "Illegal move" for every type of illegal move that is to be made, from improper data entry to moves that are outside of the chess ruleset. This could easily be communicated to the microcontroller which would notify the player that an illegal move was made and prompt them to make another move. TSCP automatically would allow for a player that makes an illegal move to make their move again. The other two chess engines would skip the player's turn if they entered data improperly or made an illegal move, which would be annoying in case the player made a mistake or the voice recognition were to make a mistake in interpreting the player. Also, TSCP included an option to display the game-state on Winboard, which is an open-source graphical user interface chessboard for Windows. This graphical interface could be displayed on an LCD attached to the Smart Chess Board so that the player could keep track of the piece positions that the software is trying to interpret. This way, in case the XY-plotter made a mistake, or a player manually knocks one of the pieces off the board then they could manually put the pieces back in its proper location by simply checking and looking at the LCD.

4.11 LED

Adding some lights to the design would help make the chess board livelier, more entertaining and aesthetically more appealing. Originally, the idea for LEDs would also have been used as a learning tool where the player would select a chess piece and possible locations of where the piece could go would light up, but the more research that was done, the more complications came into play. The leading problem was encountered when the team questioned where the LEDs would go. The difference in size of the chess pieces and board became even more questionable because the LEDs would either have to be implanted into the chess piece with the magnet and somehow receive power to light it, or if the LEDs were to be under the playing board, issues of interference with the magnetic system and motor control would cause more difficulty. Not to mention, cost of the design would increase because the overall size of the project would have to be bigger.

There were many options for what kinds of lights to use considering size, type, brightness, color temperature, power consumption, and many more parameters. The two main types of lights that were being explored and researched for the specific design in the Smart Chess Board were individual LEDs and RGB LED strips. The primary specifications that were investigated in deciding the best option were cost, size and power requirement.

4.11.1 Individual LEDs

Individual LEDs were beneficial in that they're inexpensive, having one input and one output makes them an easy option to buy and use, and they're accessible at almost any hardware store. However, considering the design that they're being used for, single LEDs would require an extensive amount of tedious organization and patience to wire and connect each one. They also output only the single color that they're bought in, limiting the purpose of increasing the chess board's aesthetics with using the LEDs.

4.11.2 RGB LED Strip

Another option for LEDs were the RGB LED strips. Every LED chip in an RGB LED strip displays a code, which just states the size of the LED chip and brightness. If the codes are to be looked up, they'll have all of information found on the strip. These strips have many kinds of different color options, power requirements and brightness output. Their design made it very convenient to use, made with flexible material to be slightly bent without damage, having had the ability to be cut to a certain length, if needed, and every LED are discreetly wired and connected through the strip so that the end wires just needs to be connected to the microcontroller. There are many types of RGB LED strips available and the two brands that were analyzed were ColorBright and Adafruit NeoPixel.

4.11.2.1 ColorBright RGB LED Strip

ColorBright was one of the brands that were looked into. They're color changing LED chips, fixed on flexible material of printed circuit board. They also come with a remote control with a few different settings and options to adjust the brightness, set to flashing mode, and different color outputs. Not to mention, they're affordable. Some of the major specifications that was found to be beneficial was the compact 10mm width, the choice to dim and control the color, the option of a 12V or 24V DC input, thick double layer of copper PCB for thermal management. They're used in many applications mostly for aesthetics and lighting in general.

4.11.2.2 Adafruit NeoPixel RGB LED Strip

Adafruit NeoPixel Digital RGB LED Strip was another type of LED strip that were investigated. They're affordable made within a flexible material, much like ColorBright, making them easy to adapt to different settings. Again, much like ColorBright, these strips with its casing, is a width of 12.5mm and 10mm without. Something about these LED strips that really stood out was the chip that's integrated in the strip. They use a single input pin, output pin, and data pin, controllable by a microcontroller with a 100nS, highly repeatable timing precision, making them very cooperative. Just like the ColorBright RGB LED strip, they're affordable and could be bought in various lengths and can be cut if they're too long.

4.12 Chess Board Housing

A very important aspect of the design was the chess board's housing. It holds together the entirety of the project, from electrical to the mechanical components. A great deal of research went into this section because it had to be made to withstand different environments, hold and protect every element of the project, it had to be the right size to avoid extra costs, to be able to fit everything in, and to be as portable as possible.

4.12.1 Building Material

There were many elements to consider while choosing different components to build the actual chess board. The base and surrounding of the board were made of wood to create a stronger structure for the design, whereas the surface of the board was made of Plexiglass to accommodate with the movement of the magnets and chess pieces. When the list of components that were required for the project was made, it was obvious that the framework along with all of the hardware were to be carefully selected to provide the most effective ensemble.

4.12.1.1 Plywood

With many options for plywood available for purchase, a decision had to be made for which type of plywood was to be used for constructing the chess board. With the requirements for the project in mind, the final decision of using sande plywood for the outer body was reached. Sande plywood is a type of wood used mainly for marine purposes. Due to the fact that this wood is used around and in water, sande plywood sheets are crafted especially to be water resistant while also maintaining a smooth and sleek finish. This sets sande far apart from other plywood combining the best of both interior and exterior use plywood. Interior use plywood has a smooth finish but without the water-resistant factor, it is prone to warping in moist environments. On the other hand, exterior use plywood, though being water resistant, has a very rough and unpleasant finish due the fact that it is hidden in its use of outdoor construction. With this chess board's outer body acting as a shell for all of the electronic components of the design as well as being seen and felt by the consumer of the product, a combination of these two features were needed in the building material. Therefore, sande plywood was chosen for the final production.

4.12.1.2 Clear Surface

A clear surface was needed for the top playing surface of the chess board. This surface could be made of any material that best fits the requirements of the product. When deciding on the material to use in this project, three different materials displayed themselves as valuable options each in different ways. These material choices were glass, plexiglass, and Lexan.

Glass is a material made from heating and melting sand at a temperature of around 3100°F. Glass is used in many different ways including panes for windows and bottles/containers. Glass as a material is very stiff and sturdy not allowing much flex at all. Glass is also prone to shattering and breaking when enough pressure is applied to the surface. Finally, the weight of glass highly exceeds that of the other options in clear material of the same thickness.

Plexiglass, also known as polymethyl methacrylate (PMMA), is a clear thermoplastic alternative to glass. Unlike glass, plexiglass is relatively shatterproof as well as being a much more lightweight choice in material. The stiffness and sturdiness of plexiglass varies with the thickness of the material. Thinner sheets of plexiglass may flex and bow with a small amount of applied pressure while thicker sheets will remain flat until a stronger force is applied.

Lexan is a clear polycarbonate thermoplastic alternative to glass. Like plexiglass, Lexan is a shatter resistant sheet of clear thermoplastic that is about half the weight of a glass counterpart. The stiffness and sturdiness of Lexan is close in nature to plexiglass, thinner sheets are less stiff while thicker sheets are stiffer with the application of pressure on the surface. While Lexan and plexiglass have this in common, Lexan will bend more rather than cracking like plexiglass.

4.12.1.3 Screws

Screws are a type of fastener used to hold two hard materials together. Screws are made of metal for strength and combine aspects from both bolts and nails. Like nails, screws have a pointed tip to help penetrate the material being fastened by the screw. Like bolts, screws have threads around the diameter to help the screw grip the material for a stronger hold. Screws are available in a variety of sizes, including diameter that can vary from #0 through #20 and length that can vary from 1/4" and above. Many different types of screws exist for a multitude of purposes, each having a specific design and use. The most common types of screws include wood screws, concrete screws, and drywall screws.

Wood screws are a type of screw most commonly used to fasten two pieces of wood together during construction. Wood screws usually contain a more aggressive thread pattern than other screws allowing for the screw to pierce and grip the wood creating a stronger and tighter hold of the material. Wood screws are most commonly made of steel and are sometimes coated for corrosion resistance when used outdoors. A common brand of drywall screws on the market is Everbilt.

Concrete screws are a type of screw most commonly used to fasten material, such as wood, to a concrete wall or structure. Concrete screws usually consist of a tooth like tip for penetrating the concrete and sharp threads to pierce and grip the concrete. Concrete screws are commonly made of steel since they are required to be strong enough to penetrate concrete. A common brand a concrete screw on the market is Tapcon, which come with a blue coating.

Drywall screws are a type of screw most commonly used to fasten drywall sheets to a wooden or metal frame during construction. There are two types of drywall screws, W-type and S-type. W-type screws are used for attaching drywall sheets to wooden frames while S-type screws are used to attach drywall sheets to metal frames. Unlike other types of screws, drywall screws have a deeper thread diameter to allow a stranger hold on the more fragile sheet rock. A common brand of drywall screws on the market is Grip Rite.

4.12.1.4 Adhesives and fillers

Wood Glue is a type of adhesive specific to the use in carpentry and woodworking. Wood glue is used to hold two pieces of wood together during the construction process. For the product to work, a certain amount of time, specific to the given product, is needed for the glue to cure fully (cure time). During that time the pieces of wood being held together by the wood glue must have pressure applied to them while allowing the glue to cure and creating a strong bond. Wood glue is often used in conjunction with screws for a stronger final product. Wood glue comes in a variety of different types. These types include polyvinyl acetate (PVA), cyanoacrylate (CA), epoxy based, polyurethane, and hide.

Polyvinyl acetate, also known as PVA, is the most popular type of wood adhesive on the market. Polyvinyl acetate is a thermoplastic synthetic polymer that is used mostly in a variety of glues. The benefit of using a PVA wood glue would be that the product is relatively inexpensive in price point while still maintaining a strong and reliable bond on wood as well as the products non-toxic nature. This deems the product safe and inexpensive for everyday use on wood products around the house. Common PVA wood glues on the market include Titebond II Premium and Elmer's wood glue.

Cyanoacrylate, also known as CA, is most commonly used in super glue. This glue creates a strong plastic-like bond between two pieces of hard material (wood in this case). The benefit of using a cyanoacrylate glue would be that the product has a fast-drying time along with the strength of the bond without the need for clamps. The downside to using a cyanoacrylate glue would be that the product begins curing immediately, causing a small window for application errors. Common CA wood glues on the market include Stick Fast instant CA glue and Gorilla super glue.

Epoxy-based wood glues usually come in two parts. These two parts include the resin and the hardener. Curing of the glue begins when these two products are combined. The benefit of using an epoxy-based wood glue would be the product's ability to fill large gaps while creating a strong bond. The downside of using an epoxy-based glue would be the long curing time in conjunction with necessary clamping. Common epoxy-based wood glues on the market today are Gorilla 2-part epoxy and J-B Weld clearweld epoxy adhesive.

Polyurethane glues are used to create a strong bond between a variety of different hard materials including plywood. When polyurethane glue dries, it expands allowing the product to fill gaps between the material being bonded together that may not have fit flush up against one another. The benefit of using polyurethane wood glue would be the product's fast drying time while filling gaps and creating a strong bond as well as the product's waterproof nature. The downside of using a polyurethane glue over the other options would be the product's much higher price point. A common polyurethane wood glue on the market is Gorilla wood glue.

Hide glue is a type of adhesive made from animal hides. Hide glue can either be in the form of a solid (most common) or a liquid (least common). This type of glue is not common for use in today's market but was much more prevalent in the past. One liquid hides wood glue still on the market today is Titebond liquid hide wood glue.

Wood Filler is a product used during the construction of anything involving wood. When plywood is used for construction, the material may have many imperfections that can affect the final look and finish of the desired product. In order to counteract these negative effects, a wood filler is used to fill the imperfections. These imperfections include cracks, gaps, and screw holes. After the filler is dry, the excess can be sanded away, and the surface can be painted or stained. The most common wood fillers on the market are Plastic wood, bondo, and Elmer's wood filler.

Silicone, also known as polysiloxane, is a product that can be used as both a sealant and adhesive in the form of caulk. Most silicone caulks dry as a rubber like material and form a strong watertight seal. The most common types of silicone sealants and adhesives are Loctite clear silicone, Permatex clear RTV silicone, and Gorilla clear silicone.

4.13 Chess Piece Manufacturing

In order to create a functioning chess set, a set of chess pieces were needed to accompany the chess board. With many options of chess pieces to choose from, it was decided that three options would be looked into in more depth. The chess piece options for this project include the following:

- Buying a set of wooden chess pieces
- Buying a set of plastic chess pieces
- 3D printing a set of chess pieces

4.13.1 Wooden Pieces

One type of chess piece set available for purchase were wooden based chess pieces. These chess pieces are constructed and carved out of wood to create the desired shape and size of each chess piece type. When purchasing wooden chess pieces for use in this project, the chess pieces offered were only available in the dimensions of those according to the chess standards. When implementing these chess pieces into the design of this project, modifications would have to be done to fit the needs of the final product. These modifications include adjusting the base diameter of the chess pieces as well as adding metal to the bottom of each piece so that the electromagnet can attach to the piece during movement. To accomplish this, a nail, screw, or something similar would need to be inserted into the bottom of each chess piece before implementation.

4.13.2 Plastic Pieces

Another type of chess piece set available for purchase were plastic based chess pieces. These chess pieces are constructed using a polymethyl methacrylate (PMMA) plastic. The pieces are shaped and sized using a mold to create the desired playing piece. Like the wooden chess pieces, when purchasing plastic chess pieces for use in this project, the chess pieces offered would only be available in the dimensions of those according to the chess standards. When implementing these chess pieces into the design of this project, modifications would need to be done to fit the needs of the final product. These modifications include adjusting the base diameter of the chess pieces as well as adding metal to the bottom of each piece so that the electromagnet can attach to the piece during movement. To accomplish this, a small metal cylinder would need to be inserted and glued into the bottom of each chess piece before implementation.

4.13.3 3D Printed Pieces

The final type of chess piece set available to use in this project were 3D printed chess pieces. These chess pieces are constructed using an abs plastic and a 3D printer. The pieces are shaped and sized using a CAD file and uploaded to the 3D printer to create the desired playing piece. When 3D printing chess pieces for use in this project, the design created for the chess pieces would match the desired needs for the project with no further modifications needed. This is because the user of the printer can set the dimensions to what is needed despite what the chess standards are.

5. Requirements Specifications

There are two main portions of specifications that are highlighted in this smart chess board project, software specifications and hardware specifications. The software section will focus solely on programming and how certain elements will be operated and hardware will clarify what physical components that was required and would be used. Whereas hardware parts, such as electrical and mechanical, would essentially be anything tangible, will be specified in that section.

5.1 Software Specifications

For the voice-activated magnetically powered chessboard to be successful, the following software specifications had to be met. The latency should not exceed 5 seconds from the time the player inputs their voice commands to when the pieces actually move. This includes all of the computations that was needed to be done, from the speech-to-text conversions to the movement of the mechanical system.

The system was expected to be able to recognize each type of chess piece separately and analyze whether a move is legal for that particular piece, as well as perform all possible moves, including capturing a piece, without having pieces come into contact with each other. All software was expected to run on a microcontroller. The speech-to-text software was to be able to support at least two different voices and be accurate enough to correctly interpret commands on the first try for most players. The board was also planned to speak to the player through a speaker, prompting them for instructions.

Once the game begins, the first player is to verbally state the piece and location to be made and the piece detection system will then come into play. To simply relocate a piece, the player just states one legal location to the another, for example, “b2 to b4.” The plotter will move to the location b2, the electromagnet is turned on, the piece is moved to b4 and the electromagnet is then turned off and waits for the other player's turn. If a piece is to be killed off, or "captured," the player is to state the location of the piece moving, say "capture," and then the destination, for example, "b2 capture b4." The plotter will go to b4, move that piece to the graveyard to clear the location, then go to b2, and then finally, move that piece to b4 location. The players just take turns stating their move, and it's a repetitive iteration of the procedure until the game is over. A quick step-by-step of how the game was approached is presented in the flowchart in Figure 7 below.

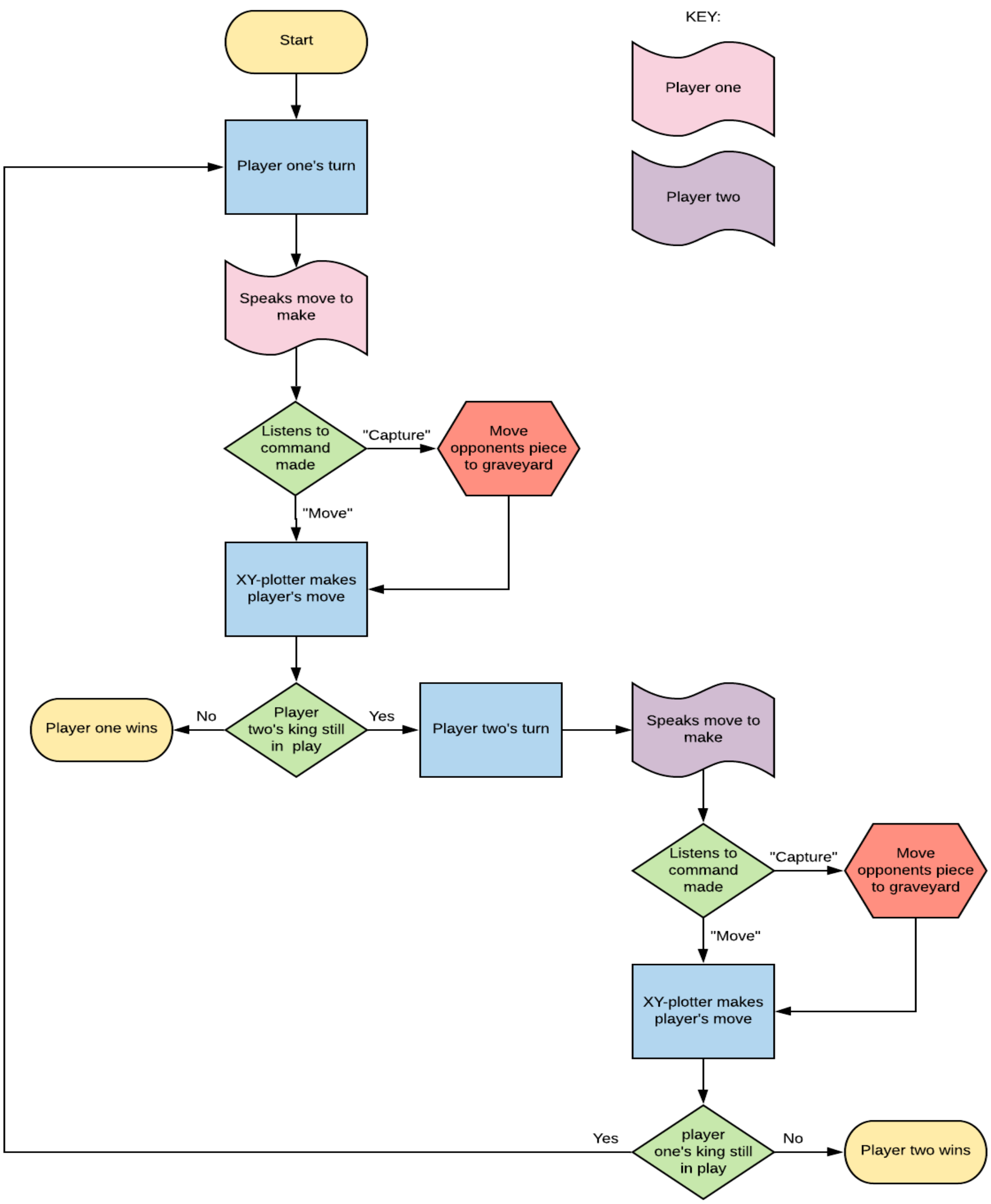


Figure 7: Overall Game Flowchart.

5.1.1 Motor Control System

Because the design required an abundant amount of precision to reduce interference, the best option was the stepper motor. The problem was that the team would need a motor control system in order to control the stepper motors with precision. The piece movement system already came with the two motor controllers and two stepper motors. This made it a lot easier for the team because since the system included the motors, all the components would obviously be compatible.

The Motor controller that came with the piece movement system was the A4988. The A4988 gives the options of a few step movements depending on the configuration. The step movements options were Full step (1/1), Half step (1/2), Quarter step (1/4), Eighth step (1/8), and Sixteenth step (1/16). Higher micro stepping would result in a smoother and quieter operation. The minimum step pulse duration for this motor controller was 1µs. A few important characteristics are listed below in Table 18 for the motor controller.

Table 18: Motor Controller Characteristics.

Characteristic	Symbol	Rating	Units
Load supply voltage	V_{BB}	35	V
Output current	I_{OUT}	±2	A
Logic Input voltage	V_{IN}	-0.3 to 5.5	V
Logic supply voltage	V_{DD}	-0.3 to 5.5	V
Motor output voltage	V_{OUT}	-2.0 to 37	V
Reference Voltage	V_{REF}	5.5	V
Sense Voltage	V_{SENSE}	-0.5 to 0.5	V
Operating ambient temperature	T_A	-20 to 85	℃

In Figure 8 below, the schematic of the typical application that the team used in order to control the motor controller. The motor controller has two different supply voltages, a 5V digital voltage from the microcontroller and the 12V from the wall outlet power supply. A 5.5 V was needed for V_{ref}, this created some difficulty because the microcontroller only supplies up to 5V.

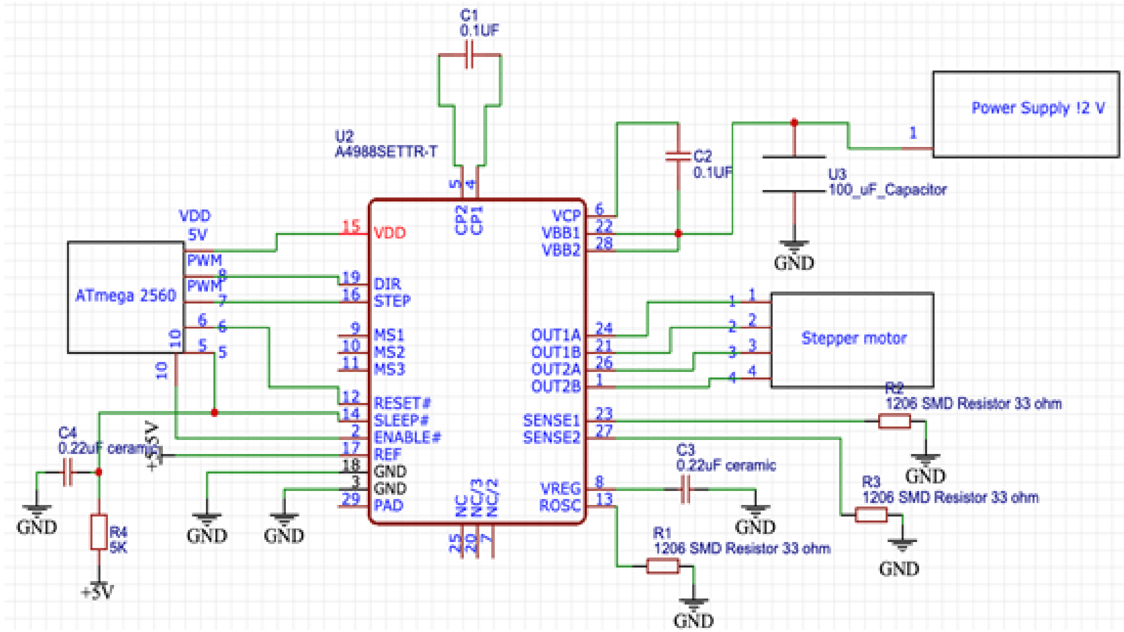


Figure 8: Motor Controller.

