Wizard's Chess

Ben Baileys, Delany Bolton, Fabio Soares,

Luke Tholen, and Shae Watkins

EE Senior Design Fall 2019

Table of Contents

1. Introduction

Imagine you are an eleven year old child. You open the mail to see that you have just received your letter informing you that you've been accepted into Hogwarts School of Witchcraft and Wizardry. It is the best day of your life; not only have you learned that magic is real and you possess this gift, but also because you no longer have to play with a run of the mill boring chessboard. Now you can play with Wizard's Chess! Wizard's Chess entails a chessboard with pieces that have the capability to respond to your voice commands. It is a revolutionary way to play chess! Now imagine: sadly you did not receive a letter from Hogwarts outlining your acceptance when you were eleven. Instead you grew up in the muggle world, studied hard, and attended the University of Notre Dame to become an esteemed Electrical Engineer. Senior Design rolls around and you are granted the unique opportunity to create your own version of Wizard's Chess. Do you take the chance? Yes! The following document outlines the high level design for muggle-world Wizard's Chess.

2. Problem Statement and Proposed Solution

Chess is the ultimate strategy game. It has been played for centuries all over the world. There are countless online versions, tournaments, and styles of chess that make it a rich and varied game. There are many different automated games on the market, however automated real world chess is an underutilized market. This project proposed to fully realize automated voice controlled chess that would enable players to experience the magic of playing chess without having to lift a finger. With the intent of combining knowledge acquired throughout several years of electrical engineering, this project will design and build a coreXY cartesian motion platform that will facilitate the movement of chess pieces across the board. The pieces will move through connection with an electromagnet, thus enabling piece selection and diagonal movement. In addition to this mechanism, this project will implement a voice recognition system, not unlike Amazon's Alexa, or Google Home, in order to give players a way to control the movement of the pieces. Finally this project will use software to implement a computerized version of chess game play which will serve two purposes. The first will be a method by which to track gameplay and piece position so that the electromagnet can move pieces accurately and legally within the confines of the rules of chess. The second purpose is a single player capability. The chess software will be able to enact a computerized AI which can play the game against an individual player. That way the project will have the ability to allow for multiple modes of gameplay, and will implement many modern gameplay features.

3. System Requirements

The following are the system requirements for each of the major parts of the Wizard Chess design.

Motors

- Power: Must be able to draw power from a wall outlet. Will require outlet adapters and cords. The motors require 12V so we will need to
- Torque: Each motor must prove enough torque to turn the coreXY pulley system, but not too much torque as to break the timing belts.
- Speed: Each motor must provide enough speed to move the electromagnet carriage fast enough to make gameplay happen in real time, an average of five seconds per piece move.

CoreXY

- Accuracy: Must be able to move pieces across the chessboard with accuracy(4 sq. in. grid squares), as well as move pieces in between other pieces(the knight).
- Robustness: Must be strong enough to withstand the tension strain imparted by the timing belt and the weight of the board placed atop.
- Timing Belt: Must be tight enough to effectively pull the electromagnet carriage around the board. It also must maintain this tension over time so the timing belt does not separate from the pulley system and ruin the ability to use the chessboard.
- Wheels: The wheels must be the correct size to fit into our t-slotted aluminum railing, and they must have enough friction as to not slip when in use.
- Y-grid movers: These 3-D printed pieces need to have enough durability to withstand the tension of the pulley system, as well as enough mass to support the y-grid guide railing.

Voice Recognition

- Accuracy: Must be able to accurately interpret a broad range of voice profiles.
- Analysis: Must be able to analyze the voice for common chess commands and respond accordingly.

Chess Game Tracker

- Accuracy: Must be able to keep track of all game movements so to know where all pieces are on the board as well as know which moves would be considered illegal.
- Interfacing: Must be able to interface with both the voice recognition system and the motors to both interpret and respond to the commands given.

Electromagnet

- Range: Must have a small enough magnetic signature to be able to move one piece at a time while not affecting the placement of others on the board.
- Polarity: Must be able to effectively swap polarities in order to move both sides of chess pieces(they will have different polarities).
- Strength: Must be strong enough to maintain magnetic connection with the chess piece through the wooden board as it is also moving.

Microphone

- Range: Must have a range of at least 5 feet in order to hear commands given at normal distances.
- Power: Must be able to derive power from the wall outlet.

Physical Build

- Weight: Must be light enough as to make transport fesabile, if slightly cumbersome.
- Robustness: Must withstand natural wear and tear and normal indoor gameplay conditions.
- Size: Must be large enough for the pieces to move effectively but small enough to easily be moved by two people.

