Interactive Automated Chess Set

Brett Rankin, Paul Conboy, Samantha Lickteig, & Stephen Bryant

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

***Abstract* — The Interactive Automated Chess Set is designed to allow for a game of chess to be played without the direct interference of a human user on the playing pieces by combining Electrical, Computer, and Mechanical Engineering skills.**

A claw suspended above the playing board (attached to the top of an A frame beginning beneath the playing board for freedom of movement) will be responsible for descending to a player's specified piece, grabbing said piece, and moving it to a specified destination. LED lights located beneath the frosted plastic sheet that makes up the playing board are responsible for creating the playing grid through a pre-programmed checkerboard design.

The user will input their choice using a Human Machine Interface (HMI) and twelve buttons that allow for a unique piece selection; the user specifies what piece to move and where to move it with the ability to confirm or clear their selection through the use of an LCD screen. A graveyard located on each side of the playing grid houses any piece that has been removed from play so that it will not interfere with continued game play. The human user will be able to go up against a pre-programmed open source artificial intelligence whose moves will also be made by the claw suspended above the board.

***Index Terms* — Artificial intelligence, capacitors, resistors, motion control, electrical engineering, computer engineering, mechanical engineering**

I. Introduction

The purpose of the Interactive Automated Chess Set was to combine not only Electrical and Computer Engineering skills, something each person in the group had at the very least a decent grasp of, but to branch out into Mechanical Engineering as well.

Instead of the pieces being moved by each respective player physically, as during a game of classical chess, the Interactive Automated Chess Set requires that all piece movement be entered into one of two HMI terminals, complete with LCD screens and twelve buttons each, with a claw suspended above the chessboard that is responsible for all of the piece movements. Beneath each square on the grid, a frosted piece of plastic sheeting made by Durex®, reside a group of three LED lights (RGB configuration) that are used to create the checkerboard pattern for the playing grid.

The overall goal of the Interactive Automated Chess Set was to allow for three different modes of play, two of which involve a chess artificial intelligence (AI), whichever the user prefers, as well as a unique spin on one of the most ancient strategy games still in existence today.

Considering the mechanical, electrical, and computer aspects are combined with the myriad of possible features that can be included in a project such as the Interactive Automated Chess Set, like audio speakers that can say anything from a basic "check" warning to encouraging a player to do better or LED lights that do a lighting show comparable to a disco whenever a piece is taken out of play, you are left with a project that has infinite potential that is only limited by the group's imagination, time, and money.

II. Three Modes of Play

When first detailing our overall goals for what we desired the Chess Set to actually accomplish the three possible modes of play were defined as follows in order of importance:

1.) Player versus Computer
2.) Player versus Player
3.) Computer versus Computer

A. Player versus Computer Mode

The Player versus Computer mode was deemed as the most important due to the fact that it involved the widest variety of applicable skills in order to accomplish with the other two modes merely an extension in one form or another of the first and thus not adding much to the complexity of the overall project.

In Player versus Computer mode the Player would enter in his or her selection into the panel designated for Human Machine Interface (HMI) #1 by following the prompts on HMI #1's LCD screen and pressing the buttons that correspond to the alpha-numeric designation of any specific piece the Player wishes to move, and to where he or she wishes the piece to move to, with certain computer protocols in effect to take care of any pieces being removed from play before the chosen piece is placed in its new square. The Computer will then, utilizing its open source AI software, make its own move also utilizing the same protocols that take care of any pieces that need to be removed from play. This format will continue until the game reaches its conclusion.

B. Player versus Player Mode

During a Player versus Player mode both players would enter in their respective selections as specified for the Player in the Player versus Computer mode and continue in that fashion until the game again reaches its conclusion.

C. Computer versus Computer Mode

A Computer versus Computer mode would be for demonstration or tutorial purposes only. Once a user pressed in a GO selection for that mode on one of the HMIs the Chess Set would play against itself until the conclusion of the game or until game play was halted through a selection on one of the HMIs.

III. User Interface

Our original design called for two user interfaces, HMI # 1 and HMI #2, located directly opposite of each other. Our philosophy is to keep the design simple, practical, and cost effective. After considering all of the likely user interactions our final design was aimed at making our product as user friendly as possible. After considering our options we decided on a panel with twelve individual push buttons, utilizing momentary normally open switches, and a LCD screen display would be both the easiest to design and implement as well as the most user friendly. The layout below in Figure 1 and Figure 2 shows the four function switches, the eight zone selection switches, and the twenty digit by four line LCD display.