The microcontroller was used to control the digital logic while the power supply supplies the power to move the stepper motors. The direction (DIR) and the step (STEP) pins were required to be connected to pins that could provide PWM on the microcontroller in order to operate. The sleep and reset pins were very useful but can be hooked up together in order for them to be disregarded. The sleep pin was also used to minimize power consumption and to save power and the reset pin turns off all FET outputs, making all the STEP inputs ignored until the reset input is set to high. The most important pin would be the Enable pin, this pin controls the on and off pf the FET outputs. Another important feature mentioned was the Microstep select (MSx). The Microstep resolution is set by the voltage on the logic inputs, a truth table is shown below, in Table 19, in order to display the different configurations.

Table 19: Microstep Resolution Truth Table.

Microstep Resolution	MS1	MS2	MS3
Full Step	L	L	L
Half Step	H	L	L
Quarter Step	L	H	L
Eighth Step	H	H	L
Sixteenth Step	H	H	H

5.1.2 Voice Recognition

Voice or Speech recognition can be a computer software program or a hardware device with the ability to decode human voice. Voice recognition omits having to use a keyboard or press any buttons in general. This portion was executed on a computer with an automatic speech recognition (ASR) program. Depending on the ASR, it could require the user to train it in order to recognize the users' voice. Because the chess board required commands from multiple people to be recognized, was going to have to choose a speaker independent system. The speaker independent system would be able to recognize most users' voices with no training.

The voice recognition program that the team had decided to go with was BitVoicer. It's a speech recognition application that enables simple low processing power devices to become voice operated. It analyzes audio streams to identify sentences in these steams and then sends commands to the microcontroller connected to it. It supports many different languages and has the ability to store numerous commands and sentences. It also uses a USB/serial or TCP/IP communication interface. The benefit about using this software was that there are many projects that uses features similar to the smart chess board posted on the website, which can be referred to at any time.

5.1.3 Chess Engine

Due to time constraints and the complexity involved in creating a chess engine from scratch, it would be a much better solution to adapt an existing open-source chess engine to perform the tasks needed. Some of the factors which were to be considered in deciding which chess engine was to be use included frequency of errors, customizability, and computer resource consumption. A chess engine would be chosen while being mindful of its accuracy, without making errors as this could cause failures in the overall system that compound and would require the chess game to be started again from the beginning. It was intended to be able to recognize which player's turn it is, move legality, endgame conditions, and piece capturing with a high degree of accuracy. It was highly unlikely that a chess engine would be found that worked perfectly for the purpose of this project, so one had to be selected that would most likely be adaptable to this project. Thus, when looking for a chess engine to use, the ease of customizability was a big factor. It was also important to choose a chess engine that used the least computer resources possible. This ensured that the processing time would be minimized so that the player's moves could be executed as quickly as possible; also, the chess engine was expected to be running off of the microcontroller and so there is a limited memory it needs to work with. If the chess engine was to be too resource intensive, there was also the option to run it off of an external computer, which could've slowed down the game considerably.

5.1.4 Motor Control System

Since the XY-Plotter kit the team chose already came with its own stepper motors, as shown in Figure 9 below, the instructions of how to program the software was also included. The software package for the system is called the GRemoteFull package, it included the control program, control software and G-code examples. The software is explained step by step in the manual. It explains how to control the XY-Plotter with the keyboard and mouse, and it took quite some time to figure out how it worked and how to interface it with the voice recognition software.

Figure 9: Makeblock Stepper Motors.

5.1.5 Piece Movement System

This project required that the implemented XY-Plotter perform specific tasks commanded from the microcontroller with agility and precision while following the grid layout of the chess board. The set of instructions commanded were specific only to the application and dimensions of the chess board design and does not exactly match any other uses of XY-Plotters in today's applications. In order for the XY- Plotter to complete such tasks, a program was to be coded and implemented into the system to work with the specified chess board design requirements. This being stated, the XY-Plotter chosen was required to easily and effectively have a written program implemented to control the movement and speed of the device as well as understanding and following a desired grid layout as set in the program.

The XY-Plotter that was purchased for this project was designed to be used to hold a pencil and draw vector illustrations that were originally developed on a computer. This XY-Plotter is intended to be operated using the mDraw computer software, which was developed to be compatible with this specific plotter. mDraw supports the SVG filetype which is based on the open source software Inkscape. Inkscape is a professional quality vector graphics editing software which can be used to create or edit vector graphics including illustrations, diagrams, line arts, charts, logos and complex paintings.

5.1.6 Piece Detection System

The requirement for the piece detection system is that the game's computer (the microcontroller) must be able to analyze each move that the player commands and check whether this move is possible and whether an enemy piece is to be captured with this move. Therefore, a piece detection system needed to have the ability to detect which piece was selected when the player issues a command.

It was expected to analyze whether the move that was requested by the player was a legal move by comparing it to a chess ruleset, it needed to be able to detect whether or not another piece already occupied the tile that the selected piece was moving to, and which player that piece belong to, in order to capture it.

For this project, the piece detection system was expected to be implemented by using a memory-based software to keep track of the piece locations. At the beginning of each game, the players would have to manually set the pieces up in their starting locations. Every game of chess starts with the pieces set up in the same way so at this point (Turn 0) the software would always know where all the pieces are. This way the pieces do not need to be uniquely identified at any point in the game, since they will be recognized based on their starting positions at Turn 0. The piece locations could be stored using a data structure like an array or in different registers on the physical hardware. The array would be more resource efficient to implement as well as more elegant, but it would be harder to create, given the team members' skillsets.

After each successful move was to be made, the array would be updated to reflect the new position of the piece that was moved. This would be done for each move until the game ends. Once the data structures were set up, the rest of the program only needs to check a few different things; what type of move, whether the move was legal or not, was a piece being captured. This could've been done by writing different functions for each condition to be checked and activating the functions via series of "if" statements in the main program. All of this software would be written in C++ on the Arduino IDE to make it easy to implement and test on the ATmega2560 microcontroller that would serve as the brain of the Smart Chess Board.

5.1.7 LED Code

The LED portion of the project began with just LED strips, specifically, Adafruit NeoPixel Digital RGB LED Strips. Their compatibility with Arduino Mega and guidance and references provided on how to program the LEDs was one of the main reasons why this brand of lights were selected. With the team's minimal experience with programming, the website has a learning menu that provided a basic code and tutorial for them to work and with some manipulation and recoding, the LEDs will function as expected.

Arduino conveniently has the Adafruit NeoPixel library that can easily be installed, or they have the file ready to be downloaded on Adafruit's tutorial page. A sample code was also provided to get the LEDs started and to light up. With the datasheet, and instructions, the desired color of the LEDs could be operated and displayed as well as the change of brightness and speed, or even the pattern of how the output of the light was wanted. The website offers an abundant amount of information on the LEDs and how to program them.

As the team progressed, it became clear that the LED strips weren't going the be the only lights to be used. Individual LEDs were also decided upon. Their ease and simple two-pin design, connectivity and code made the addition of the lights an effortless decision. By using the function digitalWrite() with HIGH or LOW, the LED could simply be turned on or off.

5.1.8 LCD / Audio System

Selecting an LCD screen and audio system were some of the last features that was purchased. But because they're so commonly used, it was expected that the many references and instructions on how to program them would be helpful. There were quite a few modifications and updating to the provided codes so that the features would operate with the voice recognition system and the other components in the project as anticipated.

The team originally selected a generic 3.5-inch TFT LCD Touchscreen for the Arduino Mega2560. The screen was expected to display specific things such as which player's turn it was, in addition to, the desired position the user input. The seller of the LCD screen did not provide a datasheet or much information on the product, but lots of tutorials and instructions on how to implement a general LCD screen with the microcontroller could be found on. After programming and getting the LCD to work, it was determined that the LCD was defected and the Elegoo UNO R3 2.8-inch TFT Touchscreen was purchased and used. Unlike the first purchase, the seller provided plenty of information, resources, as well as sample codes to get started. The biggest issue was getting it to work with the voice software, BitVoicer.

Audio would complement the user's experience by creating special sound effects when certain things happen. For example, if a piece is taken down by the opponent, it would make the game more entertaining and fun of the sound of swords knocking against each other or even a sound effect that would be suitable to represent a victory could be heard. The team would like to implement this into the project, but some things to consider for the audio would be its link to the software, the file must be readable, and how the size would affect the memory.

5.2 Hardware Specifications

The Chess board was constructed out of wood and plexiglass along with screws, wood glue, wood filler, and silicone to hold the board together. The plexiglass made up the playing surface of the chess board (the top surface of the chess board), and the electrical and mechanical components were encased in a body built from plywood. Lines marked after being precisely measured were used to represent the patterned squares of a chess playing surface on the plexiglass board. This allowed for the plexiglass to still be transparent enough to view the mechanism below while illustrating a proper playing surface. The board was constructed light enough to be lifted and moved with ease, but it was constructed of a large enough dimension to accommodate room for the PCB, XY-Plotter, magnet, and all of the other components needed for this device.

For this to be possible, the materials used met standards to be sturdy, yet lightweight. The wood portion was crafted using Sande plywood with a thickness of 12mm and the top surface was a sturdy sheet of plexiglass with a thickness of 5.6mm. The board met requirements to be at least a measurement of 635mm x 635mm x 242 mm (L*W*H). The plexiglass top piece met requirements to be at least a measurement of 300mm x 300mm (L*W) with a border around the board making the total measurement of the top 635mm x 635mm (L*W). Each square on the plexiglass was 37.5mm x 37.5mm with a total of 8 squares per row and 8 squares per column (a standard 8x8 chess board).

The chess pieces will consist of 1 King, 1 Queen, 2 Bishops, 2 Knights, 2 Rooks, and 8 Pawns. These pieces were bought and constructed out of a lightweight wood to keep the weight low for ease of movement by the mechanism. The diameter of the base of each chess piece was a set diameter of 17.5mm, regardless of the height of each piece. The height of each chess piece closely followed the ratio set by the chess standards with the king being the tallest piece while the pawn is the shortest piece (view Table 22 for the dimensions of each piece). With the pieces resting in the center of each square, this left a maximum of 20mm between each piece allowing the magnetic mechanism the room needed to freely slide the 17.5mm diameter chess pieces between squares, following along the lines, to fulfill the player's desired move. LEDs were incorporated into the project placed at the bottom edge of the box with a few located on top, near the playing surface.

5.2.1 Microcontroller

With all the requirements necessary to successfully implement the design, considering how critical the memory capacity is, the amount of general-purpose input/output pins, clock speed, power consumption and cost, it was decided that the ATmega2560 microcontroller chip would be used with Arduino IDE. This would be the best option because Arduino software makes it easy to write code and upload it to the chip. Comparing the benefits of the types of microcontrollers researched, there had to be some tradeoffs, and because this project is not sponsored, cost was a considerable factor and due to the fact of the team's limited programming skills and knowledge, the Arduino IDE would make things less complicated. The team decided to go with the Atmega2560-16au for a few reasons but mainly because of the fact that it is an AVR chip which mean it's compatible with Arduino. The ATmega2560-16au is also the same chip on the development board that the team decided to use. Because of cost restrictions, the SUNFOUNDER development board was used. This board has the same specs as the Arduino Mega2560 development board except it was half the price. The development board was used in conjunction with the ATmega2560-16au microchip in order to test all components.

The ATmega2560-16au is a low power 8-bit microcontroller. It executes powerful instructions in a single clock cycle and achieves throughputs approaching 1 MIPS per MHZ, allowing the optimization of power consumption. The chip is used in various applications; some are clock & timing, embedded design and development, imaging, video and vision, and portable devices. A few important details are highlighted in Table 20 below.

Table 20: ATmega2560-16au Microchip Specifications.

Microcontroller Information	
CPU speed	16MHz
Program Memory Size	256KB
RAM Memory Size	8KB
Number of I/O's	86
Embedded interface	I2C, SPI, USART
Supply Voltage Max	4.5V
Supply Voltage Min	5.5V

Figure 10 below demonstrates and shows a schematic of all the components that were connected to the ATmega 2560. Starting from the top left, there was a 12V source coming from the wall outlet into the LM2596 step-down adjustable DC-DC switching buck converter. That same 12V source power also energized the electromagnet and the XY-plotter. Focused back to the regulator, it drops the 12V source down to 5V in order to provide a suitable power supply for the microcontroller to work. From the microcontroller, several peripherals were either supplied a source of power, sent PWM signals, or sent communication signals.

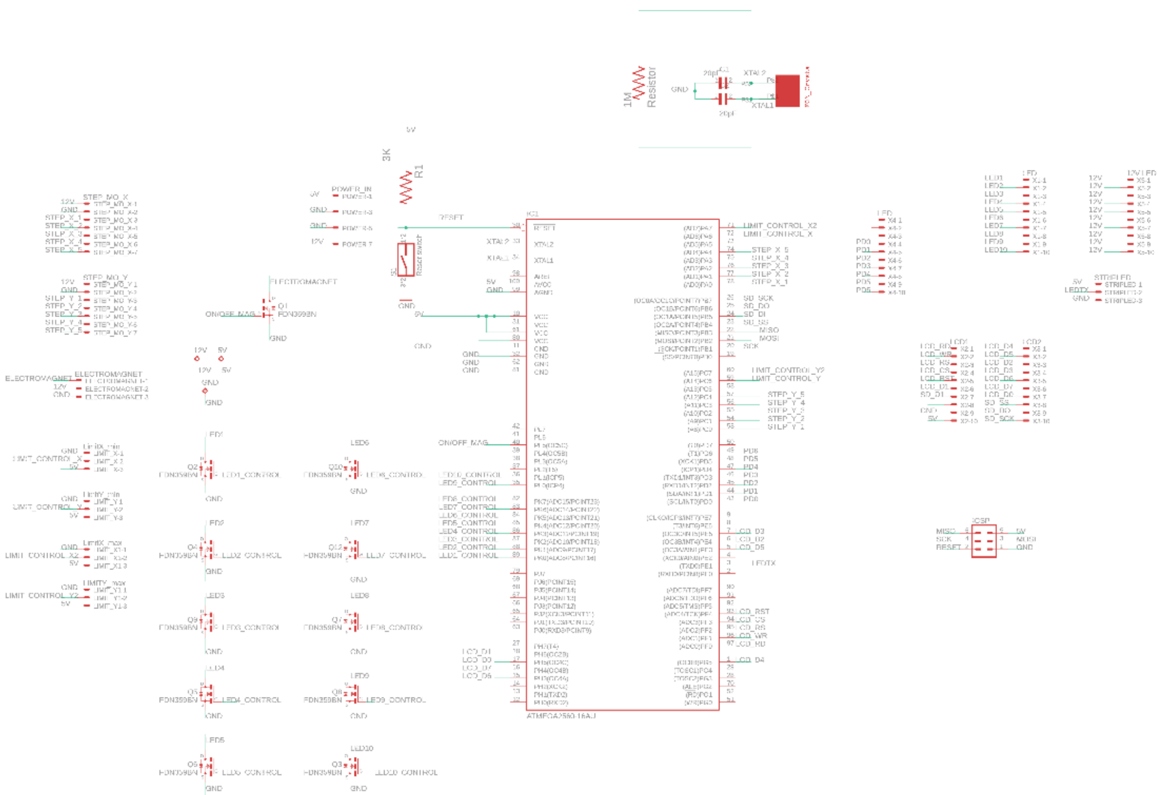


Figure 10: Microcontroller Schematic.

This first peripheral that was to be addressed was the LED strip. The LED had three connections, a ground, a 5V power source, and a data connection. The ground had the same reference point for every component that was connected in this schematic while the 5V power source was provided from the regulator and the data was to be sent via PWM through pin PL3 on the microcontroller. For the LCD to operate it was to be fed a 5V source from the regulator and tied to the same ground. From the microcontroller the LCD was provided with a 3.3V and sent multiple PWM signals. This peripheral required the most amount of connections from the microcontroller, one (1) digital supply voltage, and eight (8) PWM capable pins.

As mentioned before, the 12V power supply would provide power for the electromagnet through the source of a mosfet, it was used to control when the electromagnet turns on or off. The gate of the mosfet was connected to pin PG5 on the microcontroller. This connection sends a 5V signal from the microcontroller to the gate allowing the electricity to flow to the electromagnet. The voice recognition software, BitVoicer, needs to be able to communicate with the microcontroller and data was to be sent though PE1, the transmit pin, and PE0, the receive pin.

5.2.2 Piece Detection System

Initially, the team predicted for a system that would allow the computer to keep track of which pieces are on which tiles. The purpose of this was so that it could check for move legality and move pieces off the board when they get captured. For this to work, a system needed to be developed.

The Hall effect sensor system would be easier to implement, but the drawback was that it wouldn't detect which piece was on a tile, only whether there was a piece or not and so the player still needed to use some discretion, so they don't accidentally capture their own piece by making that command. The memory-based system would avoid these problems because it would allow the computer to know exactly which piece was on each tile at any given time. However, such a system would be much harder to implement since all of the associated software would need to be written. The RFID tags represented a happy medium in that it would allow the computer to know exactly which piece was on each tile and would not be as time consuming to implement as writing a program from scratch. The biggest obstacles were the cost of the RFID tags, which were more expensive than the other two options, and the practicality of mounting the RFID scanner in a way that gives it access to all the tags.

Given that time constraints were a major factor in this project, it was more feasible to create a piece detection system that was less effective and work around those defects than try to design a perfect system and waste time in the process. This notion was supported by the findings of other senior design groups who attempted to create a similar product. Both the DeepRGB and Interactive Automated Chess Game groups used a Hall effect sensor-based system to aid in piece detection while the Telepresence Chessboard group used an RFID-based system. None of the groups researched used a memory-based system, although the DeepRGB group did investigate it as an option.

Due to problems encountered while mounting the RFID tags and reader to the pieces and the XY-plotter, respectively, and problems with the reader potentially detecting the tags from multiple different pieces due to how close they are to each other, the memory-based software system was considered preferable. This memory-based piece detection system wouldn't need any specialized hardware to be implemented since it could be run on the same microcontroller (ATmega2560) as all the other software used in the project. The memory portion of it needed to be created from scratch or adapted from any other similar projects that could be found, but the chess rules portion can be directly borrowed from an open-source chess engine.

5.2.3 Piece Movement System

With the playing area of the chess board being a flat surface with both an X and Y plane, the electromagnet was required to move freely in both the X and Y directions mimicking the path and motion of chess pieces during normal play. To accomplish this, it was decided that an XY-Plotter would be implemented inside the chess board's structure to move the electromagnet to the desired chess piece and transport said chess piece through the path of the commanded move to its finishing location. An XY-Plotter has free range of motion in both the X and Y directions and can travel freely and precisely within the designated area of the plotter's specification. XY-Plotters are commonly found in CNC machines and 3D printers, as well as, kits sold online where the buyer builds the plotter programmed to draw any given design with a pen and paper. For this hands-free chess board design, an XY-Plotter could be built, or a kit could be bought, modified and implemented under the board to follow the given paths commanded and move pieces using an attached electromagnet with the same precision seen in other applications of this device.

XY-Plotter Requirements List:

- Dimensions had to fit within the chess board
- Lightweight, with respect to the rest of the board
- Smooth motion in both the X and Y directions
- Relatively quick speed but still stable
- Relatively low power consumption
- Easily programmable to work in this application
- Relatively cheap to buy or build

5.2.3.1 Dimensions

Since the XY-Plotter resides within the interior area of the chess board, it was very critical that the device dimensions fit the chess board interior dimensions near perfectly. The XY-Plotter had to at least reach each corner of the chess board's playing surface but must not exceed the exterior walls of the chess board due to interior spatial limitations. This dimensional requirement was put in place to assure that the XY-Plotter and attached electromagnet had full range of motion throughout the length of the chess board in both the X and Y directions allowing the mechanism to successfully retrieve and deliver the chess pieces to and from each square of the playing surface.

With the chess boards top surface being a measurement of 635mm x 635mm and the building material (plywood) being a thickness of 12mm, that left an interior area of 611mm x 611mm (L x W). The actual playing area on the top surface of the chess board, where the chess pieces reside and move, was a measurement of 300mm x 300mm. With these restrictions put in place, there was a tight window for the dimensions of the XY-Plotter. The XY-Plotter needed to be between 611mm x 611mm and 300mm x 300mm (L x W). Not only did the dimensions for length and width needed to be accounted for but the depth of XY-Plotter and interior depth of the box needed to be compared as well. The chess boards depth was a measurement of 254mm and after accounting for the thickness of the building material (12mm), that left an interior depth of 176mm, bringing the total interior volume to 611mm x 611mm x 242mm (L x W x H). This interior volume had to house all of the hardware used in the hands-free chess board's design and not just the XY-Plotter.

5.2.3.2 Lightweight

This product was meant to be portable and light enough to easily be carried from location to location depending on the desired playing atmosphere of the two opponents. With the XY-Plotter being one of the major components of this chess board design, the weight of the plotter made up a high majority of the product's total weight only falling behind the weight of the building materials making up the enclosure itself. In order to keep the final product lightweight enough for easy maneuverability and portability, the XY-Plotter needed to be relatively lightweight in respect to the rest of the hardware. Since the XY-Plotter was bought as a kit, weight was one of the deciding factors for which plotter was chosen. If the XY-Plotter were to be constructed, the same requirements were to be met.