System Block Diagram

4.1 Overall System:

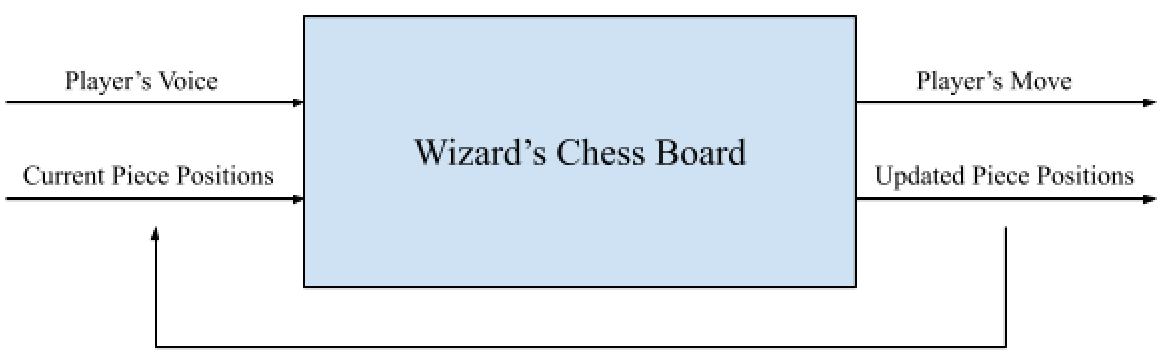

Fig. 1 - Overall System Inputs and Outputs

As shown in Figure 1, Wizard's Chess will have two inputs, the player's voice and the current piece positions and status of gameplay. The outputs will then be the updated piece positions on the board as well as the physical move of the player's desired piece.

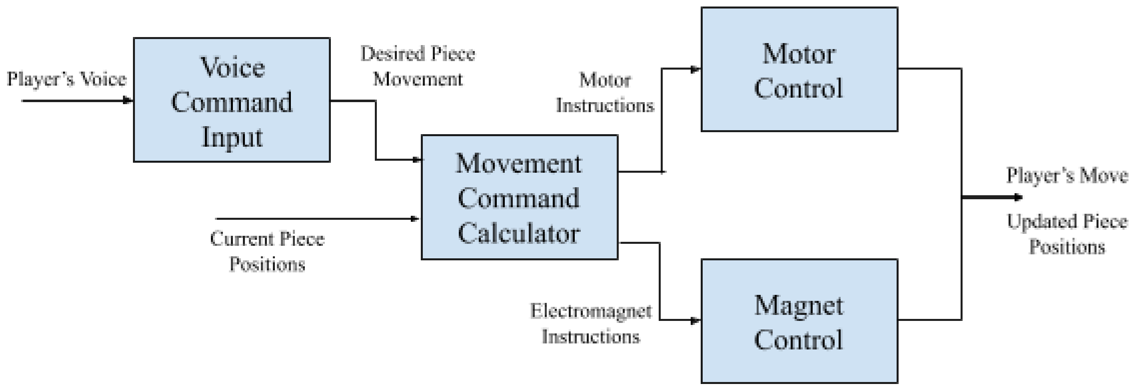

Fig 2. - Overall System Block Diagram

The second figure shows the overall system broken down into several subsystems. The voice command input and the movement command calculator take in the user's voice, interpret the voice as a chess command, and compares the command to the current piece positions in order to determine where the specified piece needs to be moved and if an opponent piece needs to be removed from the board. These commands are then executed by two subsystems controlling the physical motion of the pieces; a motor control subsystem to move the electromagnet along the calculated path and an electromagnet control system to control when to move and when to release a piece. Each of these subsystems is broken down in the following section.