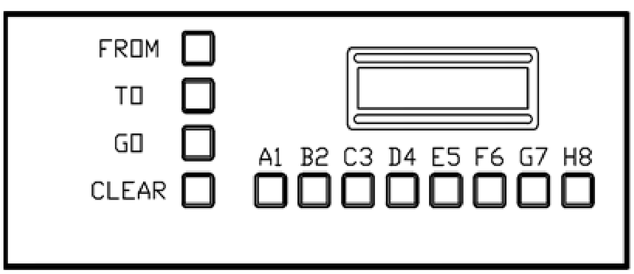

Fig. 1. The HMI panel design for the Interactive Automated Chess Set. This picture was drawn using AutoCAD educational software

Our HMIs are designed for user clarity and simplicity. To reduce the total I/O count the user is asked to push the letter of the destination X coordinate then select the number of the Y coordinate allowing the same eight switches to be utilized for both functions.

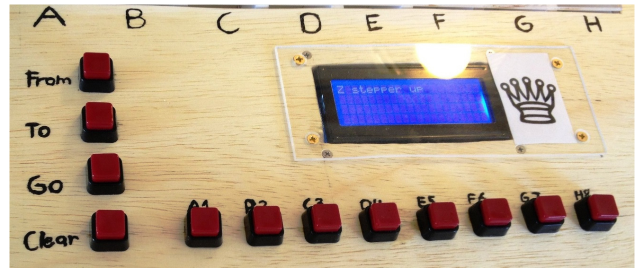

Fig. 2. The actual HMI #1 panel and buttons for the Interactive Automated Chess Set

IV. Software

Since the goal of this project was not to reinvent the wheel we began with an open source chess engine available online at http://chessprogramming.wikispaces.com.

Our software system consists of several modules with a message passing architecture. In our case, the main module is the controller of the system. It interfaces between the chess module and the I/O module. The chess module is the model of our system and it contains internal representations of the chessboard as well as runs the artificial intelligence processes. The I/O module contains device drivers that interface to each electronic subsystem.

One of the most important modules we created was the chess engine itself. It contains several global variables to control the interface to the outside world. Side is the side that is currently moving. Move is defined as the move being input or output from the engine. PromPiece is the requested promotion piece, in our case it is always queen. MaxDepth controls the search depth limit, and Post is our debugging flag.

The engine contains a function to start the engine, called InitEngine. The next function, InitGame, starts a new game by setting the side variable to the starting side. The function Think causes the AI to think up a move from the current position. The move is stored in the Move variable. The DoMove function performs the move currently in the Move variable and toggles the current side of the engine. The ReadMove function converts an input move into the type of move used by the engine. PrintMove prints the current move to standard output. In our case, it sends it over a serial wire which we utilize for debugging purposes. The Legal function checks the move stored in Move for legality. The ClearBoard function empties the entire board, and the PutPiece function is called to put each piece on the board. We renamed DoMove to player_move and created a new function called ai_move that encapsulates the AI processes to create a cleaner interface to the main program.

Another important module we created is the IO module. It contains all of the functions that interface to the individual electronic subsystems such as the LED controller and the motor controller. The motor controller is further abstracted into a composition of stepper motors and other motor control equipment and contains the internal functions, move crane, move stepper, and activate clamp. The only function that it exposes to the IO controller is move piece. All of the classes discussed above can be found in Figure 3 below.

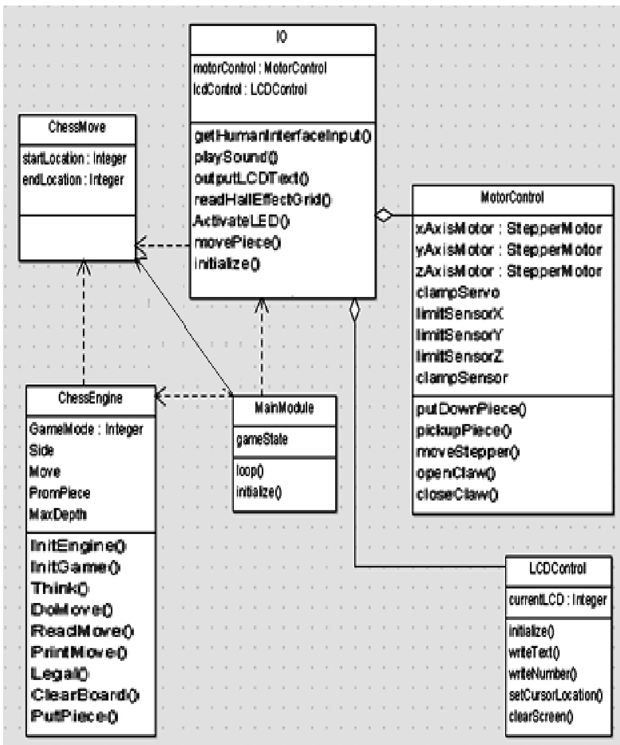

Fig. 3. Class diagram.