5.2.3.3 Motion

In order for the final product to work successfully, the XY-Plotter chosen needed to meet the strict requirements set on and about the motion of the device. These requirements include steady and smooth movement in both the X and Y directions as well as precisely following the grid lines designated in the program. In this design, designated chess pieces were required to carry out each move by sliding from the starting square to ending square on the grid, via the move, rather than being picked up and placed down during traditional chess play. With that being stated, the chess pieces needed to be able to precisely maneuver on the grid lines between each square during a move from point A to point B. If the XY-Plotter was a fraction off on either side of the line, standing chess pieces could be bumped and moved, or knocked over, therefore, ruining play. If the XY-Plotter's movement was choppy and jerky rather than smooth and steady as stated in the requirements, the chess piece being transported may become unstable and tip over, furthermore, bringing the game to a halt. These motion requirements were a key in the decision of an XY-Plotter for this implementation.

5.2.3.4 Speed

Just like the requirements set on motion, speed requirements were needed to avoid similar errors to those listed in the section above. The XY-Plotter chosen to be implemented in this product could not operate at an excessive rate of speed. While transporting the chess piece in play via the electromagnet, if the plotter reached a high enough speed, especially while performing a 90-degree maneuver to the left or right, the piece may become unstable and tip over possibly colliding with multiple pieces. If this occurred, the game must be stopped, and the pieces must be placed in the correct locations to continue the game. If this occurred multiple times during play, the product would be unusable. On the other hand, if the XY-Plotter moved at a considerably low rate of speed, the game will drag out as both of the opponents spend most of the game experience watching the chess pieces slowly maneuver the desired paths. In choosing an XY-Plotter, the team found one with a programmable speed rate to avoid this issue.

5.2.3.5 Power Consumption

In order to design a hands-free chess board using voice activation to issue commands and an XY-Plotter with an electromagnet to transport chess pieces, many electronic devices were required for the operation of the product. These electronic components required some form of power being supplied to the board for operation and the two biggest consumers of the supplied power, as stated in the power supply section above, were the electromagnet and the XY-Plotter. When it came to the source of power, it was decided that a standard wall outlet would be used rather than battery power or renewable energy. The standard wall outlet in the United states is a 120V, 15A source putting out a maximum power of 1440W. Though this amount of power was available through the chosen power delivery method of this design, nowhere near that amount of power would be consumed in the operation of this device.

In order to keep this hands-free chess board ecofriendly and cost effective for users, a decision was made to keep the power consumption, while in use, similar to the power consumption seen by most inkjet printers on the market today. This brings the desired power consumption of the design to 30-50 watts while on standby and 300-500 watts while operating. To reach this power consumption goal, one of the main contributors of power consumption, the XY-Plotter, had to meet this goal of low power consumption not only while powered on and operating but while on standby as well.

5.2.3.6 Type of Magnet for Piece Movement

The magnet that was attached to the piece movement system needed to be able to “pick up” any of the chess pieces by attracting it, via its magnetic field, through the surface of the plexiglass chessboard. It needed to be able to hold onto the pieces after grabbing them and slide them across the board smoothly. The piece movement system could only support movement in two directions and cannot move closer or further from the board. Therefore, the magnet needed to be able to be switched on and off easily so that a piece can be “dropped” or let go of.

An electromagnet was the best solution for this application. Specifically, a DC electromagnet, would be more effective, for it behaves like a permanent magnet, providing a consistent magnetic field. However, the strength of the field could be fine-tuned by controlling the current flowing through the electromagnet’s coils until the best performance was achieved. The magnet could also be shut off by shutting off the current flowing through it. AC electromagnets were not considered appropriate because the strength and polarity of the magnetic field fluctuate together with the AC signal's fluctuations. Reversing the polarity would cause a complete failure of the system as pieces would be getting pushed away and knocked down unpredictably by the magnet.

5.2.3.7 Price

Out of all the hardware being implemented into this design, the XY-Plotter was known to consume the biggest portion of the budget. With that being true, it was decided that about 40% - 50% of the set budget of $800 be delegated solely to purchasing or designing and building the XY-Plotter being implemented into this product. Therefore, the price of implementing the XY-Plotter was a key factor in the process of choosing the correct plotter for the purpose of the design. In order for the project to stay within budget, either on or below, the XY-Plotter used in implementation would not exceed $400. Table 21 below shows the advantages and disadvantages of building and buying an XY-Plotter.

Table 21: Building Vs. Buying XY-Plotter.

XY-Plotter	Pros	Cons
Building	• Customizable dimensions. • Options for build materials. • Designed to required specifications.	• Extra research for every part required. • Buy all the parts. • Time consuming. • Relatively expensive.
Buying	• Save time for design in other fields. • Relatively inexpensive. • Includes stepper motor and analog servo.	• Limited customizability. • Limited options for materials used. • Difficulties meeting requirements.

5.2.4 Power System

Since standard wall outlets was used to fully power the device, the maximum power that could be used was 1440W. This was calculated based on a 120V, 15A outlet under a continuous load. Using any more power than that would cause the breaker associated with that particular outlet to trip and shut off the power to that outlet. Regardless, power consumption should be kept as low as possible for a host of other reasons. The less power used, the less heat would be generated by the device, which extends its life and prevents fire hazards from overheating electronics. Minimizing power consumptions leaves open the possibility of a fully battery-powered device. The intent was to first create a device which could be powered by a standard 120V wall outlet, however, if it was possible, then battery capabilities could be added as an optional feature.

The AC adapter from the wall would convert the AC power to DC power using a transformer, rectifier, and other electronic components that cleans the signal up in order to produce a constant DC signal. At this point everything else would be operated using DC. The team used a 12V 8A DC wall adapter as the power source. It was used to power every component associated with the chess board, the XY-plotter, the electromagnet, LEDs, etc. The 12V supply was connected to a buck switching DC-DC voltage regulator in order drop the voltage from 12V to 5V, this supplies the microcontroller and all the other necessary devices.

5.2.4.1 Voltage Regulator

The printed circuit board within the Smart Chess board utilizes various components. These components require different voltages and currents in order to operate correctly, if given the wrong input, these components may malfunction or worse, be destroyed. This was where a voltage regulator came into play. A voltage regulator is a component designed specifically to maintain a constant voltage. Regulators may consist of a feed-forward or a negative feedback, usually the negative feedback consists of a diode which provide protection to the component itself.

The Smart Chess Board is to be plugged directly into a wall outlet. The plan was to use a regular AC/DC adapter that would convert the outlet output from 120 volts, 15 amps AC to 12 volts, 15 amps DC. From there, the power source would be routed to the specific components that require 12 volts in order to operate, components such as the electromagnet, plotter and stepper motors. Another connection was made from the 12V source to the 5V regulator. The regulator is to then step down the supply voltage, providing a voltage of 5 volts to the microcontroller and any other component that required a 5-volt supply. The voltage regulator that the team decided on was the LM2596 step-down adjustable DC-DC switching buck converter. The team decided to go with this component for a few reasons. For one, the component already came as a module, which means it already has the necessary components in need in order to work. This would eliminate the need to build a circuit from scratch which could possibly have had errors, or simply just wouldn't be designed as efficient. The second reason was that by using this module, it would not require a heatsink.

Figure 11 below shows a schematic of the LM2596. The schematic shows a 12V source connected to the regulator that drops the voltage down to 5V. The capacitors, inductor, and the Shockley diode serves as protection. The capacitor before the regulator had to be bigger than the capacitor after the regulator so the electricity doesn't back feed though the circuit and destroy the regulator. The 12V power source was provided by the wall outlet and the 5V output voltage went to the microcontroller.

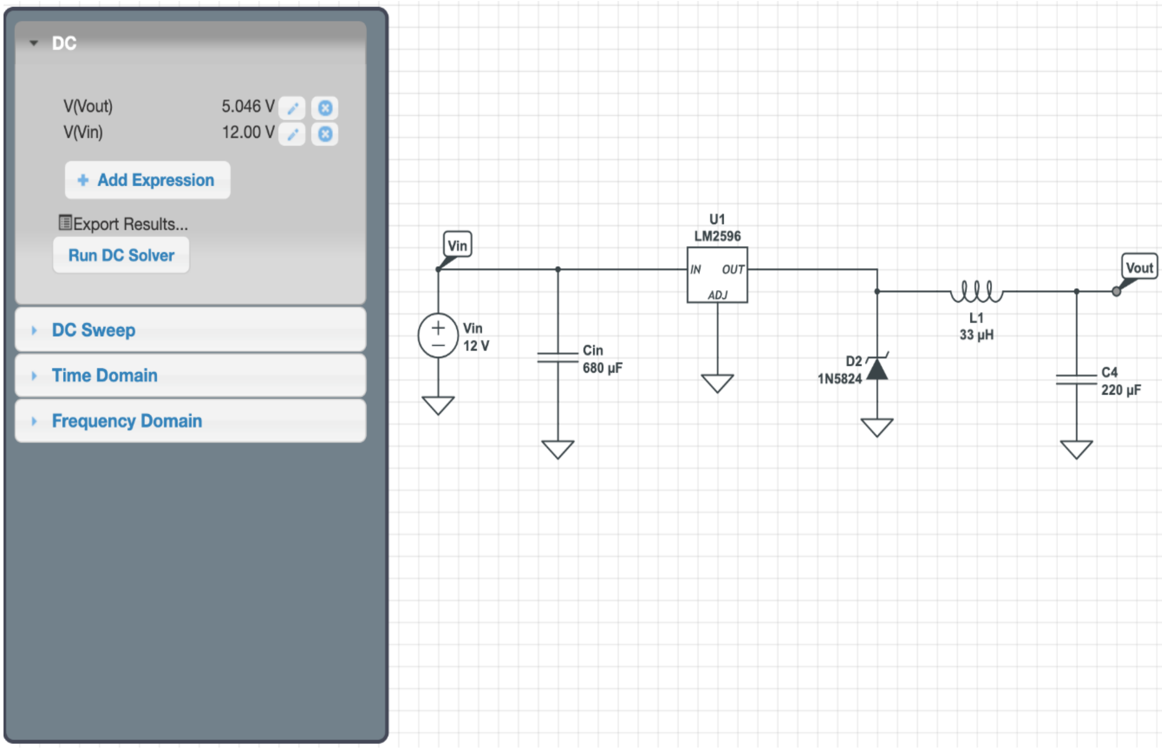


Figure 11: Voltage Regulator Schematic.

5.2.5 PCB Design

EAGLE design software was utilized to draw the schematic and the PCB. From the research done, a Rigid board would be the most practical choice to use for the chess board. It is the most common type of PCB out there and the most practical considering the application. The team ended up creating two iteration of the PCBs. The first design was manufactured by JLC PCB in China. This was a great choice because it ended up being a fraction of the price as compared to other manufacturers. The downfall with this was the estimated time to receive the board.

By the time it came around to order the second PCB, it had to be ordered from Osh Park, due to unforeseen circumstances. The budget took a big hit for this because it ended up being over three times more expensive than the first board, as seen in Table 22 shown below.

Table 22: Initial and final PCB design cost.

Manufacturer	Quantity	Cost
JLC PCB	5	$20.00
Osh Park	3	$ 67.35

After ordering the first board and realizing the mistakes that were made during the design phase, the team was given a great opportunity to redesign and fix those issues and do a second and final design of what was to be drafted. A lot of design changes were made, for example, a lot more GPIO pins ended up being used compared to the initial plan. The final PCB design can be seen in Figure 12 below.

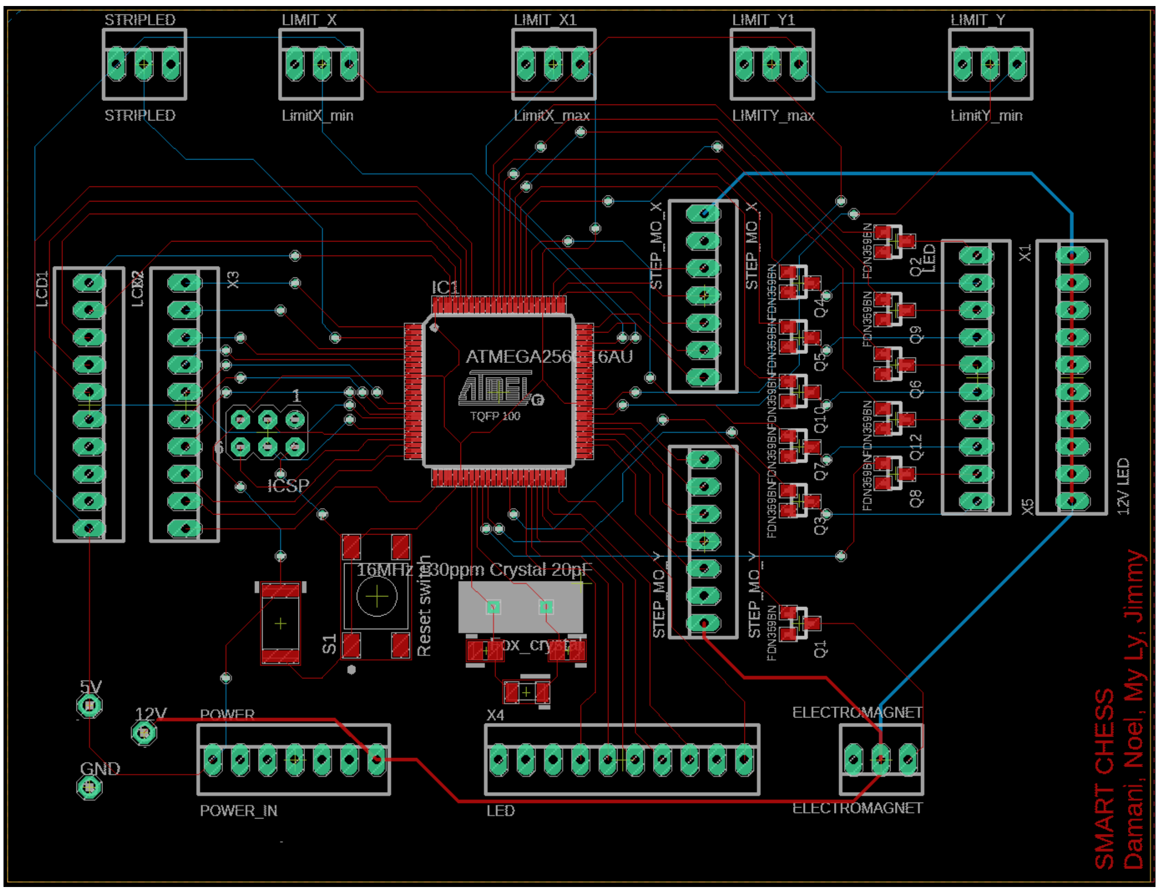


Figure 12: Final PCB design.

5.2.6 LEDs

Since the lighting option for this design was more for entertainment and cosmetics rather than a learning tool, using an RGB LED strip would've be the most accommodating choice. These LED strips have the ability to turn on, off, or flash a wide range of colors and options, depending on what they are programmed to do. The plan was to align the LED strip along the edges of the top surface of the chess board and have them display different colors depending on the circumstances. When the game begins, the LEDs were to continuously light up from one end to the other. Each player would be assigned a different color to know whose turn it is. When a chess piece was to be killed off the board, it would flash red a few times, when a piece was to be promoted, i.e. when a pawn reaches the opposite side of the board, it would flash blue, and when the game ends, the color of the winner will flash.

There were many choices of LEDs to choose from, considering brand and specifications, but as a team, NeoPixel by Adafruit were the ones that stood out the most. They come in all different sizes and shapes, through-hole single LEDs, in a strip, in a ring, and as well as in a matrix. Its option of being purchased with an integrated driver also made it a lot more convenient and simpler to add to the chess board.

The Adafruit NeoPixel Digital RGB LED Strip was affordable and flexible. Having a rubbery casing and bendable PCB strip, along with the width of 12.5mm with the casing and 10mm without, and 4mm thickness with the casing, makes it very accommodating to avoid interference with any of the other components. Not to mention the benefit of it not taking up much space, promoting the portability aspect of the chess board. The maximum amount of wattage required for these strips are 9.5W, 2 amperes at 5 volts per meter. When illuminating in white, the maximum was required, whereas the other colors only require a third to a half of that maximum current. The chip integrated in the strip used a single input pin and a single output pin, controllable by a microcontroller with a 100nS highly repeatable timing precision, making it very cooperative.

The development board and microchip the team decided to test on was the ATmega 2560-16AU board, and with this LED, they're compatible and have the capability and support to easily be controlled by programming them. It has the option to individually be controlled, with the pulse width modulation installed in each and every LED, to whatever the desired color. The flexible material it's made of is also weatherproof and can be cut to an appropriate length to fit inside the board.

5.2.7 LCD / Audio System

To provide an interface between the system and players to communicate moves without any physical contact during play, it was essential to have some sort of connection to instruct and direct. A great deal of research went into this section of the project and identifying a capable product at a reasonable price was a bit difficult. So to make it easier for the team, it was decided that the main means of communication from the player and the voice software would be a headset for the user so the system can clearly understand what was to be said and a screen for the system so the player can easily see the system's response.

While researching the different products for this section of the project, a few things had to be kept in mind, compatibility to the microcontroller, cost, and size. The main purpose of the LCD screen was to visually see what was supposed to be happening on the playing field, as well as what was being communicated by the players and the instructions by the voice software. The team found a relatively inexpensive LCD screen that was compatible with the Arduino. It was initially determined that a generic 3.5 TFT LCD Touchscreen for the Arduino Mega2560, with a resolution of 480x320, would be sufficient enough to perform the expected functions. The downfall with using this LCD screen was that there was no datasheet available, so the only attainable references were tutorials and internet resources.

The Elegoo UNO R3 2.8-inch TFT Touchscreen was the other option and purchase. The seller provided plenty of information, resources, and plenty of sample codes to get started. Some other advantages of this LCD included the support of the Mega 2560, had a 240 x 320-pixel resolution, a PCB module display size of 78.22mm x 52.7mm, an 8-bit parallel interface, and compatibility with 5V or 3V MCU with 5V-3.3V change-over circuit.

The team decided that a headset would be the most convenient selection for it would have the headphone portion, as well as, the microphone, possibly saving the group some money with a two-in-one device, depending on quality.

5.2.8 Building Materials

Though the electrical and mechanical components play a massive role in this project, the final project would not be able to come together and function as a whole without the presence of the chess board itself. For this design, both a playing surface for the chess pieces and a space below said surface to encase the electronic components must be designed and crafted using specific building materials to satisfy the main requirement of a functioning chess board.

Though less technical of step compared to the rest of the project, the choice in building materials contributed greatly to the final product as well as meeting the set requirements such as weight, portability, and total cost. Through much consideration it was decided that the list materials for construction of the chess board be as follows: plywood, plexiglass, wood screws, wood glue, wood filler, and silicone. These few materials came together creating the outer skeleton of the chess board in the form of a plywood encasing with a plexiglass playing surface, all being held together by the wood screws, wood glue, wood filler, and silicone.

5.2.8.1 Plywood

With the type of plywood for the project selected, the next decision was on which sheet of sande plywood would be most suitable for construction used for the chess board. It was known that a sheet of plywood was required to craft the five full sides and upper frame for the playing surface of the chess board. A typical sheet has a standard measurement of 4in x 8in (L x W), or converted to centimeters to match the dimensions diagrams, 122cm x 244cm. With the standard plywood sheet surface area, the common three thicknesses of plywood were ¼ of an inch (6.35mm), ½ of an inch (12.7mm), and ¾ of an inch (19.05mm). Those three options were compared with the requirements set for the building materials of this project before the final specification of sande plywood sheet was chosen for use in the product. The requirements looked over were that of weight, strength, and the price of the sheet as a whole and per square centimeter.

As stated in the previous sections, the final product of this chess board design was meant to be lightweight and portable for ease of use by the consumer. With that being stated and the sande plywood making up a majority of the outer body of the chess board, it was highly important that the selected plywood sheet keeps as low of a weight per square centimeter as possible without compromising the structural integrity of the material. With the difference in plywood sheet options resulting in the thickness per sheet, it was obvious that the thinnest sheet weighs the least while the thickest sheet weighs the most due to the increase in volume of plywood measured in cubic centimeters as the thickness increases. The weight of each sheet of plywood is as follows. The ¼ inch sheet of sande plywood weighs around 11.34 kg per sheet which comes to 38.1 g per cm^2. The ½ inch sheet of sande plywood weighs around 22.68 kg per sheet which comes to 76.2 g per cm^2. The ¾ inch sheet of sande plywood weighs around 34.02 kg per sheet which comes to 114.3 g per cm^2. The weight of each sheet of plywood was considered when deciding on the choice.

The next requirement that needed to be met by the chosen sande plywood sheet was that of strength. Since the chosen plywood for this product would make up the body of the chess board. The strength of the product as a whole relied almost solely on that of the building material used in construction. Unlike the weight of the plywood sheet, where the thinner sheet was the lightest weight while the thickest sheet weighs the most, the opposite was true when it came to the strength of the plywood. When a sheet of plywood is thinner, it becomes flimsy and less sturdy taking away from its structural integrity, therefore, reducing the strength. When a sheet of plywood is thicker, the sheet is stiffer and can withstand more force, therefore increasing the strength. With these facts being stated that means that the ¼ inch sheet of sande plywood has a low level of strength, the ½ inch sheet of sande plywood has a mid-level of strength, and the ¾ inch sheet of sande plywood has a high level of strength. The strength of each sheet of plywood was considered when deciding on the choice.

With both the weight and strength of each of the three choices of sande plywood taken into account, the last aspect to compare between the options was the price of each sheet of plywood. With the budget for the project firmly set with the decision to not exceed the set price while also attempting to come in as under budget as possible without compromising the structure and function of the final chess board project, the price of the plywood sheet must be included in the requirements alongside the weight and strength standards. When it came to price, the ¼ inch sheet of plywood cost $22.92 or $0.08 per cm^2, the ½ inch sheet of plywood cost $35.95 or $0.12 per cm^2, and the ¾ inch sheet of plywood cost $45.98 or $0.15 per cm^2. The cost of each sheet of plywood was considered when deciding on the choice.

After comparing all of the requirements with the specifications of the three options of sande plywood (¼ inch, ½ inch, and ¾ inch sheets of sande plywood), it was decided that the final choice of plywood for use in construction of the chess board outer body be the ½ inch sheet of sande plywood. This conclusion was drawn from looking at and comparing three different aspects of the plywood sheets in question. These aspects included the weight, strength, and price of each sheet of sande plywood. When it came to the ¼ inch sheet of sande plywood, although the weight and price ranks at the top, the sheet was much too weak for the application of this product. When looking at the ¾ inch sheet of sande plywood, although the strength was far above the rest, the weight and price greatly exceeded the requirements. The ½ inch sheet of sande plywood was superior when combining all three aspects compared to the other two proving to be the best choice for the application.