4.2 Subsystem and Interface Requirements:

4.2.1.Voice Command Input

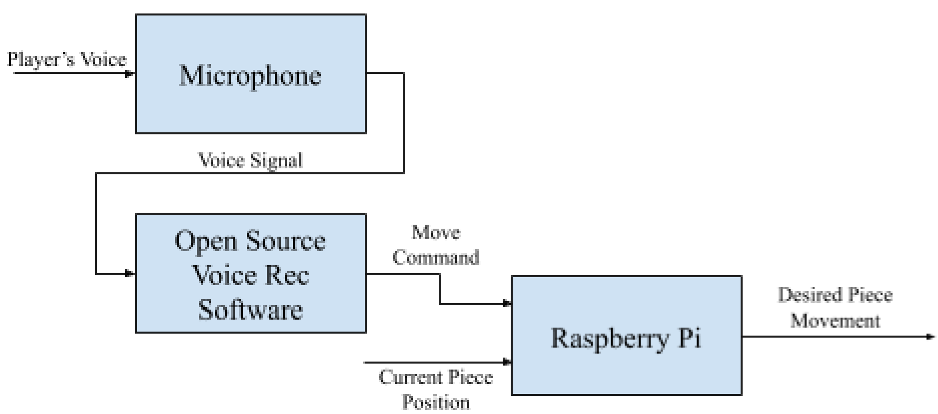

Fig 3. - Command Input System

The voice command input subsystem utilizes a microphone to detect a user's voice and then converts the air compressions into an electrical signal. This signal is sent to an open source voice recognition software that processes the electrical signal to determine what the user had said and utilize this information to determine what commands to send to the Raspberry Pi. Move commands are sent from the software to the Raspberry Pi where a software is tracking piece locations and legal movements. The Raspberry Pi uses both current piece position and move commands to transport pieces around the board and remove pieces from the board when they are overtaken, this is done as an output from the Raspberry Pi.

4.2.2 Movement Command Calculator

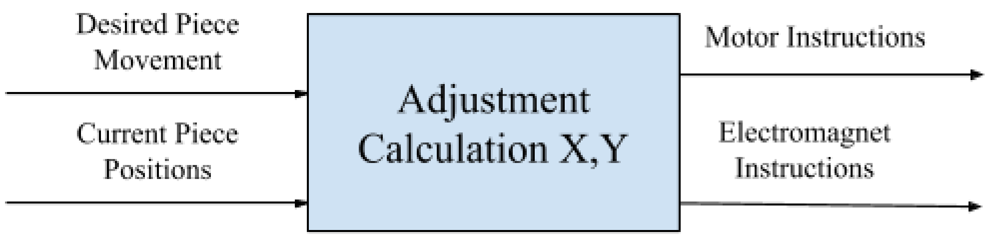

Fig 4. - Movement Command Software

The software to control the calculation of the piece movement will take the desired movement input from the Raspberry Pi after it has been interpreted from the voice command. It will also receive the current piece position from the chess tracker software. With these two inputs it will be able to calculate the required stepper motor instruction to move the piece to the correct location. Additionally, the calculation will determine the electromagnet instructions in order to facilitate the physical movement of the piece. These instructions will include the required polarity of the magnet, and the duration of its engagement.

4.2.3 Motor Control

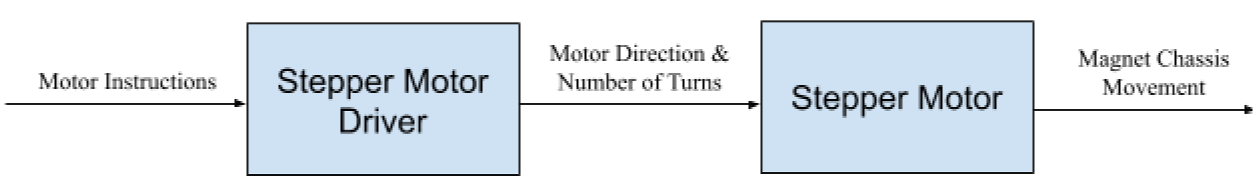

Fig 5. - Stepper Motor Control System

The motor control subsystem will take the stepper motor commands as inputs into the stepper motor drivers. The drivers will then translate that into the motor direction, number of turns, and motor speed that will accomplish the required instructions.

Additional Requirements:

- Power: 12V power supply for the motors

4.2.4 Magnet Control

Fig 6. - Electromagnet Control System

The electromagnet is moved beneath the board by the stepper motors in the CoreXY configuration. The electromagnet polarity/duration instructions from the movement command subsystem will be an input to the driver which will then interpret and send the on/off signal to the electromagnet. The magnet will then physically move the chess pieces across the board to their desired final location.

Additional Requirements:

- Power: A power supply is required for the electromagnet driver and magnet itself.

4.3 Future Enhancement Requirements

This section outlines possible future enhancements to the Wizard's Chess Board.

Enhancement 1: Single Player Mode

The first potential enhancement would be the ability to have a single player game mode. Single player gameplay requires sufficient software to emulate chess game play and the ability for the computer to not just track gameplay, but implement strategy and play the game. It would also require a speaker to play the computer's projected move to the player. This would draw heavily on the voice recognition subsystem and would just develop it so it could take inputs and outputs, like Alexa or Siri.

Enhancement 2: Wifi Game Play

Raspberry Pis have their own built in wifi module. This in theory, would enable players to play chess with this board in remote locations away from the board, as long as they have internet connection. To fully realize this, it would be beneficial to have an app so that the player could keep visual track of the game and board layout, as well as facilitating ease of move input.