V. LED Setup and Controllers

A. LED Setup

Since each square beneath the playing grid needs to be lit up in a RGB (Red, Green, and Blue) configuration, the total number of LEDs required is as follows:

$$8 \text{ rows} \times 8 \text{ columns} = 64 \text{ squares}$$

$$64 \text{ square} \times \frac{3 \text{ LEDs}}{\text{square}} = 192 \text{ LEDs}$$

In order to guarantee that each square could be lit up independently of one another each LED in a column was connected to the cathode of each LED of the same color in the same column; for each LED in a row the anode was connected in the same fashion as the cathodes to the other anodes of the same color in that row. All of the LEDs were connected in this fashion as shown below in Figure 4 and Figure 5. Each color LED is connected to its own LED driver chip.

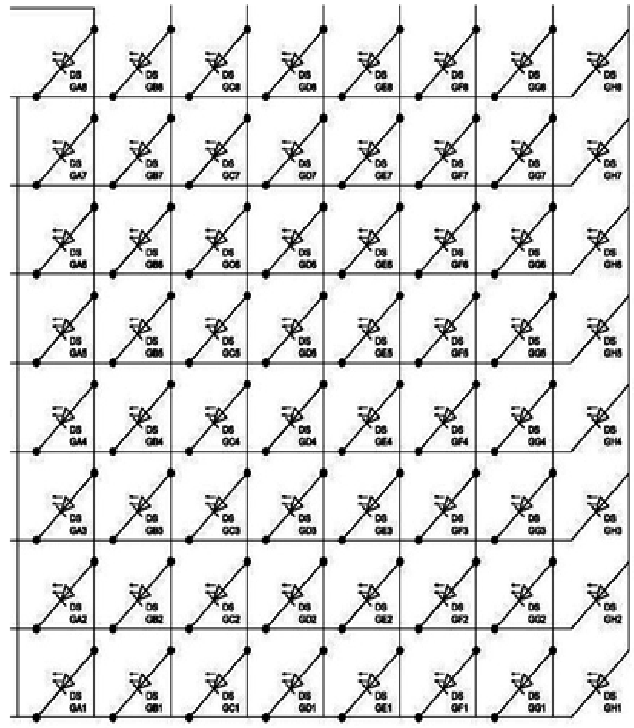

Fig. 4. The configuration for a single LED color showing the anodes connected for each row and the cathodes connected for each column; each color LED was connected in this fashion in parallel with the others. This figure was drawn using AutoCAD educational software.

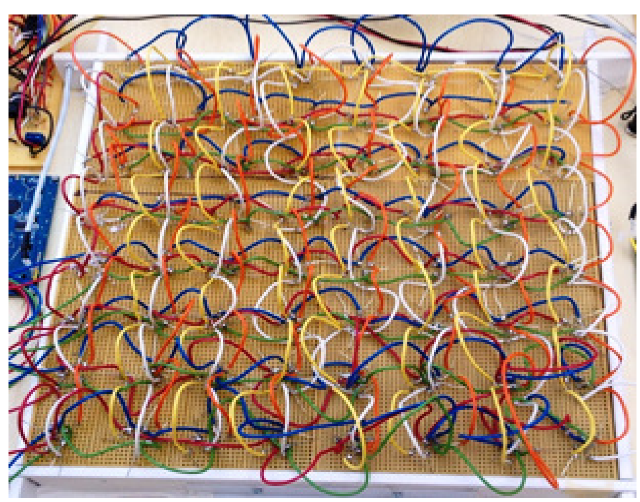

Fig. 5. The entire LED setup where all three colors are connected in parallel with one another.

B. LED Controller

We plan on implementing the LED controller by modifying a library we discovered during our research for the Max7219 LED driver. It contains functions such as MAX7219_Write, MAX7219_LookupCode (which we won't need since we're not displaying actual digits), MAX7219_SendByte, MAX7219_SetBrightness, MAX7219_Clear, MAX7219_Init which must be called by the I/O module before using the controller, MAX7219_DisplayTestStart, as well as a MAX7219_DisplayTestStop. These functions served as a good starting point as they allow us to initialize, perform a test, set the brightness of our driver, clear it, and send bytes to it. Bytes will be sent in a format that maps each individual byte to a rank of the chessboard since there are three LED matrices to control separately, one for each color.