5.2.8.2 Plexiglass

The hands-free chess board designed for this project consisted of a clear playing surface on the top of the board allowing for viewing of the mechanism (XY-Plotter and electromagnet) during game play and function of the device. In order for this requirement of the project to be met, it was decided that a plexiglass sheet would be implemented as the playing surface of the chess board. The sheet of plexiglass in question chosen was OPTIX clear acrylic plexiglass sheet. This acrylic plexiglass sheet is clear allowing for the proper viewing as stated in the requirements, but much more research went into this decision. The OPTIX clear acrylic plexiglass sheet was chosen based on many aspects of the product. These aspects included the size of the sheet, the weight of the plexiglass, the strength of the product, and the price per sheet of OPTIX clear acrylic plexiglass.

With the dimensions of the chess board in place, the chosen plexiglass sheet was required to meet those dimension specifications for proper implementation and fit to the final product. With the top surface of the chess board being a measurement of 400mm x 400mm (L x W), containing a 300mm x 300mm (L x W) playing surface, and a 50mm border, the chosen sheet of plexiglass needed to be a bigger dimension than the playing surface to allow ample room for mounting the plexiglass sheet to the wooden frame via silicone. The OPTIX clear acrylic plexiglass sheet chosen was a measurement of 460mm x 610mm (L x W) allowing for the sheet to be cut to size to perfectly match the dimensions of the wooden frame (400mm x 400mm). This specific sheet of plexiglass was also chosen due to the thickness of the sheet which was 5.6mm.

The way this chess board was designed, the clear top playing surface (plexiglass in this case), was mounted underneath the wooden frame of the top surface being held together using the chosen silicone adhesive discussed below. In this case, the silicone adhesive was needed to support both the plexiglass playing surface as well as the 32 chess pieces in play. Although the silicone adhesive chosen met strength requirements for this application, the best way to aid the silicone was by lightening the load being held up by the product. The OPTIX clear acrylic plexiglass sheet was chosen for use of the playing surface due to the lightweight nature of acrylic plexiglass. This sheet of acrylic plexiglass was stated to weigh a total of 1.95 Kg meaning the sheet weighs 69.53 g per cm^2. This choice in material helped lighten the amount of weight being supported by the silicone allowing for a sturdier playing surface.

With the top surface of the chess board tasked to hold the weight of the 32 chess pieces in play while maintaining a flat surface for the XY-Plotter, located beneath, to track, the plexiglass sheet chosen has strength requirements that must be met for proper implementation of said product. That being said, the choice of plexiglass was decided partially on the factor of strength. The thickness of the OPTIX clear acrylic plexiglass sheet chosen was 5.6mm. This thickness contributed to the strength of the of the plexiglass sheet by creating a stiff surface that was less resistant to flexing and bowing due to weight or pressure applied in either direction.

The decision to use OPTIX clear acrylic plexiglass as the playing surface of the chess board was based on many aspects including the price of the plexiglass sheet. The price of a sheet of OPTIX clear acrylic plexiglass was stated to be $29.78 bringing the price per cm^2 to $1.06. Although the price may seem high in theory, compared to the other choices on the market, such as Lexan or glass, the price of the plexiglass was relatively low and cost effective.

5.2.8.3 Wood Screws

During the construction of the outer body of the chess board, wood screws were required to fasten the plywood sheets together that form the outer walls of the product. In order to account for this requirement, Wood screws was implemented to fully fasten the chess board together. After conducting research of the types of screws as well as the different brands on the market, the decision was made to use Everbilt flat head Phillip tip wood screws. This product was chosen for construction of the chess board due to many contributing factors. These factors include the diverse range of sizes, the lightweight nature of the screw, the strength of the material used for production, and the low price per container (100 screw) compared to the many options for wood screws on the market.

When choosing the proper screw to fasten the selected materials together in the construction of this chess board, many different wood screws suitable for the job exist on the market. With the choice of Everbilt flat head Phillip tip wood screws in mind, the only other decision needed for the ideal screw was the diameter and length of the wood screw chosen. Screw diameter measurements are standardized amongst all screws and follow a numbering system to designate the diameter of the outermost edge of the threads on the screw. The numbering for this system ranges from #0 to #20 with diameter increasing with the increase of designation number. For example, a #0 screw had a diameter of 1/16 inch (1.524mm), while a #20 screw had a diameter of 5/16 inch (8.128mm). Screw length on the other hand was measured as the length of the screw from the head to the tip. Like screw diameter, the length of a screw varies a lot and can be anywhere from ¼ inch (6.35mm) or greater.

With these two varying measurements, hundreds of combinations of screws were available for purchase. When choosing the size of screw for application of this project, the specifications of the material being fastened were used to help match the perfect screw to the project. After comparing sizes in conjunction with the plywood specifications, a decision was formed to use Everbilt #8x1-¼ inch (4.06mm x 31.75mm) wood screws in construction of the chess board's outer body. This screw was chosen because the diameter of the screw was exactly 1/3 the diameter of the selected plywood allowing room for the screws threads to grip the material without risk of splitting the wood. The length also allowed for plenty of threads to grip the material of the plywood without having enough length to veer too far to either side and protruding from the wall of the chess board.

The Everbilt wood screws chosen supplied plenty of benefits to this project and met all of the requirements put into place by the project. These benefits mainly came from the materials used to craft the screw. The Everbilt screws chosen were made from zinc plated steel which was crafted from the galvanization process leaving the finished screws relatively resistant to corrosion. These benefits from this material include the lightweight nature of each screw, the strength/grip of the screw, and the low price per container of screws.

The material of each screw was relatively lightweight helping the total weight of the final product remain low. Each screw weighed 2.11g bringing the total weight of the container of 100 screws to 0.211Kg. The chosen screws had coarse threads rather than fine threads. Both thread types were excellent in their respective applications but for wood fastening, coarse threads would dig in and grip the material of the wood allowing for a stronger hold. Amongst the rest of the benefits that come with this choice of wood screws, since these screws were made out of zinc plated steel, the price per screw was much lower than their stainless-steel counterparts. The price per container of wood screws (100 screws) was $6.25 coming to a very low price of $0.06 per wood screw.

5.2.8.4 Wood Glue

Although there were multiple options for wood glue available on the market, a decision had to be made for which brand of wood glue would help secure the plywood walls in the construction of the chess board. With the requirements for the project in mind, the final decision was made to use Titebond wood glue in the production of the board. This was due to the brands highly rated reviews as well as the amount of diversity in the wood glue line up.

With the brand of wood glue selected for use in the project, the next decision was on which wood glue in Titebond's lineup would be most suitable for use on the chess board. The Titebond lineup includes Titebond Original, Titebond II Premium, and Titebond III Ultimate. These products were all unique in their own way and benefited in different areas of the requirement needs. The specifications looked over and compared for each of the three choices were the features offered, the strength of the product, and the price of each unit. These specifications all played a huge role in choosing the final product.

The first specification looked over were the features offered by each wood glue to see the benefits of each individual product. When looking over each product specifications, the three key features that stood out were the location of use, any type of weather proofing, and the set time and temperature. When it came to location of use, Titebond Original was stated for interior use only. On the other hand, Titebond II and III (Premium and Ultimate) were stated to be for both interior and exterior use. Titebond Premium being for mostly interior with a little bit of exterior use while Titebond Ultimate could be used primarily in both climates. Weather proofing goes hand in hand with location of use for these products. Titebond Original has no form of weather proofing since it shall only be used on interior applications. Titebond Premium was water resistant since it could be used sparingly in exterior applications while Titebond Ultimate was waterproof because of its interior and exterior use. The final specification comparison was between the setting time and temperature of each of the three wood glues. Titebond Original must be applied in a temperature of 50° F or greater and sets in 4-6 minutes. Titebond II Premium must be applied in a higher temperature of 55° F or greater but has a shorter setting time of 3-5 minutes. Titebond III on the other hand, though allowing application in the lowest temperature of 47° F or greater, has the longest setting time of them all at 8-10 minutes.

The next specification looked over and compared was the strength of the three products in the Titebond wood glue lineup. With the wood glue being tasked to help hold the structure of the chess board together alongside the chosen wood screws, strength of the product was a huge deciding factor for the final choice of Titebond wood glue. When it came to the strength of each product, Titebond measured the strength in pounds per square inch (psi). Titebond Original comes in with the lowest strength of the three at 3600 psi. Titebond II Premium had a little higher of a strength rating at 3750 psi but yet wasn't the highest strength rating in the lineup. Finally, Titebond III Ultimate came in with the highest strength rating of the three products at 4000 psi. Although the difference in strength rating between the three doesn't seem like a lot, the amount of force able to be withstood was significant between them.

After reviewing the features and strength of the three products in the Titebond wood glue lineup, the final specification that was looked over and compared was the price of the three options. As well as the strength and features of the products available, the price of each option must meet the requirements set by the project description. The requirement that must be met for this specification was the budget of the project which would contribute to the final decision of either of the three Titebond wood glue options. When it came to the price of each product, as the level of wood glue increases, so does the price. Titebond Original came in with the lowest price of the three at $3.47 per bottle. Titebond II Premium had a small increase in price at $3.97 but was far from the highest priced option. Finally, Titebond III Ultimate came in with the highest price of the three options by a long shot at a price of $5.97 per bottle.

After comparing all of the requirements with the specifications of the three products in Titebond's wood glue lineup (Original, Premium, and Ultimate), it was decided that the final choice of wood glue for use in the construction of the chess board be Titebond II Premium. This conclusion was drawn from looking at and comparing three different aspects of the Titebond wood glue in question. These aspects included the features, strength, and price of each bottle of wood glue. When it came to Titebond Original, although the price ranks at the top, the wood glue was only meant for interior use and was much too weak for the application of this product. When looking at Titebond III Ultimate, although the strength was far above the rest, the price greatly exceeded the requirements. Titebond II Premium was superior when combining all three aspects compared to the other two proving to be the best choice for application.

5.2.8.5 Wood Filler

After the construction of the chess board was completed, a wood filler as needed to fill in the cracks and imperfections of the plywood molded outer body as well as cover and hide the screw heads for a sleek and smooth finish. This was done so that the final product would be presentable to the user. After research of many different wood filler brands on the market, the decision was made that the wood filler for use in this project would be DAP's product known as Plastic Wood all-purpose wood filler. This decision was based on the many features offered with this product as well as the reasonable price.

Looking at the features of Plastic Wood, a very important aspect of the filler was its ability to be used for interior and exterior applications much like choices for plywood and wood glue (Sande plywood and Titebond II Premium). With this feature in place, the product would have no boundaries for playing location. The next important aspect of the wood filler was the finish of this product. Plastic Wood all-purpose wood filler was stated to have a hard/smooth finish that looks and acts like real wood. This allows the final product to appear smooth and flush with no imperfections and no evidence of any filler being used. The finish of this product was also strong, claiming to be both shrink and crack resistant. The final important aspect of plastic wood was the ability for it to be stained and painted. This feature was important due to the fact that the final product will be both stained and painted for the user.

Alongside the features offered with the Plastic Wood all-purpose wood filler, the reasonable price of the product was a big factor in the decision of the wood filler.

At $4.48 per container, this product remained on the cheaper side of all of the wood filler products on the market while still offering great benefits and features.

5.2.8.6 Silicone

When constructing the top surface for the outer body of the hands-free chess board, the plexiglass playing surface must be mounted to the wooden border, therefore, creating the finished top piece of the product. In order for this to be accomplished, a silicone adhesive was required to bond the plexiglass sheet with the wooden frame piece. After conducting research on the many different options of silicone adhesives on the market, the decision was made that the silicone adhesive for use in this project would be product known as Loctite clear silicone waterproof sealant. This decision was based on the many features offered by this product as well as the reasonable price compared to the other silicone adhesive options.

The features of the Loctite silicone adhesive chosen match well with the requirements stated in this project. One aspect of these features provided by the chosen product needed for use in this chess board application was its ability to be used for both interior and exterior applications. This feature coincides with many of the choices for the other building materials, including sande plywood, Titebond II Premium wood glue, and Plastic Wood all-purpose wood filler. With this feature in place, the product would be safe for use in both indoor and outdoor locations.

The next aspects of the features provided by the silicone adhesive was its ability to withstand weather. The chosen Loctite product was stated to be both waterproof and resistant to extreme temperatures. Both of these aspects were highly important for the given application. With the electronic portion of this product residing within the constructed body of the chess board, maintaining a waterproof structure was key to keeping the electronics safe from shorting and causing damage. Also, with the electronics on the interior of the chess board as stated above, the interior temperature may remain higher than normal conditions. Since the chosen silicone adhesive was resistant to extreme temperatures, the applied adhesive shall remain sturdy and resist melting causing a collapse of the playing surface.

The final aspect of these features important to the project is the clear drying nature of the Loctite silicone adhesive. Since the plexiglass is clear and will only be bonded to the wooden frame using this chosen silicone adhesive, it is important that the product dries and cures clear resulting in a clean and professional looking finish.

Alongside the features offered with the Loctite clear silicone waterproof sealant, the reasonably low price per container was a big factor in the decision of this product. At $4.57 per container, this product remained on the cheaper side of all of the silicone adhesive products on the market while still offering great benefits and features.

5.2.9 Chess Parts

The game of chess has its own standards of size and proportions for the board and its pieces. The size of the chess board and pieces may vary with standards with specific countries and it is important to have the ratio of the chess piece to each square a certain proportion to one another. Since the King is the biggest piece, it would be the primary model to determine the appropriate dimensions of the board. The King's base diameter is to be within 40-50% of its height and approximately 75-80% of the board's individual square where the piece is to be placed. But because of the project requires hands-free movement, it was not possible to abide by these standards if the design was to work as envisioned.

5.2.9.1 Chess Pieces

A proper game of chess consists of 32 playing pieces in a complete set. Each player has a total of 18 pieces, one King, a Queen, two Bishops, two Knights, two Rooks, and eight Pawns. The chess pieces used in this project were designed specifically to meet the requirements and needs set by the design of the chess board. Table 23 shown below will state the dimensions of each chess piece in detail.

Table 23: Dimensions of Chess Pieces.

Piece	Height (mm)	Diameter (mm)	Weight/Piece (g)
King (2)	32.5	12.7	70.8
Queen (2)	31.75	12.7	62.5
Bishop (4)	26.98	12.7	50.0
Knight (4)	23.8	12.7	41.6
Rook (4)	20.64	12.7	37.5
Pawn (16)	19.8	12.7	33.3

Every aspect of the dimensions of this chess piece set were custom tailored, including the height, diameter, and weight of each playing piece. The pieces needed to be small enough to be able to move past each other but large enough to be visible. The sizes of the squares on the chessboard were limited by the working area of the XY-Plotter. The plotter had a working area of 310mm x 390mm which meant that the chessboard could not be bigger than 310mm x 310mm since it was a square. For this project, the area was rounded down to 300mm x 300mm to allow for a buffer in case the plotter cannot reach the edges effectively. This meant that the size of each square on the chessboard was 37.5mm x 37.5mm. Since the pieces needed to be able to move past each other, assuming that they were in the exact center of the square, the maximum base diameter of a piece of half of 37.5mm, or 18.75mm. To be safe, this was rounded down to 17.5mm because it could not be guaranteed that the pieces would be in the exact center of each square.

The material that the pieces were made out of was an important consideration because it determined how powerful the electromagnet needed to be as well as how sturdy the pieces are; lighter pieces would be easier to knock over, so a balance between very heavy or very light pieces was ideal. Wooden pieces would be the best, but wooden pieces cannot be 3D printed so they would need to be bought or manufactured by hand.

While some of the team members do have the tools and knowledge to manufacture pieces by hand, it would be easier and less time-consuming to buy them outright, and they would probably be more aesthetically pleasing as well. Plastic pieces could also be bought, or they could be 3D printed. 3D printing the pieces would be somewhat time-consuming, however, this meant that they would be custom-made in the exact size that was needed for the board. They could be made hollow and magnetized by gluing a small magnet or piece of magnetic material to the inside-bottom part of them so that the magnet can pick them up.

If pieces of any material were bought, the sizing would not be as good as if they were printed. Most chess piece sets have different base diameters for different pieces; the king would be the biggest and the pawns, the smallest. This meant that a set would need to be found where the biggest piece, the king, has a base diameter of no more than 17.5mm. This was much smaller than the majority of chess piece sets that were available for sale. Even if one were found, the rest of the pieces would then be very small and would be hard to see on the board and tell which piece was which from a glance.

The drawbacks from buying pieces could be dealt with due to the ease with which they could be magnetized. If wooden pieces were bought, or even certain types of plastic pieces, then they could be magnetized by drilling an iron or steel screw through the bottom of them or somehow inserting an iron or steel nail into them. This way nothing would be needed to be glued to the pieces, which would be better since anything that was glued could fall off and it would save time that it took to carefully glue and re-glue magnets or another metal material to them. If a screw or nail was embedded into the pieces, then it could be made deep enough to where the metal doesn't scrape against the plexiglass, which could potentially damage it. Most pieces that could be bought came with a piece of felt pre-glued to the bottom of the piece which reduces friction and allows it slide smoothly against a material like the plexiglass more easily. A screw or nail could be drilled through or under the felt material which wouldn't affect its functionality whatsoever.

The most anticipated outcome was that pieces would need to be 3D printed since no suitable chess piece sets have been found for sale on the internet after multiple team members have searched for several hours in total. There were free CAD libraries online such as grabCAD, which allowed for pre-made CAD models of different parts, including chess pieces, to be downloaded for free and modified to fit the dimensions and shape that would be the best for this project. It would also be free to 3D print the pieces because the university has a 3D printing lab that students can take advantage of for free (cost included in tuition). The most common material used in 3D printing found during research was ABS plastic, which was determined to be a hard and sturdy plastic that could be used in many common household items. Depending on the way the pieces were to be designed in CAD, it could be possible to embed a screw or nail into the 3D printed pieces without damaging them, a solution which would be the best of both worlds.

5.2.9.2 Chess Board Housing and Playing Field

The encasement of the chess board was an essential portion of the project for it embodies all of the components of the chess board. Getting the right measurements determined its portability and functionality, and its casing must be strong enough to hold and protect the materials contained inside. It was determined that the playing field had to be 300mm x 300mm Figure 13 below shows the dimensions of the playing field, a x b being and c x d equivalent to 37.5mm x 37.5mm.

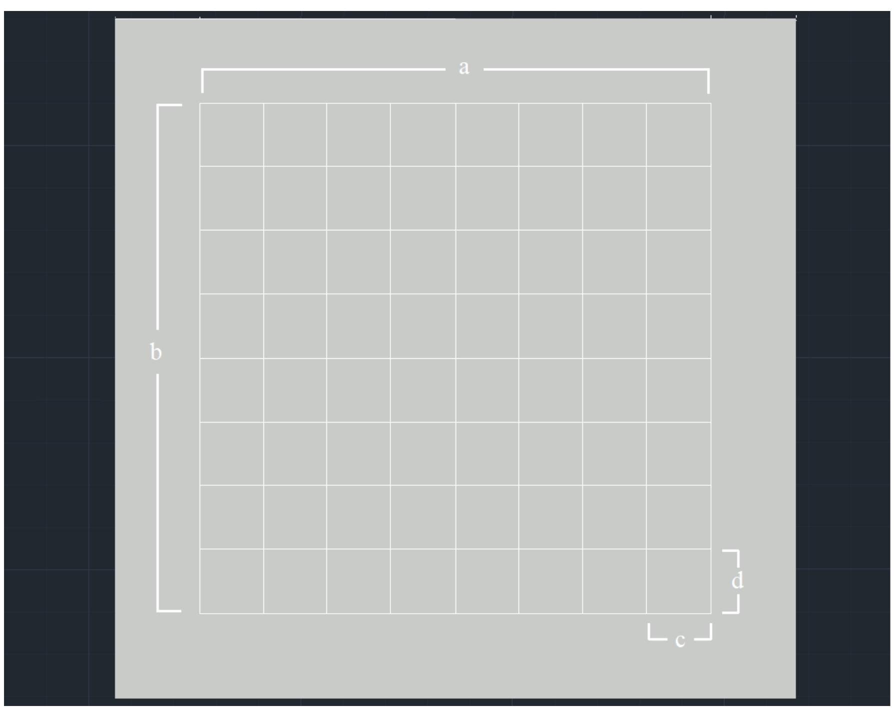


Figure 13: Top View Playing Field.

The measurements of the entire top surface of the chess board is shown below in Figure 14. This portion of the board was of Plexiglass and the patterned squared was expected to be of two different textures to easily differentiate them. It included the playing field along with the border to accommodate the “graveyard” for when a chess piece is to be eliminated from the game or a pawn is to be promoted. From the diagram, it could be seen that the dimensions of the whole surface were a x b, equivalent to 635mm x 635mm.

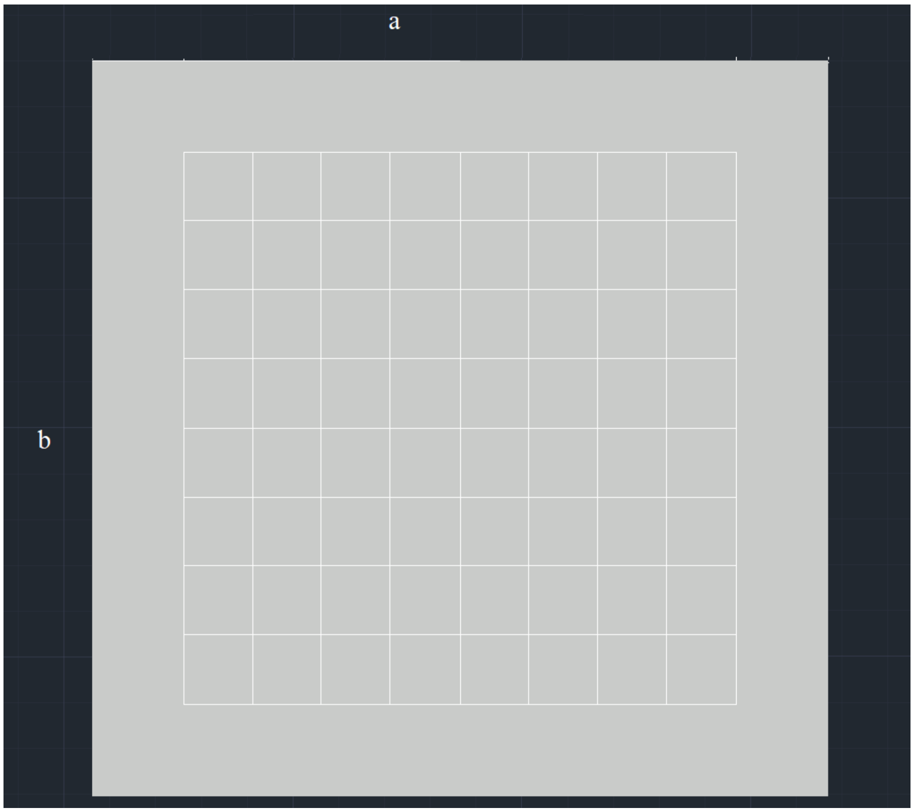


Figure 14: Top View with Border (Border Thickness of 50mm).