Enhancement 3: Identification of Pieces

As of now, our idea is to keep track, through software, of all the movements done by each chess piece and its player. In other words, we start the game with the correct positions and for every play, we update the state/location of where each piece is located. However, and a possible enhancement, is if we could physically identify each piece (e.g. using NFC or RFID). This would allow us to, for example, restart the game repositioning each piece in their correct spot.

5. High Level Design Decisions

5.1 - X-Y Grid

- **T-Slot Aluminum Frame Pieces**
  - Cost Efficient: These supports are lightweight and relatively inexpensive. The exact piece that will be used is the 25-2525: 25mm X 25mm T-Slotted Profile with four open T-Slots.
  - Modular: These are easy to assemble, with the T-Slot shape we were able to get compatible mounting pieces. We will use the PZRT 2020 Series aluminum profile connector set, which includes 20pcs corner bracket, 40pcs M5 x 10mm T-slot nuts, 40pcs M5x10mm hex socket cap screw bolt for 6mm slot aluminum profile accessories. These can accomplish all the necessary mounting to the frame.
- **Plywood Base**
  - Durable: Cheap base used to support the frame and for mounting the stepper motors. We were not too particular about this piece, it was cut to size for the board so dimensions were not a problem and it is easy to get at any hardware store.
- **3-D Printed Parts**
  - Made to Order: We printed some mounting pieces to move along our frame. They are also able to be printed immediately down the hall in Stinson-Remmick as fast as we can draw up schematics in Autodesk.
  - Iterative: Since we printed them we are able to adjust and shape them multiple times until we get the perfect fit for our unique requirements. This is also included in tuition cost so it will not contribute to the budget.

5.2 - Movement Command Calculator

- **Raspberry Pi**
  - Ability: The raspberry pi is capable of running a program such as the movement command calculator.
  - The raspberry pi can communicate with the stepper drivers and the electromagnet driver. Since a raspberry pi is capable of doing a wide range of processes it should be easy to communicate between the drivers and the pi.
  - Available: There is also a raspberry pi available for use so it will not contribute to the cost of the project.
  - There is also a wide library of example code for the raspberry pi since it is a widely used device. This will help with learning how to use the device.

5.3 - Voice Control

- **Microphone**
  - Compatible: The microphone we choose will need to work with our Raspberry Pi controller as well as our voice-control software.
  - We can start with a simple breadboard microphone for initial concept development and move on to including a microphone in our board design but the breadboard option is ideal for testing and working out the kinks early in project.
- **Open Source Voice Command Software**
  - Our chosen software will need to run on a Raspberry Pi controller. Amazon Alexa software can be run on a raspberry pi using an amazon developer account. With this data we hope to filter voice inputs only to chess moves and relay that data to the drivers.

5.4 - Motor Control

- **Stepper Motor Nema 17 Bipolar 40mm**
  - We chose this because it is a bipolar stepper motor that is able to run off of low voltages.
  - Capable: This motor is also small, lightweight, and capable of moving the pulleys required for our design. The motor will need to move the pulley configuration with the electromagnet attached so we chose one with a high enough torque, step angle, and steps per revolution for the smoothness and accuracy needed for our design.
  - Compatible: The pulleys we are using are mountable with compatible shaft diameter and hole diameter. With a shaft diameter of 5mm, the pulleys slide onto the steppers nicely and have a set screw to secure them in place.
- **Raspberry Pi Motor Driver**
  - Chosen because it is simple to program and easy to understand. Since it is a sparkfun product, the data sheets are all in a known location and we can reference them if we have any questions or concerns. We can also reference these if we

choose to construct our own motor driver boards. By using theirs as a guide we can use what we need from sparkfun's driver and make adaptations if needed for our specific purpose.
  - Resources: Test programs for this driver are also available to help with the learning curve. Since it is an already developed driver board, people have made a range of test program to illustrate its functionality. Building off of these examples we can run our specific movement commands with ease.
  - Adaptable: This driver is also able to run off an arduino or raspberry pi for ease of use.