VI. Mechanical Assembly and Motors

A. Mechanical Assembly

The X axis of our system is defined as movement either to the left or right of the user, the Y axis of our system is defined as movement either up or down horizontally with respect to the user, and the Z axis is considered as the up or down movement of the grabber.

The playing surface is made up of 1.5 by 1.5 inch squares arranged in an eight by eight matrix. Along either side of the playing grid a two by eight matrix will be located to place discarded pieces. The two discard matrixes will be located 1.5 inches offset from the sides of the board, approximately one square's width. The Y axis must be able to traverse across rows 1-8 with a total displacement of 10.5 inches. The X axis must also be able to traverse across columns A-H plus five of the discard columns, for a total displacement of 19.5 inches. The Z axis must be able to lift any piece high enough to clear the highest piece on the board but there is not any need to lift the piece much higher than that to provide a safe clearance margin.

The motion track system must also allow for full travel of the carriage plus some amount of over travel margin. Over travel detection has been used to kill the motor drive in the event that an over travel condition occurs and before encountering the mechanical hard stops. The Z axis is driven by an eccentric gear system by mitigating any over travel conditions by design. All three axes of motion required home sensors to calibrate the mechanical system with the control system. When the control system boots up after power has been restored all three motion axes are driven to the home position, essentially the (0, 0, 0) position. Figure 6 below shows the lower front view with the Y axis stepper motor mounted to the moving table that is the base for the A frame structure. The Y axis rack gear can be seen attached to the bottom side of the stationary chessboard frame structure. The stepper motor bracket was shimmed to properly supply sufficient upward force from the spur gear to the rack gear. Figure 7 below shows the actual slides under the board that allows for the A frame to move.

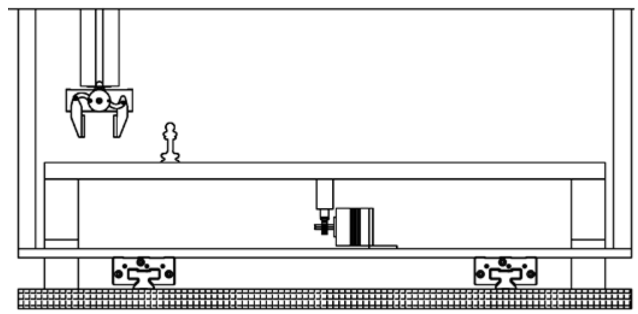

Fig. 6. Y Axis linear slides and stepper motor. This picture was drawn using AutoCAD educational software.

Fig. 7. The gears and sliders located under the board that allow the A frame to move unimpeded.

During the research phase, design problems were clearly identified with the study of a similar chessboard found on the lets make robots website (http://letsmakerobots.com/node/26979); the Y axis in that example had some binging issues. With just one linear bearing for each side the vertical members acted as a leaver arm on the bearings, any slop in the bearings would be multiplied by the height of the vertical members. Our design utilizes an A frame structure that is moved as a structure along the Y axes. The A frame structure, as shown above, is driven from under the chessboard with just one motor and our chessboard straddles the base of the A frame while the legs are positioned closer to the players. This design approach has proven to be more ridged with respect to tolerances and resulted in better positioning accuracy. In order to prevent timing and other issues we have also reduced the number of motors from the example above down to one for the axis of motion. The cut away side view of our design concept can be seen in Figure 8 below.

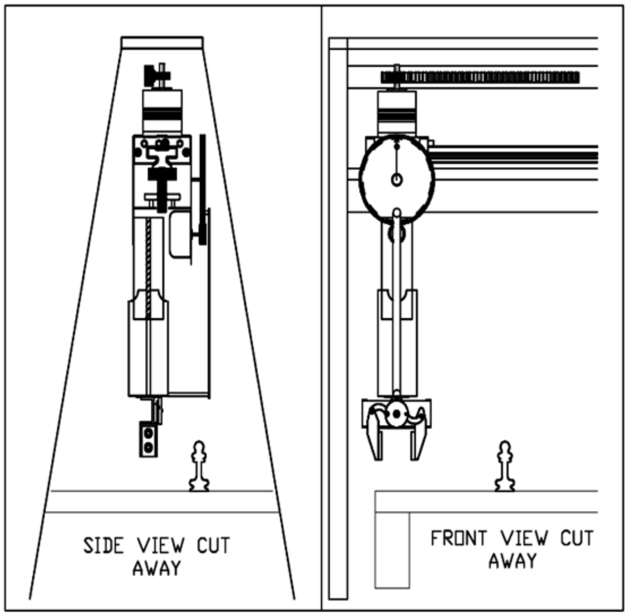

Fig. 8. Grabber and Z axis cut away views. This picture was drawn using AutoCAD educational software.