The sides of the chess board were made of plywood instead of Plexiglass to sustain the weight and force of the chess pieces stationed and moving above the surface of the board. Its dimensions were expected to be an equivalent of 635mm x 242mm. The base of the board was also of plywood since it would support everything within the board and especially if it were to be transported to a new playing location. The dimensions were equivalent to 635mm x 635mm.

After much consideration of the major components that would affect surface of the chess board, the plotter’s accessibility to go a certain distance in the X and Y direction and chess piece size, the final playing surface of the chess board was organized in a grid format. Each row was labeled with a specific number from 1 to 8 while each column was labeled with a specific letter from A to H. The total Plexiglass had concluding measurements of 635mm x 475mm, the layout of the playing surface with grids can be seen in Figure 15 below.

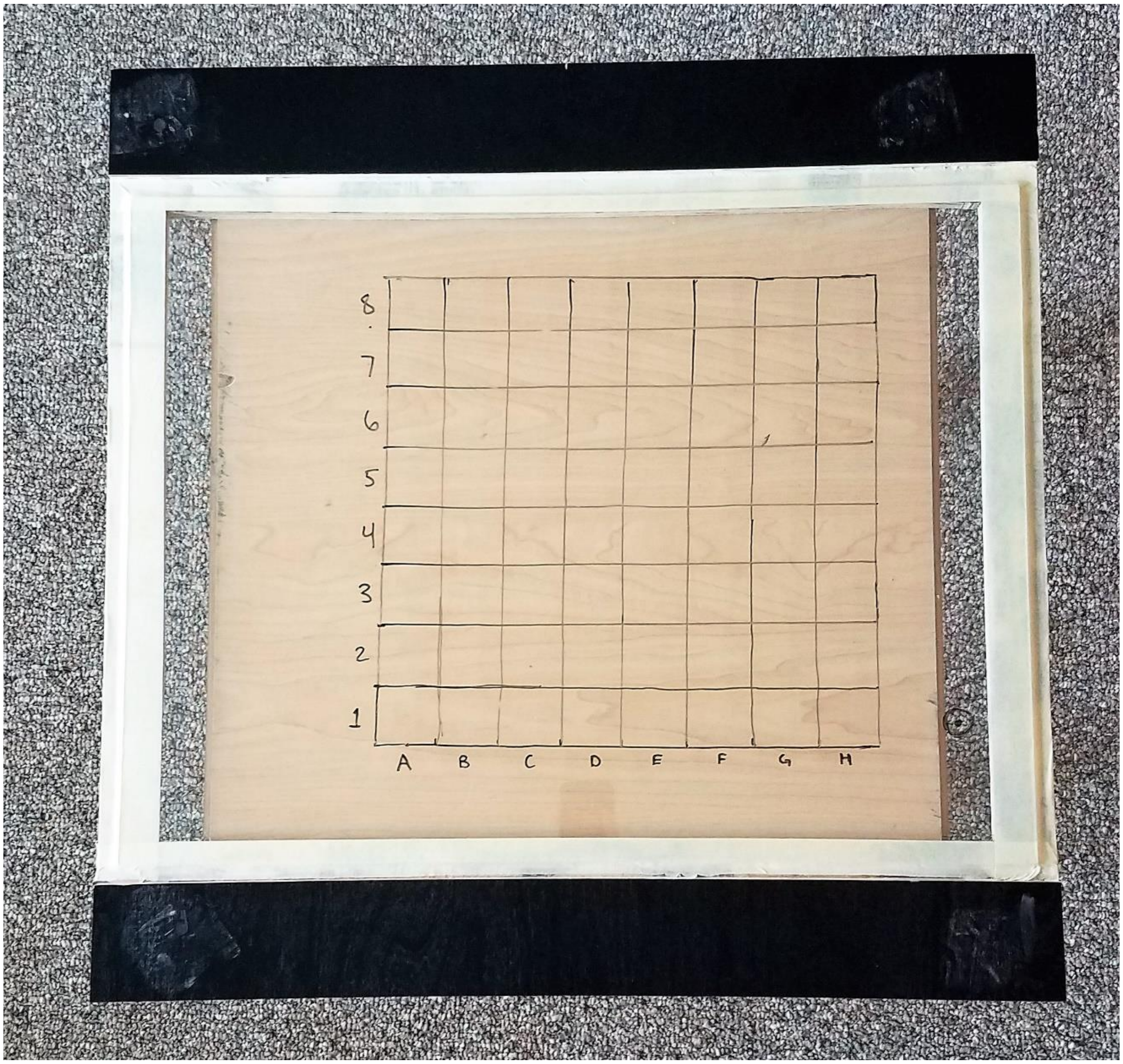


Figure 15: Final chess surface diagram.

5.2.10 Electromagnet

Choosing the correct type of electromagnet was a bit complicated because a great deal of variables needed to be considered. The electromagnet needed to use the least power as possible while maximizing the holding strength. It had to be able to work with DC voltages and it should not be too big, otherwise it would attract all the other pieces around the targeted piece. The electromagnet the team chose was the Uxcell 12V 180N electric lifting magnet.

Because the electromagnet needed 12V in order to operate, it was required to be powered straight from the wall adapter and be turned on and off using the signals sent from the microcontroller. The microcontroller was to be tied to the gate of a MOSFET and it would act as a switch allowing current to flow through the electromagnet. The electromagnet and the XY-plotter would then have to work in unison.

Figure 16 below shows the schematic that the team planned to use in order to control the electromagnet. As mentioned before, the microcontroller does not provide enough power to directly power the electromagnet. Therefore, the team had to use the 12V supply tied in with the source and the ground. The positive side was connected to the electromagnet, the electromagnet was then connected to the drain portion of the MOSFET, and then the gate was controlled by the voltage sent from the microcontroller. Since the turn on voltage of the gate was about 0.7 V, whenever the 3.3V signal was applied to the gate from the microcontroller, the MOSFET allowed current to flow through the electromagnetic causing it to be energized.

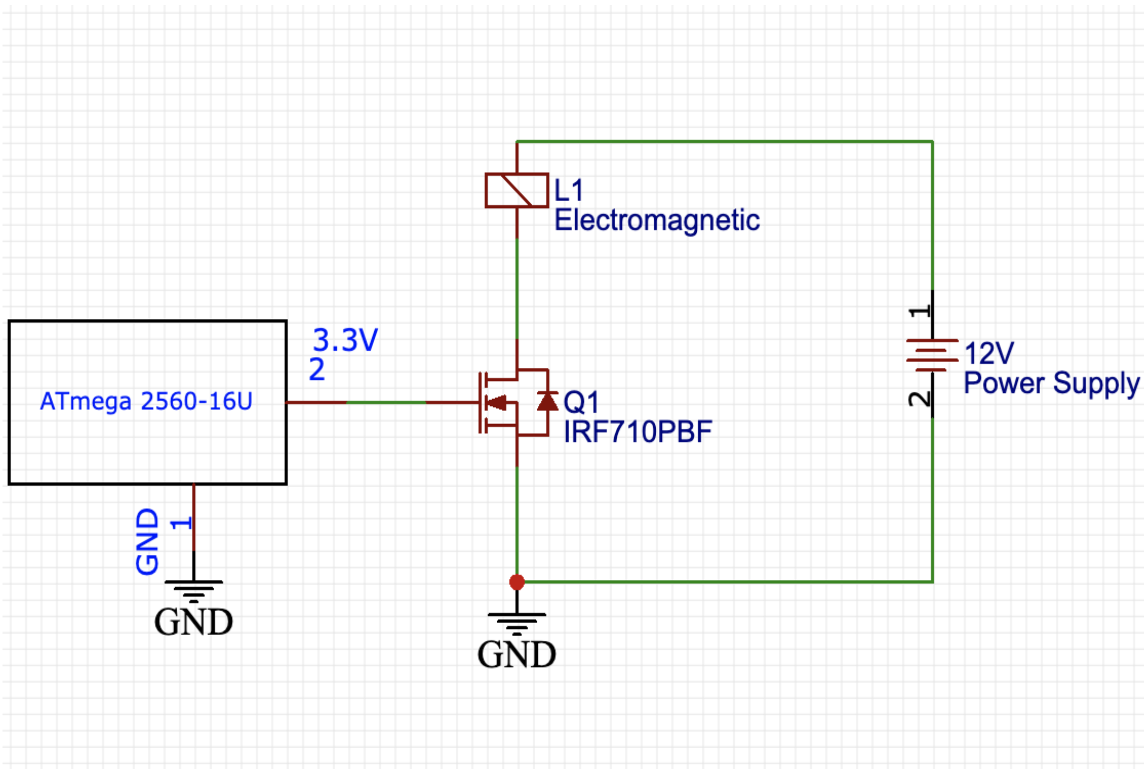


Figure 16: Electromagnet Schematic.

6. Assembly

One of the key factors in successfully accomplishing this project was researching and obtaining all of the right parts and supplies to build and assemble the design. After some discussion with the team, all of the supplies needed and required for the chess board were selected.

One of the most important aspects of this project was making sure that everything was durable and would be held together securely, not to mention having all measurements and parts precisely cut and bought so that the entire project would be aesthetically pleasing, and everything fits correctly. With that in mind, Table 24 shown below presents the materials required along with its measurements they come in as well as the costs that pertained to the construction of the housing of the chess board that encased the entirety of the project.

Table 24: Building Materials List.

Type	Name	Specs	Weight	Price
Plywood	Sande Plywood	L*W*T (cm) • 144x244x1.2	Per Sheet • 22.68 Kg	Per Sheet • $35.95
Plexiglass	OPTIX Clear Acrylic Sheet	L*W*T (mm) • 460x610x5.6	Per Sheet • 1.95 Kg	Per Sheet • $29.78
Wood Screws	Everbilt Flathead Phillips Tip	Diameter • 4.06 mm Length • 31.75 mm	Per Pack • 0.585 Kg	Per Pack • $6.25
Wood Glue	Titebond II Premium Wood Glue	N/A	N/A	Per Container • $3.97
Wood Filler	Plastic Wood All Purpose Wood Filler	N/A	N/A	Per Container • $4.48
Silicone	Loctite Clear Silicone	N/A	N/A	Per Container • $4.57

6.1 Chess Board Housing

Using AutoCAD, with limited knowledge and experience in it, the team was able to get a 3-dimensional diagram of the chess board to use as a visual reference shown in Figure 17 below. Having a length x width of 635mm x 635mm and a height of 254mm, the team knows the parameters and must have all other components be within the bounds of these measurements to be able to fit inside and successfully complete the design.

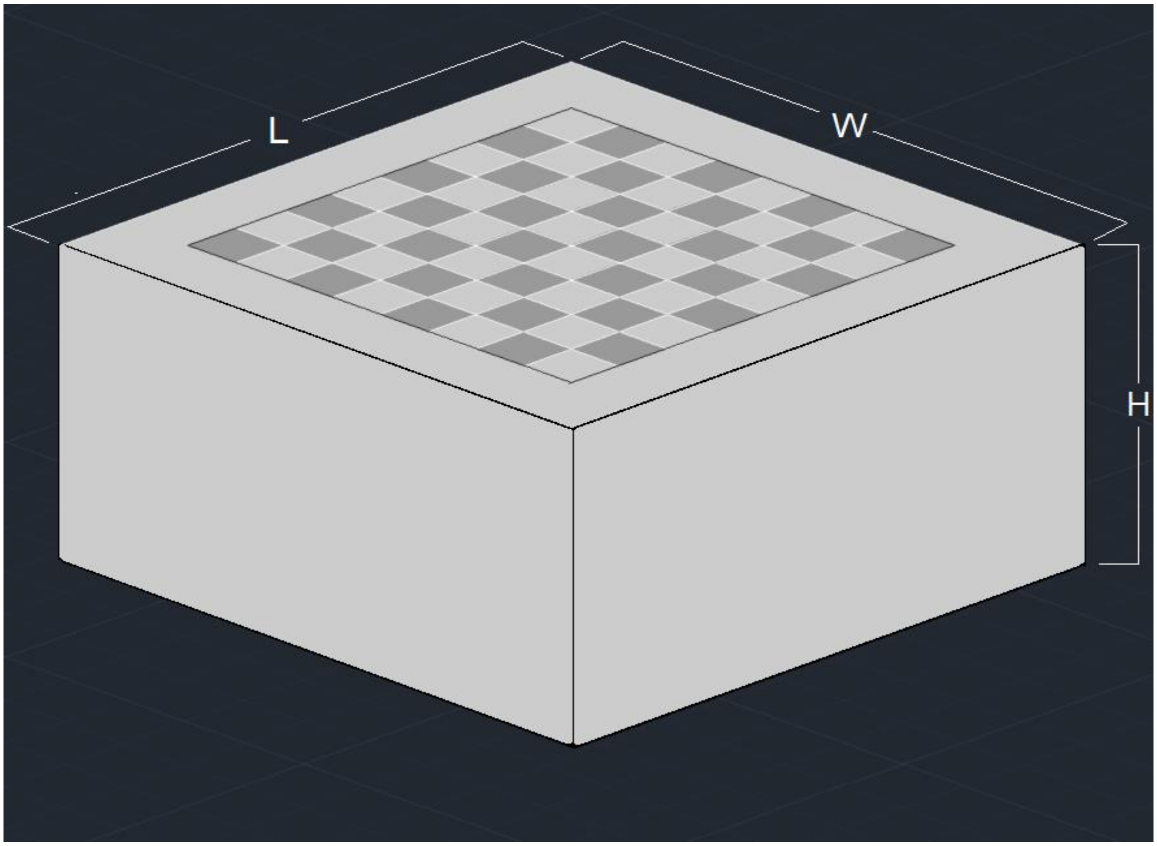


Figure 17: Chess Board Housing Dimensions.

6.2 Piece Movement System

After much research and discussion within the team and comparing the advantages and disadvantages between a robotic arm, buying and building an XY-Plotter, it was decided that buying an XY-Plotter would be the best decision for the piece movement system. For approximately $300, the XY-Plotter Robot Kit V2.0 was purchased with all the components required for the movement system was included. Parts such as the beams, brackets, plates, belt connectors, stepper motor, along with its brackets and driver, power micro servo, timing pulley, timing belt, shafts, coupling, linear motion shaft and sliding unit, cables, wall adapter power supply, and so much more.

What was so great about this XY-Plotter Kit was that all of the components were included so the team didn’t need to purchase any other parts regarding the plotter except for the electromagnet. With some questionable instructions and manual provided by the seller, the plotter was able to be built. The included components are listed below and can be seen in Figure 18.

- Aluminum Extrusion Parts
- Timing Pulley 18T
- Linear Motion Shaft D8 x 496mm
- Linear Motion Slide Unit 8mm
- Makeblock Orion Controller
- Stepper Motors
- Stepper Motor Drivers
- 9g Micro Servo Pack
- Micro Switch
- Cables
- Other Hardware and Accessories

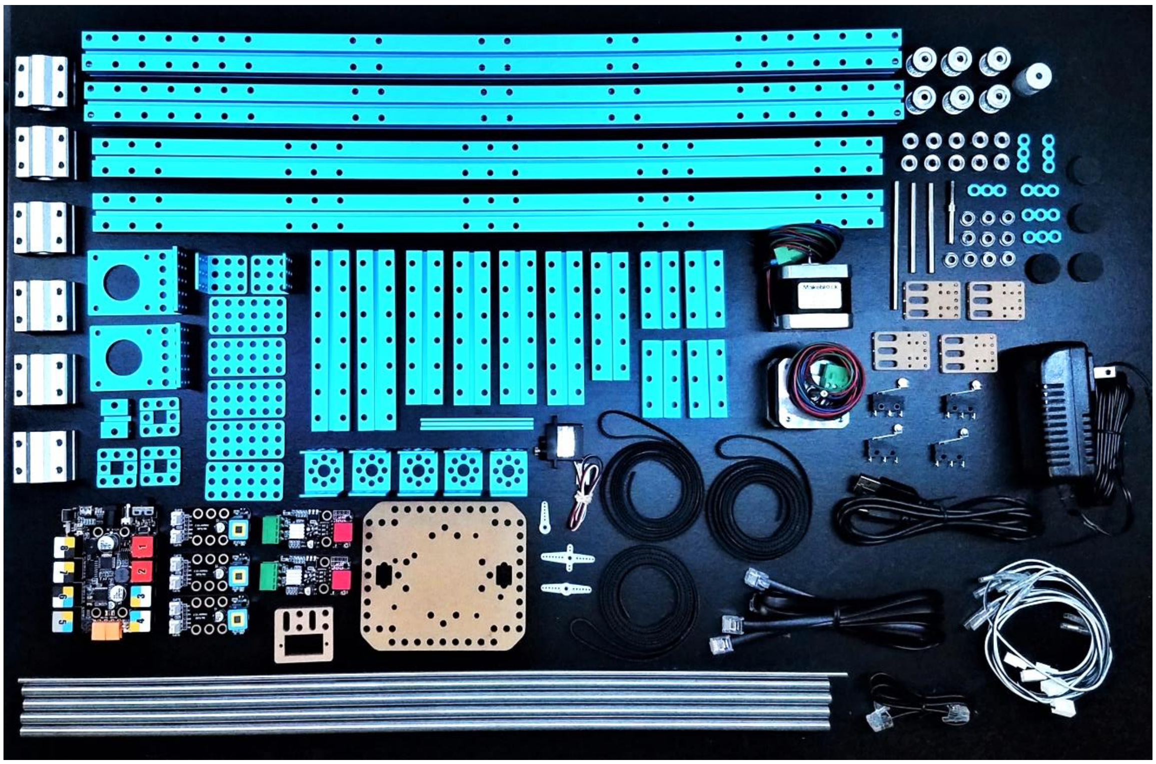


Figure 18: XY-Plotter Robot Kit 2.0.

After the assembly of the XY-Plotter, the dimensions were much bigger than expected. This caused some concerns with the portability aspect of the project, as well as the measurements planned for the actual chess board, which affected the size of parts and compnents. The dimensions of the plotter after assembly measured to be approximately 620mm x 620mm x 140mm (L x W x H). Below, in Figure 19, the finished assembly of the XY-Plotter could be seen, along with the main specifications.

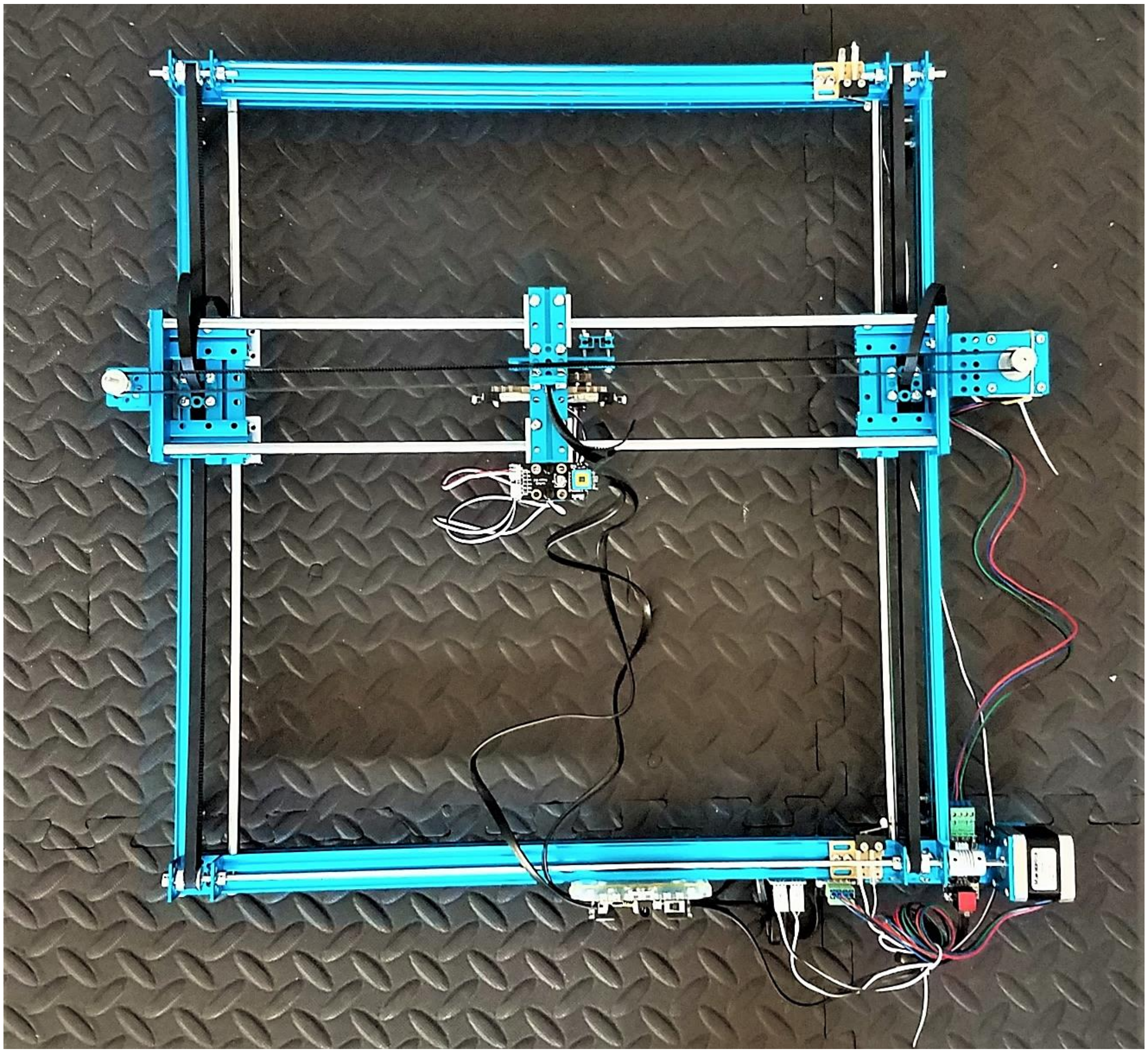


Figure 19: Assembled XY-Plotter.