5.5 - Magnet Control

- **Electromagnet Driver**
  - Heat Dissipation: The driver will need to control high levels of current for sustained amounts of time so it will need to be able to handle the amount of heat generated while the magnet is on.
  - Connectivity: The driver for the electromagnet should be compatible to the raspberry pi just as the stepper driver was. This will make setup and testing easier and allow for troubleshooting using the raspberry pi.
- **Electromagnet**
  - Voltage: The electromagnet is able to run on the same voltage as the raspberry pi or arduino.
  - Size: The magnet is small enough to mount on our x-y axis without adding critical weight. It will also have to mount to our 3-D printed centerpiece which may require some modification to the centerpiece. This should not be too difficult since the centerpiece is 3-D printed and can be redesigned.
  - Power: The electromagnet will have to have a balance between too much and too little magnetic force. Because the electromagnet will have to be strong enough to move a chess piece but delicate enough to move ONLY one chess piece, we will have to research tuning the electromagnet in order to achieve this delicate balance.

6. Open Questions

Microphone:

Related with the microphone, we are still figuring it out the best approach. As of know, we intend to use some sort of external component that can be used with our Raspberry Pi. On a quick search, we can find cheap parts such as "Mini USB Microphone" with price around $5.95

and “USB Microphone for Raspberry Pi Mic Portable With Holder Free Drivers Black” with price around $2.

Voice Control:

Related with the voice command that the user will use to move their chess pieces we intend to use Amazon Alexa. The reason why we choose Alexa for this first approach is because they have their own library and several tutorials on how to train a voice model. With this model, Alexa can recognize a voice command and triggers certain functions. Furthermore, Alexa is easily integrated with Raspberry Pi .

Chess Software:

Related with the software, we will need to create some sort of chess game. The game itself should not take long as there are many tutorials online on how to implement chess in Python. However, and most important, it is important that the chess game keep track of all the movements on the physical chess board. This will be more complicated as the code will have to receive an input from the user (e.g. move king to A3) and change some sort of logic matrix which will keep track of all the pieces of each player. Furthermore, this code will have to communicate, back and forth, with the step motors sending instructions to where each physical position should be moved.

Piece Movement:

Related with the physical movement of the chess pieces, we are hoping to use some sort of proximity sensor with normally open contacts change states when a magnetic field is applied. With this, the sensors act as non-latching electrical switches. One of the devices that we can use is called “Electric Lifting Magnet Electromagnet Solenoid Lift Holding” with the price around $12. One of the challenges that we anticipate with this approach is that we are not sure if this magnetic will only move one piece per time or it will also impact the surrounding pieces. One of the ideas to overcome this problem is to make the chess board bigger, with chess pieces spread out.

7. Major Component Costs

The following table outlines the costs of the man components for the Wizard's Chess Board:

Description	Part Name	Quantity	Cost per Item	Total Cost
Motor	Stepper Motor Nema 17	2	$12.99	$25.98
Motor Driver	Shield Stepper Motor Driver V44 A3967	1 (set of 3)	$11.98	$11.98
Timing Belt with Pulleys	Timing Belt Pulley, 10pcs 5mm 20 Teeth Timing Pulley Wheel	1		$12.99
80-20 Bars	4 Pcs 2020 Aluminum Extrusion (500mm)	1	$29.99	$29.99
80-20 Corners	2020 Series Aluminum Profile Connector Set	1	$24.59	$24.59
Mounting Board	MDF Panel 1/2 in. x 4 ft. x 8 ft.	1	$15.46	$15.46
DC Power Supply	DC 12V 10A 120W Power Supply Adapter Transformer	1	$14.99	$14.99
Controller	Raspberry Pi 3 A+ Computer Board	1	$36.95	$36.95
			Total Cost:	**$172.93**

8. Conclusions

Chess is a millennial, fun and strategic game played by many different generations. The game itself has not changed since its invention which leads us to one of our goals, a modern approach to chess. In addition, our generation grow up watching the series of movies of Harry Potter. One of the most fascinating parts movies is their magical chess game, played in several parts of the film, where the chess moves itself based on the voice of each player. Instead of being "magical"our team decided that we can use our engineering skills to give a modern look to chess and make it fun.

This implementation of chess will allow for a new and exciting form of gameplay. Through the use of a coreXY cartesian movement grid and voice recognition software the board will be able to receive voice commands from players and translate that into physical piece movements in the game. The coreXY system will use a series of pulleys and motors to move an electromagnet around in 2 dimensions under the board. The electromagnet will then engage and facilitate movement. It will allow for a truly hands free and exciting chess experience.

9. References

- Core XY Data
  - https://www.instructables.com/id/CoreXY-CNC-Plotter/
  - https://hackaday.com/2019/11/12/core-xy-explained/
- Equations of Motion for Motor Control
  - http://corexy.com/theory.html
- Raspberry Pi Amazon Echo
  - https://lifehacker.com/how-to-build-your-own-amazon-echo-with-a-raspberry-pi-1787726931