- Frame: Anodized aluminum
- Physical Dimensions: 620mm x 620mm
- Working Area: X x Y: 310mm x 390mm
- X-Y Accuracy: 0.1mm
- Maximum Working Speed: 50mm/s
- Power: 100-240V~50/60Hz AC/DC
- AC/DC Power Adapter: 12V/3.0A
- Main Controller: Makeblock Orion (Arduino UNO compatible)

7. Project Prototype Testing

Prototype testing was a crucial step of the design process for saving time and money on product development. For this project, a bottom-up testing approach was chosen. Each component at lower hierarchy was tested individually and then the components that rely upon these components were tested.

7.1 Software Testing

All of the software testing in the planning and research phase on this project was to be performed on a team member's personal computer in a Windows environment. The software testing plan consists of four parts and would be tested separately until each section passes. The software testing plan could not be complete until these four sections pass the required test. The software testing sections included voice capture and recognition, piece movement system, piece detection system, and electromagnet. Figure 20 below is a flowchart showing the path to completion of the designed software testing plan.

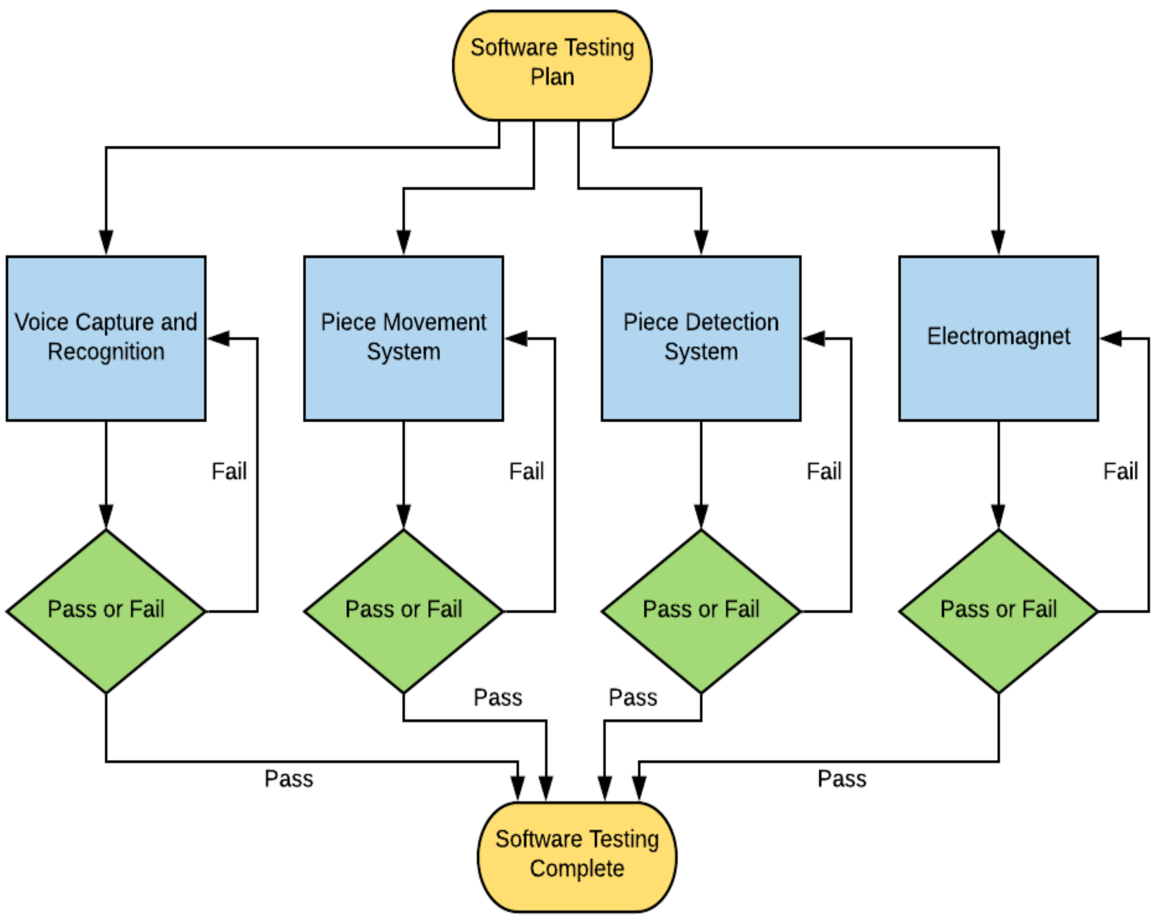


Figure 20: Software Testing Flowchart.

7.1.1 Voice Capture and Recognition Software Testing

To test the voice capture and recognition system, BitVoicer was downloaded and installed on some of the team members' PC. The microphone used was the default microphone array that was attached to the team members' PC (Realtek High Definition Audio). A Voice Schema was created that contained all of the possible commands that would be needed to play a game of chess. All of the piece names were added (pawn, bishop, knight, rook, queen, king) as well as special commands such as long castling, short castling and calibration (setting up all the pieces at the correct locations at the start of a game). The rank numbers from 1 to 8 were added to the schema, along with the file letters from A to H. For the letters, the NATO phonetic alphabet was used because a lot of the letters sound very similar to each other when spoken. The NATO phonetic alphabet was developed for radiotelephonic communication and represents every letter using a word to avoid confusing two letters which sound alike, as shown below in Table 25.

Table 25: NATO Phonetic Alphabet for Radiotelephonic Communication.

Latin Alphabet	NATO Phonetic Alphabet	Phonetic Pronunciation
A	Alfa	(AL-FAH)
B	Bravo	(BRAH-VOH)
C	Charlie	(CHAR-LEE)
D	Delta	(DEL-TAH)
E	Echo	(ECK-OH)
F	Foxtrot	(FOKS-TROT)
G	Golf	(GOLF)
H	Hotel	(HOH-TEL)

This alphabet was used because it greatly increased the accuracy when inputting audio streams into the software. With a minimum confidence level threshold of 60.00, the player would be able to speak at a natural rhythm and cadence without needing to over-pronounce their words and the software could still understand what they're saying with an adequate level of accuracy. At first the regular Latin alphabet was used, but this required a confidence level of over 80.00 to properly differentiate between different sentences which meant the player had to speak extremely slowly and deliberately.

Once this Voice Schema was created and set up on the BitVoicer application, then it was tested by multiple team members to ensure that the system was truly speaker-independent and can function with any person's voice. This also ensured that there was a sufficient tolerance to account for natural differences in pronunciation. Still, the testing set that was used was based on American English, and all of the team members that were involved in the testing naturally speak in that accent, so it is unclear how the system would perform for a player with a different accent. Usage of the NATO phonetic alphabet should help to mitigate the impact of regional differences in pronunciation, provided that the player is a native English speaker. Other languages were not tested for this project, although BitVoicer supports a lot of the world's most spoken languages. In theory, there should be no problem in developing a Voice Schema for any of the other supported languages as long as a spelling alphabet was used to deal with ambiguities.

Once it was established that the Voice Schema which was tested allowed for accurate interpretation of all possible permutations of items, the communication of commands with the microcontroller was tested. The testing setup for the communication involved connecting an Arduino development board containing an ATmega2560 microcontroller to the PC running the BitVoicer application. This was done via UART using a USB-to-serial cable to connect the team member's PC to the microcontroller via the Tx1 and Rx1 pins. The microcontroller was connected to a breadboard with an RGB LED on it which was powered by the same power supply as the microcontroller. A new BitVoicer Voice Schema was created that only consisted of sentences/items to turn on or off each of the three colors on the LED. These 6 sentences were associated with 6 different commands that would turn on or off each of the three colors. These commands were written in the Arduino language, which was very similar to C++, and adapted from the build-in Arduino library that came with the BitVoicer application.

A team member spoke each of the anagrams into the microphone and observed the communication pane on the BitVoicer application to check if the commands were being sent properly as can be seen in Figure 21 below. Then the team member observed the RBG LED on the breadboard and verified that it turned on and off and changed colors whenever the appropriate commands were spoken. Once it was determined that this simple testing scheme worked properly, then it was concluded that the microcontroller and the PC were communicating with each other appropriately. In spite of the fact that UART supports bi-directional serial communication, only one direction was required for this application: from the PC to the microcontroller. Thus, only this capability needed to be tested.

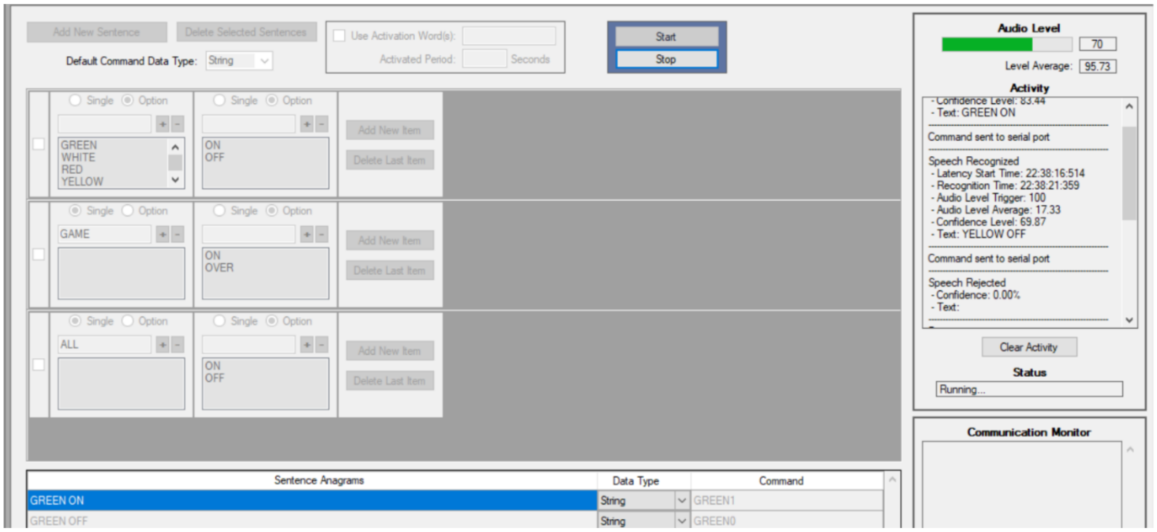


Figure 21: BitVoicer Testing.

Once it was confirmed that the voice recognition software can accurately identify relevant voice commands and communication with the microcontroller, then the chess engine needs to be tested to see if it can be run off the microcontroller. The chess engine that was chosen for this project, the TSCP, used too much memory to be able to run completely off the ATmega2560 microcontroller board that was purchased so the chess engine was needed to run off a team member's PC. The testing for the chess engine was discussed further in section 7.1.3 Piece Detection System Software Testing.

7.1.2 Piece Movement System Software Testing

The piece movement system, consisting of the XY-Plotter, was tested by writing a test program that could go through all of the possible movements and worst-case scenarios that the XY-Plotter could experience. This XY-Plotter was originally meant to draw with a pencil vector graphics that were developed on a computer; however, for the Smart Chess Board project there was no need to draw anything and so the XY-Plotter needed to be controlled directly based on commands sent from the voice recognition system and processed by the microcontroller. The GRemoteFull software package that was included in the XY-Plotter V2.0 Software Manual contained a control program, control software, and G-code examples and was used to manually control the XY-Plotter. The software was downloaded from the software manual and installed on a team member's PC. The XY-Plotter mainboard was connected to the PC using serial communication via UART. During the installation, the proper mainboard and COM port was selected to complete the process.

Once the GRemoteFull software package was successfully installed on the team member's PC, GRemote automatically opened a graphical user interface. Using this interface, the XY-Plotter V2.0 was able to be controlled with a mouse and keyboard. The directional arrow keys were used to control movement in the X and

Y axes. The “<” and “>” keys were used to control movement in the Z axis. The XYZ axis increments could be set by the developer up to a minimum of 0.5mm steps. The application contained many other shortcut keys which could be used for various functions, but none of these were needed since the XY-Plotter was going to be interacted with by translating the interpreted voice commands from the microcontroller directly into the corresponding stepper motor movements.

The GRemote software could recognize G-code files with the suffix .cnc. CNC files can be created from converting normal imagine formats such as jpeg or tiff, from SPG files, or from 3D CAD files from any CAD program. GRemote would automatically convert these images into G-code. The syntax for this G-code involved specifying which of the stepper-motors to activate and how many millimeters away from the origin it is to be going to move.

Since the pieces would not be manually moved, the pieces required the ability to maneuver to the desired spot without any obstructions. Given that the XY-Plotter has the capability to move within 0.5mm increments, there were no problems in maneuvering them around other pieces as long as the ratio between the piece base diameter and the square sizes were small enough to allow them to pass through.

A set of dummy chess pieces were created out of small blocks of wood, with small magnets attached to their bottoms. These dummy chess pieces were the size of the biggest chess piece, the king, to ensure that the worst-case scenarios were tested. The test performed involved setting up the dummy chess pieces at the same distance away from each other as they would be if they were on the center of a chessboard square on a piece of Plexiglass. This Plexiglass was of similar thickness to the one to be used in the final housing. Then the XY-Plotter was manually controlled through the keyboard shortcuts with the lowest axis increments set up until a piece could be moved in between two other pieces without knocking them over or being obstructed.

7.1.3 Piece Detection System Software Testing

The software testing plan for the piece detection system consisted of running the chess engine and entering data into it using the voice recognition system. The chess engine (TSCP) itself would serve as the piece detection system. This chess engine kept track of all of the piece locations at every point in a game of chess and gets updated after every successful move made. It does this by storing the piece locations in a 10x12 “mailbox array”, the functioning of which were discussed in section 4.8.1.1.

In order to test the functionality of this piece detection system, different chess commands were entered into the TSCP chess engine. The TSCP came with a function that allowed the user to display the contents of the mailbox array directly to the command prompt. After each move was made, the array was displayed and compared with the expected result.

Different cases were tested to make sure that the chess engine could properly identify every type of illegal move that could occur. The cases that were tested were pieces moving out of bounds, pieces moving in ways that they're not allowed, pieces trying to capture other pieces from their same team, and improper syntax. For each of these cases, the expected result was that the TSCP chess engine returned the message “Illegal move”. If this happened, then the functionality of the piece detection system was determined.

After some initial testing, eventually it was decided not to use the TSCP chess engine as the method through which to take care of move legality issues. The TSCP worked beautifully on its own, and the BitVoicer provided an adequate voice recognition software, but integrating them proved to be more of a challenge. All of this testing was done in a Windows environment, and this may have been a cause for the issues with integrating the two pieces of software properly with the microcontroller.

The motivation to use these two tools in particular came from their prior use in other chess projects that were looked at for guidance in addition to further independent study. However, these other projects were running their software in a Linux environment and their documentation reflects this. Our team did not use Linux and decided the time needed to figure out a way to write a script to transfer the output of the TSCP program back and forth between the Arduino was not worth it.

Instead, alternate solutions that had not been looked at before were considered. The one that gained the most traction was to write the chess program rules directly into the same software that controlled the motors and BitVoicer. This would require more critical thinking than using an existing chess engine, however it would not be as difficult to write as a custom chess engine. The computational algorithms that were researched earlier such as Minimax and Alpha-Beta Pruning were only required when the chess engine was playing chess. The part of a chess engine that took care of the move legality was the only the part that was needed, so a chess engine or another project's code could be looked at as guidance for that specific part only.

7.1.4 Electromagnet Software Testing

The software testing for the electromagnetic was not as extensive as the hardware testing. Since the electromagnet required a supply voltage of 12V, it would not be possible to send a signal straight from the microcontroller to turn on the magnet. With the hardware testing already being completed, the only challenge with the software testing was to see if a signal could be sent from the microcontroller to turn on the magnet. The plan to test the electromagnet was to use the 12V supply as the power source for the magnet, tie the microcontroller to the gate of a MOSFET and use the MOSFET as an on/off switch, which allowed the electromagnet to receive the 12V supply. If the team could accomplish sending a signal from the microcontroller and be able to control the length of time the electromagnet could be one for, then it had passed the software testing phase.

7.2 Hardware Testing

The limitations to any design and this design was the hardware. In the following sections, the team covers the various methods used to test the different hardware components within the project. A lot of the hardware testing will be done using the Sunfounder ATMEGA2560 Microcontroller development board. The board was used in conjunction with a breadboard, oscilloscope, power supplies, a multimeter, and other necessary supplies. A signal was to be sent from the microcontroller to a prebuilt circuit on the breadboard just to see if the component actually worked or not, and it helped to understand the best possible way to get the desired output or response.

7.2.1 Voice Capture and Recognition Hardware Testing

Although voice recognition was mostly software based without hardware or peripherals, there would be no way to capture the input from the user. In order to test the voice recognition, the team would need a microphone or some other way of interfacing a person voice with the BitVoicer software. After the voice was processed by the software it would then communicate and send commands to the microcontroller via serial communication. To start interfacing with the XY-plotter seemed complicated, so the decision was to test the voice recognition software with a breadboard and LEDs. Simple commands such as lighting them up in order or having them all blink at once will demonstrate that the team had understood how to send command via BitVoicer properly.

Once testing the LEDs had been properly accomplished, the team could then move on to interfacing with the XY-plotter. The XY-plotter was be more complicated because it involved building a library with a name for each position that would correspond to specific coordinates that would be recognized by the plotter. The electromagnet would also have to be able to turn on and off at the correct times when moving to the different positions.

7.2.2 Piece Movement System Hardware Testing

The piece movement system hardware test consisted of two components. These components included the XY-Plotter, which was used to transport the pieces along the playing surface of the chess board, and the electromagnet, which was attached to the XY-Plotter to grab and release each chess piece during movement. In each case, the hardware tests must be successfully completed to verify each part before the parts may be implemented into the chess board’s design.

7.2.2.1 XY-Plotter Hardware Testing

Before the implementation of the XY-Plotter in the chess board design, the plotter's hardware components needed to be tested to ensure that every possible movement that was called for in the specifications and design was possible. The playing surface of the chess board was set as a 300mm x 300mm square consisting of 64 37.5mm x 37.5mm squares arranged in an 8x8 pattern.

To begin the testing plan of the XY-Plotter, the mechanism must be able to reach each of the four edges of the of the playing surface from top to bottom. In order to complete this task, the plotter would attempt to trace the entire 300mm x 300mm outer boarder of the chess board's playing surface. During this process, the center of the plotter must remain directly on the gridline and no corners shall be cut while tracing said line. Obviously, the board was designed with the given specifications for the XY-Plotter in mind, but it needed to be tested to make sure it actually had a full range of motion in real life.

Once this task was successfully completed, the XY-Plotter was then traced the gridlines set in place by the software. The playing surface consists of 64 37.5mm x 37.5mm squares arranged in an 8x8 pattern. The lines of these squares must be traced by the XY-Plotter showing that the mechanism as able to follow the set gridlines on the board. This test was necessary due to the fact that these gridlines are the desired path for chess piece movement during gameplay. Like the task set prior, the center of the plotter must remain directly on the gridline during the entirety of this process.

Once this task was completed, the XY-Plotter would be required to pass one final task. This final task consisted of the XY-Plotter maneuvering to each of the 64 squares of the playing surface and stopping directly in the center of each of the desired squares for a period of 1 second. Since the chess pieces would rest on the center of its respective square, the XY-Plotter would need to reach those locations in order to attach to stationary chess pieces and transport them to the proper location following the completion of the move.

Since the pieces was not to be manually moved, the pieces would require the ability to maneuver to the desired spot without any obstructions. While moving the pieces, they had to be standing in an upright position also. To achieve these two goals, the team had to understand exactly how far a step or fraction of a step was in relation to the squares on the playing surface. In order to achieve this, the exact size of the playing board had to be used. In order to move the pieces into different positions, the team would have to understand how to guide the piece movement mechanism in between the squares along the line.

The stepper motors on the XY-Plotter also needed to be tested to ensure that they could move at full steps (1/1), half steps (1/2), quarter steps (1/4), eighth steps (1/8), and sixteenth steps (1/16). This needed to be completed to ensure that the

XY-Plotter had the full ability to follow the grid lines put in place in the software aspect of the project.

Figure 22 is a flowchart showing the process of the hardware test for the XY-Plotter system. For each of the three tasks, the process would be the same. First, the desired task would be performed. If the task passes, the test would move onto the next task. If the task fails, it would fail under the category of a hardware fail or a software fail. During a hardware fail, the problem would be diagnosed, and the task would restart. During a software fail, the test would be terminated, and the software would then be retested according to the test plan. These steps would repeat until the XY-Plotter hardware test is complete.

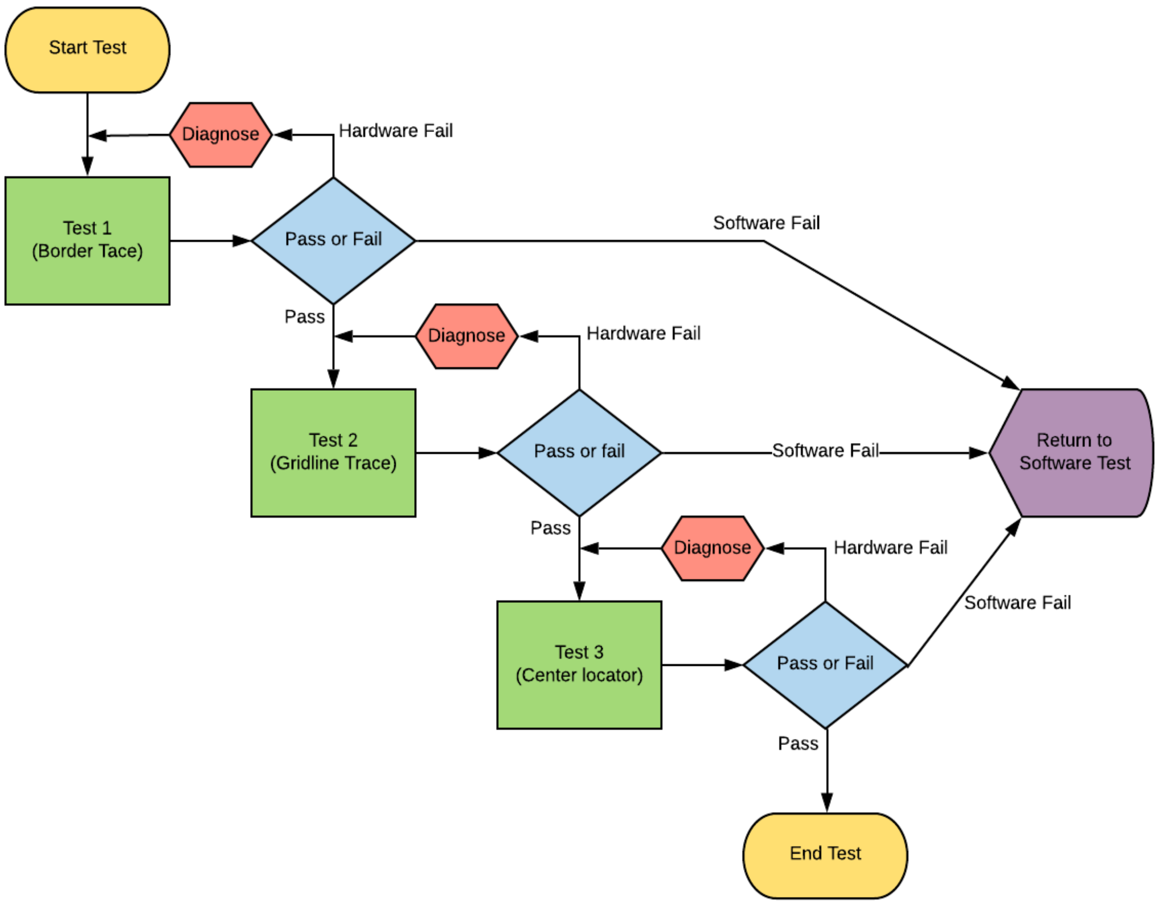


Figure 22: XY-Plotter Hardware Test Flowchart.

7.2.2.2 Electromagnet Hardware Testing

The electromagnet that was ordered by the team for use in this project was the Uxcell 12V 180N electric lifting magnet. This magnet was mainly chosen because the operating voltage required was relatively low compared to its Peak force. The only way the team determined this was to physically test the magnet. Hardware testing for the magnet was done with a power supply from the lab, different size pieces of Plexiglas, and actual pieces, or an exact replica of the pieces, were used in order to determine if the chosen electromagnet met the requirements that were set for this project.

It took some trial and error before finally deciding on electromagnet. The first one that was purchased was the 12V 50N and it wasn't strong enough to attract to the magnet in the chess pieces. The team then went above and beyond and purchased the 12V at 500N and it was definitely bigger and a lot stronger, attracting all the pieces surrounding the electromagnet. So finally, the last and most compatible electromagnet that was purchased was the Uxcell 12V at 180N. The size wasn't too bulky, and it had just enough power to attract the magnet without interference. In Table 26 below, the different electromagnets are compared.

Table 26: Electromagnets.

Peak Force	Cost
12V 50N	$ 8.57
12V 180N	$11.53
12V 500N	$20.34

A power supply was used from the lab during the hardware testing because it was able to provide a variable power supply that also gave accurate voltage and current measurements. The team therefore were able to adjust the strength of the electromagnet. By doing this it was observed what differences a change in voltage made and also how much current the magnet drew while it was in use.

A variety of Plexiglas thicknesses were used in testing to determine how the electromagnet attached and detached itself from a piece. This was considered for a few reasons. The first reason was that the thickness of the Plexiglas was easier to change rather than the supply voltage for the whole project. The second reason was that if the plexiglass was too thick or thin for the holding force, the piece would not move easily. The actual piece or pieces needed to be tested in order to understand how strong of a magnet the team was dealing with. It would not make sense to test the magnet with one chess piece then use a totally different chess piece, that would've made the test invalid.

Figure 23 below is a flowchart showing the process of the hardware test for the electromagnet implemented for this design. For each of the three tasks, the process was the same. To begin, the desired task would be performed. If the task passed, the test would then move on to the next task. If the task were to fail, it would fail under the category of a hardware fail or a software fail. During a hardware fail, the problem would be diagnosed, and the task would be restarted. During a software fail, the test would be terminated, and the software would be retested according to the test plan. These steps were to be repeated until the electromagnet hardware test was complete.

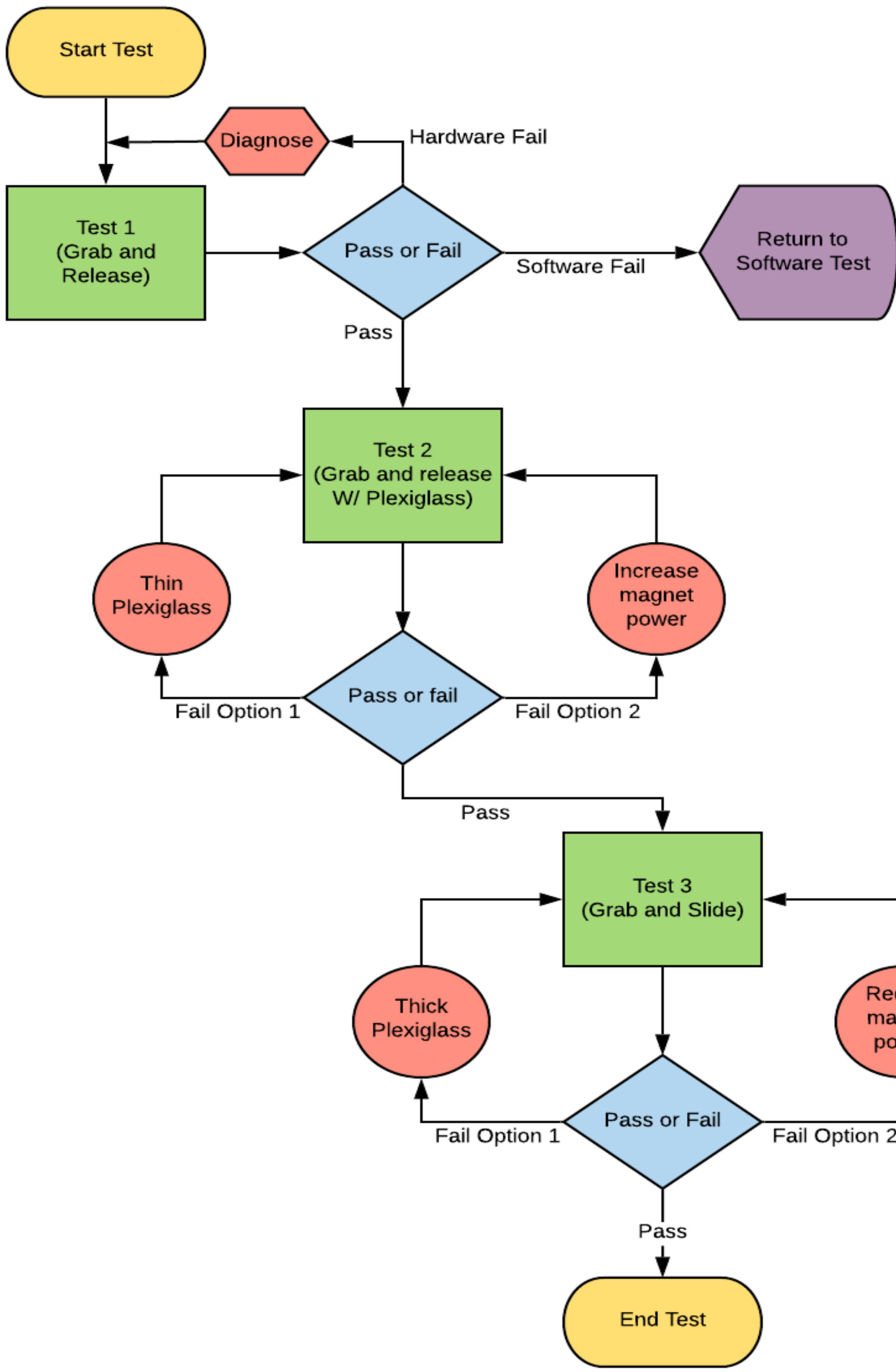


Figure 23: Electromagnet Testing Flowchart.

To begin the testing of the electromagnet, the first task that was assigned consisted of using the electromagnet to grab and release each different type of chess piece considered using for this project. Testing began by attaching the electromagnet to the desired chess piece for a period of one second before releasing the piece and continuing to the next one. By completing this task, the electromagnet was proven that it was capable of operating in the way the team intended it to be designed.

Once the previous task was successfully completed by the electromagnet, the second task would be assigned to the mechanism, further proving that the device was fully capable to function in the intended manner. The second task of the electromagnet hardware test was similar in nature to the first task but with a slight difference. During this task, the electromagnet would be required to grab and releases each different type of chess pieces with a clear plexiglass sheet between the two. The electromagnet would begin by attaching to the desired chess piece for a period of one second before releasing the piece and continuing to the next one. If at any time the electromagnet fails to complete this task, the problem was to be diagnosed, and the task would be restarted.

After completion of the previous two tasks, the final task of this hardware test would take place. During this task, the electromagnet was required to grab each different type of chess pieces with a clear plexiglass sheet between the two. The electromagnet must be able to then slide the piece a distance of 300mm while maintaining a strong hold on the chess piece. If at any time the electromagnet fails to complete this task, the problem was to be diagnosed, and the task would be restarted.

7.2.3 PCB Testing

The first PCB that the team ordered had numerous errors, it wasn't critical but just caused an inconvenience when trying to work with the chip. The wrong size header pins that were placed on the board caused the team to have to attach a 22-gauge wire to each pin header in order to use them. The last reason was that there weren't enough GPIO pins available to control the necessary peripherals; this was caused by a change in the design.

When first trying to program the microcontroller on the PCB, the team decided to go the cheaper route and use the development board as an in-service-programmer ISP. This method was definitely a bad choice because we lost countless hours trying different ways to program it. There were instructions to attach a capacitor in between reset and ground on the development board, some instructions even said to use a resistor. Code was also altered in order to target the correct pins and it still didn't work. The team decided to use the method of using the Atmel-ice programmer. It was the programmer that was needed to specifically program AVR chips and it was used as a last resort because it was costly. After trying to program the chip with the Atmel-ice, the team was finally able to upload code. Testing was done by an LED at a random GPIO pin and seeing if the team was able to make it blink the detaching programmer and observe the continuous blinking.

After finalizing that we were able to upload code and verified that the Schematic wasn't the problem, the team decided to place the order for the final PCB design. There were many changes that were made to the final PCB board but a few of the main changes were; placing the correct size pin headers, using more GPIO pins, grouping and placing parts correctly, and adding more components. When the board finally arrived, the team decided to go ahead and test it by programming it with the Atmel-ice and seeing if an LED could be flashed. It worked without a problem. In the Figure 24 below, both the first PCB design, on the left, and final PCB design, on the right, can be seen and compared.

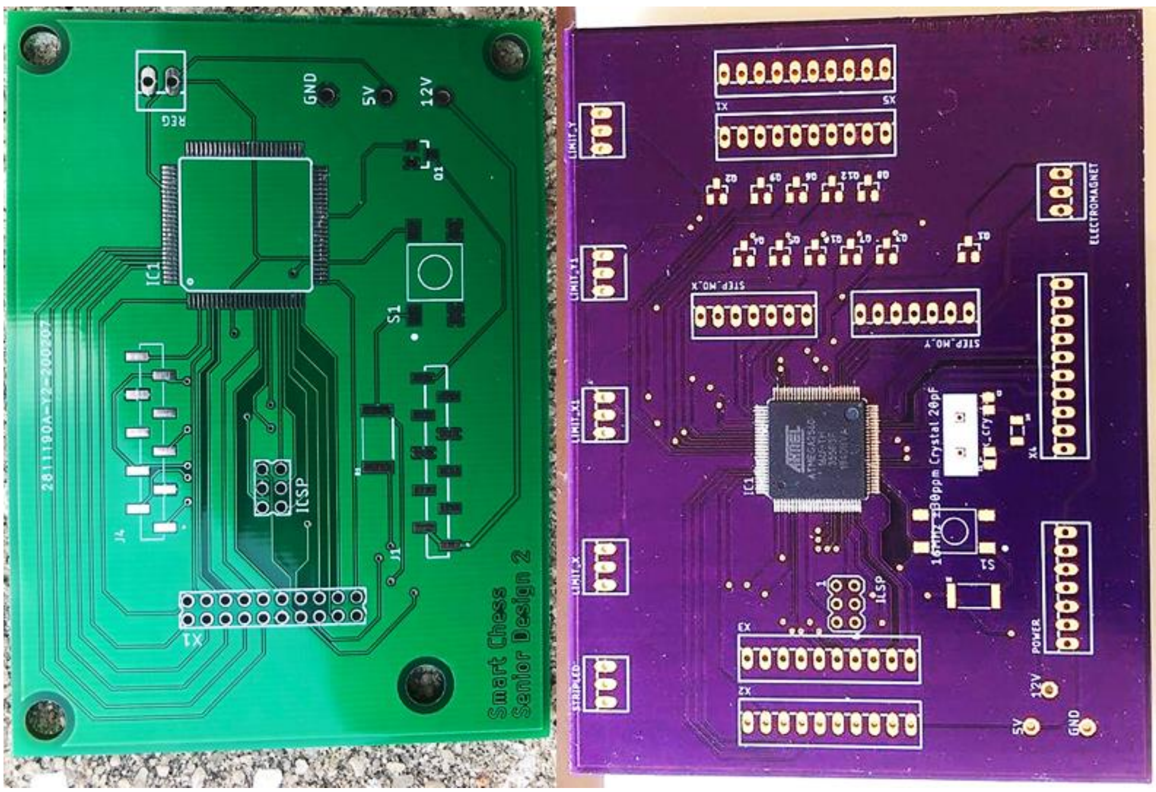


Figure 24: First vs. Final PCB Design.

8. Administrative

To stay organized, focused, and to get things done in a timely matter, without any delays, the team laid out the two semesters of Senior Design I and II and defined the various critical milestones required to successfully complete the project.

8.1 Senior Design I Milestones

Senior Design I focused on the research aspect of the project. Table 27 below states the major responsibilities and assignments due for the project that semester. As a team, the plan was to meet in person at least twice a week to complete certain obligations and discuss what was required of each team member. Outside of those in-person meetings, members were to individually work on the communicated duties necessary to complete certain goals.

Table 27: Senior Design I Milestones.

Expected Due Date	Task
September 20, 2019	Submit Initial Divide and Conquer Document
September 25, 2019	Meet with Dr. Richie to discuss the project idea
October 4, 2019	Submit updated Divide and Conquer Document
November 1, 2019	Submit the 60 Page Draft
November 15, 2019	Submit the 100 Page Draft
November 22, 2019	Begin PCB Design
November 29, 2019	Have the Final Draft of the PCB Design
December 2, 2019	Submit Final Senior Design 1 Documentation
December 2, 2019	Order Designed PCB

8.2 Senior Design II Milestones

Senior Design II concentrated on the development and implementation process of the project. This was a very crucial semester and getting everything done on schedule was top priority. Parts were being ordered from all over the place while testing had to be done for every purchase to make sure that everything worked as planned. There was a possibility of shipment being delayed, and it was because of unforeseen circumstances. Once the parts were delivered, they had to be tested to make sure they're functioning as expected and that took some time as well. Table 28 below states the major tasks and dates estimated and expected for these responsibilities to be completed.

Table 28: Senior Design II Milestones.

Due Date	Task
January 3, 2020	Begin testing individual components
January 6, 2020	Have all parts ordered
January 24, 2020	Have project built
February 14, 2020	Test and make corrections
March 13, 2020	Complete Project
March 20, 2020	Update and Correct Documents
April 3, 2020	Prepare Presentation
April 15, 2020	Final Presentation
April 21, 2020	Submit Final Document

8.3 Budget and Bill of Materials

From the teams' limited knowledge and because the project wasn't sponsored, it was required of the team to do an extensive amount of research and investigating to carefully select the optimal but also the most cost-efficient parts and devices for this project. The cost estimation of this project ended up being roughly $800, considering all major parts and components, as well as miscellaneous items. For that reason of this being just an estimation of the costs, and there was the possibility of error or final changes, there was a great chance that the budget could change. Table 29 below catalogues the main and most essential components required to making this chess board and an a rough estimation of how much they would cost.

Table 29: Cost Estimation.

Item	**Price**
Wood	$50
Plexiglass	$10
Chess Pieces	$20
Microcontroller	$50
Actuators/Servo Motors	$30
Track	$30
Magnets	$50
Batteries/Power Supply	$120
Wires and Components	$20
PCB Manufacturing	$150
LEDs/Cosmetic Lights	$20
Microphone	$50
Speakers	$50
Miscellaneous	$150
Total	**$800**

After a considerable amount of research and comparison, the pros and cons of each component required for this project was evaluated. The research required and done was beneficial in helping the team narrow down certain products and selecting the optimal choice. Of course, there were tradeoffs to some of the certain features considering the restrictions we had but there were some advantages. For example, the XY-Plotter we purchased was approximately $300 but it included the servo motors and drivers. The costs of the components practically ended up balancing out with one another. Table 30 below specifies the Bill of Materials of the most essential components.

Table 30: Bill of Materials.

Item	**Price**	**Quantity**	**Tax & Shipping**	**Subtotal**
Plywood	$35.95	1	$2.52	$38.47
Plexiglass	$29.78	1	$2.08	$31.86
Chess Piece Set	$15	1	-	$15
ATmega 2560 Microchip	$10.24	3	$11.14	$41.86
Generic Sunfounder Development Board	$13.99	1	-	$13.99
XY-Plotter	$299.99	1	-	$299.99
Uxcell Electromagnet	$11.53	1	-	$11.53
Voltage Regulator	$2.20	1	$4.81	$7.01
Standard Power Outlet	$6.86	1	-	$6.86
PCB Manufacturing 1st Design	$20	5	-	$20
PCB Manufacturing 2nd Design	$67.35	3	-	$67.35
LEDs/Cosmetic Lights	$50	2	-	$50
LCD Screen	$14.99	1		$14.99
Miscellaneous	$150	-	-	$150
Total				**$768.91**

8.4 Communication

To effectively succeed and accomplish any team responsibilities, communication is essential. It not only helps each member of the team recognize the goals and objectives of the project and conveys the obligations required from each person, it assists in establishing an understanding and respectful relationship within the group. Our team used a few approaches to communicate with one another, if it were a quick question or need of verification of something regarding the project, a text or a phone call would suffice, but for sharing documents and files, we used Google Drive and Office365. Another crucial means of communications for us were in-person meetings to discuss how far along we were on our tasks.

8.4.1 Google Drive

Google Drive was the first file storage service we used to share our research, files, pictures, and anything pertaining to the project. With a Gmail account, we got access to programs such as Google Docs, Google Slides (which could be useful for Senior Design II), Google sheets, Google Draw, etc. The ability to organize every section into different folders, documents, and files made it very easy to use and navigate our work.

Google Sheets and Google Docs were the two main programs that we used this semester from Google Drive. Google Sheets was used to inventory and document the parts needed and bought, its quantity, and the final costs. Links were also documented in sheets for easy access if it were needed to be referenced back to. This kept our finances organized and structured. Google Docs was used to share the main project report. Any changes that were to be made or new findings were to be acknowledged would be established and shared on there. Furthermore, it was particularly convenient that we were able to label and categorize topics, pictures, and references into certain research documents and folders to easily acquire them when necessary. Google Drive provided a great service but once the document got larger, the formatting kept changing, making it much more difficult to use.

8.4.2 Office365

Microsoft Office365 then became our main method of editing and sharing the report. The OneDrive worked like a cloud and was a file hosting service that was linked to our Knightsmail, so we all had access to it. Not only that, the choice of being able to work on the document on a browser, making it accessible and modifiable with practically any device, was accommodating, but its ability to be opened on any device that had Microsoft Word and consistently and continuously saved and updated the document on the drive made it remarkably convenient. After some time of use, it became the more favorable collaboration-editing tool for the paper. Though it lagged every so often after adding so many more pages, formatting didn’t mess up as much, or at all.

8.4.3 Meetings

To be ahead and on top of our game, communicating while doing individual work was an important aspect of the project, but the most important means of communication and meeting deadlines were in-person meetings. At least twice a week, acknowledging and being understanding of one another's additional responsibilities, we coincided and planned when a good time for everyone was and reviewed and discussed questions, parts, research, and anything related to the project.

Testing parts were also a major portion of these meetings. It was crucial that everyone was clear on what was going on and whether or not the parts were functioning as needed and wanted so we could either progress to the next step of the project or do more analyzing and adjusting to proceed. Especially since everything would be linked and connected to successfully complete this project.

8.5 Project Summary

Throughout the two semesters of Senior Design I and II, there was an abundant amount of time committed to research and documentation, and then finally, design and implementation. From the start, we as a team wanted to do something fun and challenging but original. Because chess is such a classic game that we all play occasionally, we thought it'd be a great idea to somehow implement that into our Senior Design Project. We had so many plans and ideas for the Smart Chess Board, but being a group of all Electrical Engineering students, we had to be realistic and know our limitations.

After a week of research, we realized there were many innovative people in the world and at UCF that had done similar projects that we intended to pursue. A few Senior Design Projects from UCF that we used as a reference for our chess board was Magic Chess, Telepresence Chessboard, Knight Light LED, and Deep RGB. Each chess board designed had something that would benefit us. The research that the other students did help us narrow and make better choices for our selection in parts and products, saving us time on further analysis. But even with all of those projects used to help guide us, it was still very challenging because it incorporated so many aspects of engineering in general.

The design we were aiming towards was a hands-freeand voice activated chess board that included extra features for a more entertaining and aesthetically pleasing product. The deeper we got into the project, the more extensive amount of programming was required to be done and everyone had very little experience. PCB designing and voice recognition was also something that the team wasn't particularly familiar with. The guidance and support from Dr. Ritchie had assisted and encouraged us to proceed and tackle on this idea. We as a team certainly learned an ample amount of information on the procedure to design and implement a plan. Senior Design I required an abundant amount of the research and product selection process and Senior Design II revealed the need of hard work and dedication to successfully succeed in the project. It will serve as a strong backbone to sufficiently approach future projects in the work field and provide an impressive background.

Appendix

References

Acplasticsinc.com. (2019). *What is Lexan?*. [online] Available at: https://www.acplasticsinc.com/informationcenter/r/what-is-lexan [Accessed 27 Nov. 2019].

Adafruit.com. (2019). *Adafruit NeoPixel Digital RGB LED Strip - White 30 LED*. [online] Available at: https://www.adafruit.com/product/1376?length=1 [Accessed 27 Nov. 2019].

Adafruit.com. (2019). *Mini Metal Speaker w/ Wires - 8 ohm 0.5W*. [online] Available at: https://www.adafruit.com/product/1890 [Accessed 27 Nov. 2019].

Adafruit.com. (2019). *RGB LCD Shield Kit w/ 16x2 Character Display - Only 2 pins used!*. [online] Available at: https://www.adafruit.com/product/716 [Accessed 26 Nov. 2019].

Advanced Conversion Technology. (2019). *Linear vs Switching Power Supplies*. [online] Available at: https://www.actpower.com/educational/linear-vs-switching-power-supplies/ [Accessed 27 Nov. 2019].

Aliexpress.com. (2019). *3.95 inch TFT LCD Display Touch Screen 480X320 CH340G Mega 2560 R3 Board for Arduino Replacement Screen*. [online] Available at: https://www.aliexpress.com/item/4000119627174.html [Accessed 26 Nov. 2019].

Arduino.cc. (2019). *Arduino - Introduction*. [online] Available at: https://www.arduino.cc/en/guide/introduction [Accessed 27 Nov. 2019].

atlasRFIDstore. (2019). *What is RFID? | The Beginner's Guide to RFID Systems*. [online] Available at: https://www.atlasrfidstore.com/rfid-beginners-guide/ [Accessed 27 Nov. 2019].

Batteryuniversity.com. (2019). *BU-201: How does the Lead Acid Battery Work?*. [online] Available at: https://batteryuniversity.com/learn/article/lead_based_batteries [Accessed 26 Nov. 2019].

Becker, M. (2010). *Cyanoacrylate: Everything You Need to Know*. [online] Woodworkersjournal.com. Available at:

https://www.woodworkersjournal.com/cyanoacrylate-everything-you-need-to-know/ [Accessed 27 Nov. 2019].

Bing.com. (2019). *What is a Stepper Motor and How it Works?*. [online] Available at: https://www.bing.com/videos/search?q=stepper+motors&view=detail&mid=82E59D26765B416C91A982E59D26765B416C91A9&FORM=VIRE [Accessed 27 Nov. 2019].

BitSophia. (2019). *BitVoicer*. [online] Available at: http://www.bitsophia.com/en-US/BitVoicer/Overview.aspx [Accessed 27 Nov. 2019].

Bitsophia.com. (2013). *BitVoicer 1.2 User Manual English*. [online] Available at: http://www.bitsophia.com/BitVoicer/v12/BitVoicerManual_v12_en.pdf [Accessed 27 Nov. 2019].

Boltdepot.com. (2019). *Wood Screw Diameter and Number Table*. [online] Available at: https://www.boltdepot.com/fastener-information/wood-screws/Wood-Screw-Diameter.aspx [Accessed 27 Nov. 2019].

Brain, M. (2006). *How Lithium-ion Batteries Work*. [online] HowStuffWorks. Available at: https://electronics.howstuffworks.com/everyday-tech/lithium-ion-battery.htm [Accessed 26 Nov. 2019].

Bryant, J. and Bowers, B. (2005). *A Dummies' Guide to Working with Wall Warts*. [online] Dxing.info. Available at: http://www.dxing.info/equipment/wall_warts_bryant.dx [Accessed 27 Nov. 2019].

Burrus, W. (2019). *Titebond*. [online] Titebond.com. Available at: http://www.titebond.com/community/the-big-three [Accessed 27 Nov. 2019].

Cbenson. (2012). *Arduino Microcontroller Feature Comparison*. [online] Available at: https://www.robotshop.com/community/blog/show/arduino-microcontroller-feature-comparison [Accessed 27 Nov. 2019].

Chess.com. (2019). *Chess Board Dimensions* | *Basics and Guidelines*. [online] Available at: https://www.chess.com/article/view/chess-board-dimensions#standards [Accessed 27 Nov. 2019].

Chessprogramming.org. (2019). *Mailbox*. [online] Available at: https://www.chessprogramming.org/Mailbox [Accessed 27 Nov. 2019].

Cool Magnet Man. (2017). *Electromagnets*. [online] Available at: https://www.coolmagnetman.com/magelect.htm [Accessed 27 Nov. 2019].

Core Electronics. (2016). *How to Use Breadboards - Tutorial*. [online] Available at: https://core-electronics.com.au/tutorials/how-to-use-breadboards.html [Accessed 27 Nov. 2019].

Dap.com. (2019). *Plastic Wood Latex Wood Filler—Squeeze Tube - DAP*. [online] Available at: https://www.dap.com/dap-products-ph/plastic-wood-latex-wood-filler-squeeze-tube/ [Accessed 27 Nov. 2019].

Das, S. (2019). *Types of PCB* | *Different Types of Printed Circuit Board (PCB)*. [online] Electronics and You. Available at: http://www.electronicsandyou.com/blog/types-of-pcb-different-types-of-printed-circuit-board-pcb.html [Accessed 27 Nov. 2019].

Developer.amazon.com. (2019). *Alexa Skills Kit SDKs*. [online] Available at: https://developer.amazon.com/docs/sdk/alexa-skills-kit-sdks.html [Accessed 27 Nov. 2019].

Developer.amazon.com. (2019). *Python Tutorial: Learn with Our Alexa Python Programming Tutorial*. [online] Available at: https://developer.amazon.com/en-US/alexa/alexa-skills-kit/resources/training-resources/alexa-skill-python-tutorial [Accessed 27 Nov. 2019].

Deziel, C. (2017). *What Types of Plywood Can Be Used Outdoors?*. [online] hunker.com. Available at: https://www.hunker.com/13401518/what-types-of-plywood-can-be-used-outdoors [Accessed 27 Nov. 2019].

DigChip. (2019). *Details, datasheet, quote on part number: LPC2148*. [online] Available at: https://www.digchip.com/datasheets/1113523-lpc2148.html [Accessed 27 Nov. 2019].

Digchip.com. (2017). *LPC2148FBD64 datasheet*. [online] Available at: https://www.digchip.com/datasheets/parts/datasheet/364/LPC2148FBD64-pdf.php [Accessed 27 Nov. 2019].

Distributors, J. (2019). *Wood Screw Size Chart*. [online] Jamestowndistributors.com. Available at: https://www.jamestowndistributors.com/userportal/document.do?docId=270 [Accessed 27 Nov. 2019].

DIYElectronics. (2019). *Stepper Motors*. [online] Available at: https://www.diyelectronics.co.za/store/97-stepper-motors [Accessed 27 Nov. 2019].

Eeherald.com. (2016). *Embedded Systems Course- module 17: Basics of LCD and it's Interface to a microcontroller*. [online] Available at:

http://www.eeherald.com/section/design-guide/esmod17.html [Accessed 26 Nov. 2019].

Electronics-notes.com. (2019). *Series Voltage Regulator : Series Pass Regulator*. [online] Available at: https://www.electronics-notes.com/articles/analogue_circuits/power-supply-electronics/linear-psu-series-regulator-circuit.php [Accessed 27 Nov. 2019].

Electronics-notes.com. (2019). *Shunt Voltage Regulator : Shunt Regulator*. [online] Available at: https://www.electronics-notes.com/articles/analogue_circuits/power-supply-electronics/linear-psu-shunt-regulator-circuit.php [Accessed 27 Nov. 2019].

Electronics Tutorials. (2019). *Hall Effect Sensor*. [online] Available at: https://www.electronics-tutorials.ws/electromagnetism/hall-effect.html [Accessed 27 Nov. 2019].

ElectronicsTutorials. (2019). *Switch Mode Power Supply*. [online] Available at: https://www.electronics-tutorials.ws/power/switch-mode-power-supply.html [Accessed 27 Nov. 2019].

ElProCus - Electronic Projects for Engineering Students. (2019). *ARM7 Based (LPC2148) Microcontroller Pin Configuration*. [online] Available at: https://www.elprocus.com/arm7-based-lpc2148-microcontroller-pin-configuration/ [Accessed 27 Nov. 2019].

ElProCus - Electronic Projects for Engineering Students. (2019). *Different Types of Voltage Regulator and Working Principle*. [online] Available at: https://www.elprocus.com/types-of-voltage-regulators-and-working-principle/ [Accessed 27 Nov. 2019].

ElProCus - Electronic Projects for Engineering Students. (2019). *Various Kinds Of Microcontroller Boards and Its Applications*. [online] Available at: https://www.elprocus.com/different-types-of-microcontroller-boards/ [Accessed 27 Nov. 2019].

En.wikipedia.org. (2019). *Radio-frequency identification*. [online] Available at: https://en.wikipedia.org/wiki/Radio-frequency_identification#Frequencies [Accessed 27 Nov. 2019].

En.wikipedia.org. (2019). *Shannon number*. [online] Available at: https://en.wikipedia.org/wiki/Shannon_number [Accessed 27 Nov. 2019].

En.wikipedia.org. (2019). *USB*. [online] Available at: https://en.wikipedia.org/wiki/USB [Accessed 26 Nov. 2019].

En.wikipedia.org. (2019). *USB hardware*. [online] Available at: https://en.wikipedia.org/wiki/USB_hardware [Accessed 26 Nov. 2019].

En.wikipedia.org. (2019). *Voltage regulator*. [online] Available at: https://en.wikipedia.org/wiki/Voltage_regulator [Accessed 27 Nov. 2019].

Energyusecalculator.com. (2019). *Electricity usage of a Printer*. [online] Available at: http://energyusecalculator.com/electricity_printer.htm [Accessed 27 Nov. 2019].

Flexfire LEDs. (2019). *ColorBright™ RGB 150 Color Changing LED Strip Light*. [online] Available at: https://www.flexfireleds.com/150-led-rgb-color-changing-strip-light-by-the-foot/ [Accessed 27 Nov. 2019].

Ford, R. and Coulston, C. (2008). *Design for electrical and computer engineers*. Boston, Massachusetts: McGraw-Hill Custom Publishing, pp.36 and 39.

Gaillard, F. and Eieland, A. (2019). *Atmel*. [online] Microchip. Available at: http://ww1.microchip.com/downloads/en/DeviceDoc/MCU_vs_MPU_Article.pdf [Accessed 27 Nov. 2019].

Glass Doctor. (2019). *What Is Plexiglass and What Is It Made Of?*. [online] Available at: https://glassdoctor.com/blog/what-is-plexiglass-and-what-is-it-made-of [Accessed 27 Nov. 2019].

Guy, B. (2016). *What are zinc-carbon batteries?*. [online] Batteryguy.com. Available at: https://batteryguy.com/kb/knowledge-base/what-are-zinc-carbon-batteries/ [Accessed 26 Nov. 2019].

Hartikka, L. (2019). *A step-by-step guide to building a simple chess AI*. [online] freeCodeCamp.org. Available at: https://www.freecodecamp.org/news/simple-chess-ai-step-by-step-1d55a9266977/ [Accessed 27 Nov. 2019].

Heath, D. (2019). *6 Best Wood Glue Reviews: Extra Strong Glue for Woodworking & Hobbies*. [online] MorningChores. Available at: https://morningchores.com/best-wood-glue/ [Accessed 27 Nov. 2019].

Hewes, J. (2019). *Stripboard*. [online] Electronicsclub.info. Available at: https://www.electronicsclub.info/stripboard.htm [Accessed 27 Nov. 2019].

Inkscape.org. (2019). *Features of Inkscape*. [online] Available at: https://inkscape.org/about/features/ [Accessed 27 Nov. 2019].

Jameco.com. (2019). *How Servo Motors Work* | *Servo Motor Controllers*. [online] Available at: https://www.jameco.com/Jameco/workshop/howitworks/how-servo-motors-work.html [Accessed 27 Nov. 2019].

Jim.sh. (2019). *jim - Revision 6494: /vendor/microwindows/current/src/demos/tuxchess*. [online] Available at: https://jim.sh/svn/jim/vendor/microwindows/current/src/demos/tuxchess/ [Accessed 27 Nov. 2019].

Katz, A. and Ross, E. (2019). *Minimax*. [online] Brilliant.org. Available at: https://brilliant.org/wiki/minimax/ [Accessed 27 Nov. 2019].

Keith, J. (2019). *Of Wall Warts & Wall Transformers*. [online] Electro Schematics. Available at: https://www.electroschematics.com/of-wall-warts-wall-transformers/ [Accessed 27 Nov. 2019].

Kitchencabinetkings.com. (2019). *What is Wood Filler?*. [online] Available at: https://kitchencabinetkings.com/glossary/wood-filler/ [Accessed 27 Nov. 2019].

Kollmorgen.com. (2019). *Servo Motors*. [online] Available at: https://www.kollmorgen.com/en-us/products/motors/servo/servo-motors/ [Accessed 27 Nov. 2019].

Loctiteproducts.com. (2019). *Loctite® Clear Silicone Waterproof Sealant*. [online] Available at: https://www.loctiteproducts.com/en/products/fix/contact-adhesives/loctite_clear_siliconewaterproofsealant.html [Accessed 27 Nov. 2019].

Magnetpartner.com. (2019). *What Is a Temporary Magnet? Complete List Of Temp Magnets*. [online] Available at: https://magnetpartner.com/blog/what-is-a-temporary-magnet [Accessed 26 Nov. 2019].

Mag-Tools. (2019). *Magswitch Magnetic Technology Products*. [online] Available at: https://mag-tools.com/pages/about-us [Accessed 26 Nov. 2019].

Melendez, S. (2018). *The Functions of a Breadboard*. [online] Techwalla. Available at: https://www.techwalla.com/articles/the-functions-of-a-breadboard [Accessed 27 Nov. 2019].

Micro-Max Minimalist Chess. (2019). *Micro-Max, a 133-line Chess Source*. [online] Available at: https://home.hccnet.nl/h.g.muller/max-src2.html [Accessed 27 Nov. 2019].

Microchip. (2019). *Atmel Datasheet*. [online] Available at: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf [Accessed 27 Nov. 2019].

Microchip. (2019). *PIC18F46K80 - Microcontrollers and Processors*. [online] Available at:

https://www.microchip.com/wwwproducts/en/PIC18F46K80#additional-features [Accessed 27 Nov. 2019].

Microchip. (2019). *Microchip PIC18F66K80 FAMILY*. [online] Available at: http://ww1.microchip.com/downloads/en/devicedoc/pic18f66k80%20family%20enhanced%20flash%20mcu%20with%20ecan%20xlp%20technology%2030009977g.pdf. [Accessed 27 Nov. 2019].

Mirza, U. (2019). *Types of Batteries and Their Applications*. [online] Brighthubengineering.com. Available at: https://www.brighthubengineering.com/power-generation-distribution/123909-types-of-batteries-and-their-applications/ [Accessed 26 Nov. 2019].

Pan, W. (2015). *5 Types of Wood Glue*. [online] Bluprint. Available at: https://www.mybluprint.com/article/5-types-of-wood-glue-what-to-know-how-to-use-them [Accessed 27 Nov. 2019].

Pcbcart. (2019). *Multilayer PCB Benefits and Applications*. [online] Available at: https://www.pcbcart.com/article/content/multilayer-PCB-benefits-and-applications.html [Accessed 27 Nov. 2019].

Philips, G. (2019). *How Does RFID Technology Work?*. [online] MakeUseOf. Available at: https://www.makeuseof.com/tag/technology-explained-how-do-rfid-tags-work/ [Accessed 27 Nov. 2019].

Rouse, M. and Gibilisco, S. (2012). *USART (Universal Synchronous/Asynchronous Receiver/Transmitter)*. [online] WhatIs.com. Available at: https://whatis.techtarget.com/definition/USART-Universal-Synchronous-Asynchronous-Receiver-Transmitter [Accessed 26 Nov. 2019].

Sapru, V. (2019). *Battery Power Online | The World of Alkaline Batteries*. [online] Batterypoweronline.com. Available at: https://www.batterypoweronline.com/markets/batteries/the-world-of-alkaline-batteries/ [Accessed 26 Nov. 2019].

Seeedstudio.com. (2019). *Choosing the Right Motor for Your Project – DC vs Stepper vs Servo Motors*. [online] Available at: https://www.seeedstudio.com/blog/2019/04/01/choosing-the-right-motor-for-your-project-dc-vs-stepper-vs-servo-motors/ [Accessed 27 Nov. 2019].

Shaddy, W. (2019). *How to Calculate the Weight of Plywood*. [online] hunker.com. Available at: https://www.hunker.com/13401564/how-to-calculate-the-weight-of-plywood [Accessed 27 Nov. 2019].

Shmyrev, N. (2019). *Frequently Asked Questions (FAQ)*. [online] CMUSphinx Open Source Speech Recognition. Available at: https://cmusphinx.github.io/wiki/faq/ [Accessed 27 Nov. 2019].

Skycraftsurplus.com. (2019). *Speaker 2 Watt 8 Ohm Full Range*. [online] Available at: https://skycraftsurplus.com/speaker-2-watt-8-ohm-full-range-12in-2w-8ohm-rp1738.html [Accessed 27 Nov. 2019].

Stockfishchess.org. (2019). *Stockfish 10*. [online] Available at: https://stockfishchess.org/ [Accessed 27 Nov. 2019].

Store.arduino.cc. (2019). *Arduino Mega 2560 Rev3*. [online] Available at: https://store.arduino.cc/usa/mega-2560-r3 [Accessed 27 Nov. 2019].

Tckerrigan.com. (2019). *TSCP - Tom Kerrigan's Home Page*. [online] Available at: http://www.tckerrigan.com/Chess/TSCP/ [Accessed 27 Nov. 2019].

Teel, J. (2019). *Linear and Switching Voltage Regulators - An Introduction*. [online] PREDICTABLE DESIGNS. Available at: https://predictabledesigns.com/linear-and-switching-voltage-regulators-introduction/ [Accessed 27 Nov. 2019].

Texas Instruments. (2019). *MSP430FR6989*. [online] Available at: http://www.ti.com/product/MSP430FR6989 [Accessed 27 Nov. 2019].

Texas Instruments. (2019). *MSP430G2553*. [online] Available at: http://www.ti.com/product/MSP430G2553/technicaldocuments [Accessed 27 Nov. 2019].

ThePlywood.com. (2019). *Why Use Sande Plywood*. [online] Available at: http://theplywood.com/sande [Accessed 27 Nov. 2019].

Thomasnet.com. (2019). *Types of Magnets*. [online] Available at: https://www.thomasnet.com/articles/electrical-power-generation/types-of-magnets/ [Accessed 26 Nov. 2019].

Thornton, S. (2017). *Display options for MCUs: LCD, LED, and OLED*. [online] Microcontrollertips.com. Available at: https://www.microcontrollertips.com/display-options-mcus-lcd-led-oled/ [Accessed 26 Nov. 2019].

Titze, I. (1994). *Principles of voice production*. Englewood Cliffs, N.J.: Prentice Hall, p.188.

Tutorialspoint.com. (2019). *Bottom Up Testing*. [online] Available at: https://www.tutorialspoint.com/software_testing_dictionary/bottom_up_testing.htm [Accessed 27 Nov. 2019].

Usplastic.com. (2019). *How much does acrylic sheet weigh?*. [online] Available at: https://www.usplastic.com/knowledgebase/article.aspx?contentkey=884 [Accessed 27 Nov. 2019].

Woodford, C. (2018). *Glass*. [online] Explain that Stuff. Available at: https://www.explainthatstuff.com/glass.html [Accessed 27 Nov. 2019].

Woodworkbasics.com. (2019). *Woodwork - PVA Glue Information*. [online] Available at: http://www.woodworkbasics.com/pva.html [Accessed 27 Nov. 2019].

Zaifrun, M. (2010). *Creating a chess engine from scratch (Part 1: Basics)*. [online] Chess.com. Available at: https://www.chess.com/blog/zaifrun/creating-a-chess-engine-from-scratch-part-1 [Accessed 27 Nov. 2019].

Content Authorization

Request for Permission to Use Copyrighted Images

LH Lauri Hartikka <lhartikk@gmail.com>
Thu 10/31/2019 4:51 PM
To: Noel Membribe

Yeah you can use the images any way you like

Br
Lauri

...

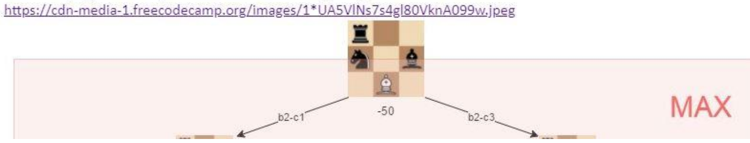
NM Noel Membribe
Thu 10/31/2019 4:22 PM
lhartikk@gmail.com

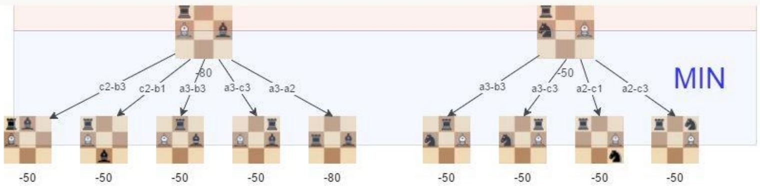
Hello Mr. Hartikka,

I am a student at the University of Central Florida, and I am a part of a team of students who are building a Voice-Activated Magnetically Controlled Chess Board for our Senior Design project. We would like to use some of your images for documentation. Our report will be published on the UCF website upon its completion (http://eecs.ucf.edu/seniordesign/). I was hoping you might be able to assist me in obtaining permission for reproduction.

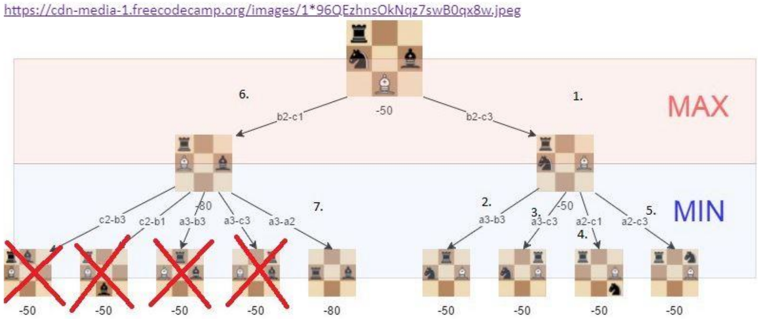
The images in question are part of your article "A step-by-step guide to building a simple chess AI" hosted on freeCodeCamp at (https://www.freecodecamp.org/news/simple-chess-ai-step-by-step-1d55a9266977/).

We are hoping to obtain permission for the following images:
https://cdn-media-1.freecodecamp.org/images/1*UA5VlNs7s4gl80VknA099w.jpeg





https://cdn-media-1.freecodecamp.org/images/1*96QEzhnsOkNqz7swB0qx8w.jpeg



Please let me know if this is possible. Feel free to direct me to someone more suitable if you are not the proper point of contact.

Thanks in advance for your help,

Noel Membribe