The diagram clearly shows that we have a ridged structure and have solved the problem with the Y axis tolerance. Along the upper section of the A frame structure is the rack gear that is fixed on the structure. While the stepper motor with the spur gear attached to its shaft moves along the X axis it is also a part of the hanging assembly. In the cut away found in the Figure above two pinch wheels can be seen providing good lateral alignment along the section of tee rail below the sliding carriage. The two pinch wheels also provide lateral stability and supply a horizontal force to the stepper motor and therefore the spur gear will also have that same force acting on the rack gear. The force will assure proper mating of the gears with no expected slippage. Vertical motion occurs as the result of the stepper motor spur gear rotating the larger spur gear above; the linkage lifts and drops the grabber assembly. This design feature allows for unidirectional motor operation allowing for less over travel protection mitigations needed. The front view of Figure 8 above shows the Z axis grabber in the lower position. Due to the rotary motion of the larger spur gear the velocity of the vertical motion is at a minimum simply by the eccentric mechanical design.

B. Motor Control

We selected the bipolar stepper motor for all three motion axes. The bipolar stepper motor proved to be a highly cost effective solution at just $14.00 each for the X and Y axes, and $5.00 for the Z axis. The stepper motor eliminated the need for feedback allowing us to operate our control system in an open loop mode. This resulted in significant cost savings and lead to a greatly simplified design complexity for the control system. The bipolar motor provides more torque to overcome the coefficient of static friction when starting motion compared to the unipolar motor. The motor controllers can be seen below in Figure 9.

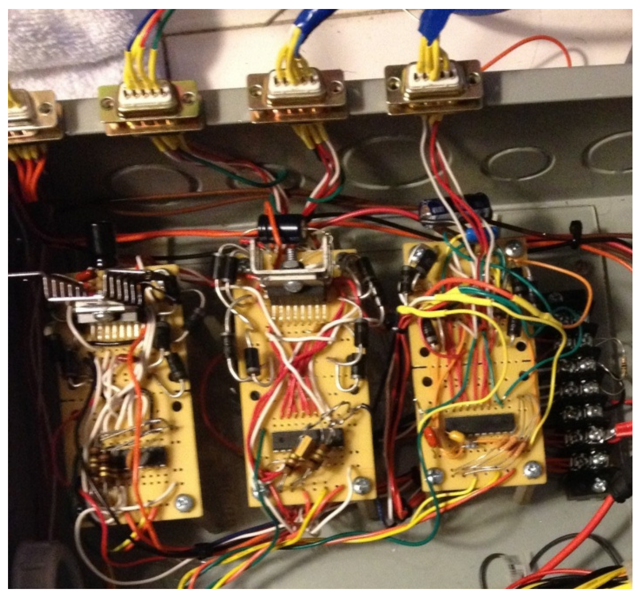

Fig. 9. A close up of the housing with the three Motor Controllers mounted inside

The motors were ultimately selected with the consideration of what spur gears can be mounted to them. All of our stepper motors are 12 Volt shave a step angle of 1.8 degrees, or 200 steps per revolution; this ultimately leads to no gear boxes being required in our design. The X and Y axes motors have a 5 mm shaft bore size while the Z axis motor came with a spur gear. The diameter of the spur gear affects the total displacement traveled in one revolution thus the diameter of the spur gear is a dependent factor on how much resolution of control we have. The size of the square is 1.5 inches by 1.5 inches, and we desire to obtain 95% accuracy in placement of the chess pieces. Using an Excel spread sheet, with various spur gear diameters entered, generated a table comparing resolution, gear diameter, displacement per motor step, and step tolerance to make our final selection, a 1 Mod Flexirack with the Aluminum extruded track for our rack gear for both the X and Y axes of motion and plastic spur gears 5mm in size.

C. Stepper and Servomotors

We wanted to use a modular strategy with requard to the stepper motor controls. Our research revealed that the ST Micro Systems L297 and L298 combination was widely used and easily available. We chose the multiwatt package for the L297, because we could easily attach the proper heat sink to it. We selected the dip package for the L298. This approach allowed use to build our H bridge control circuit on perf board solder the components to the board and use wire to inter connect the components. With three axes of motion we wanted to duplicate the same circuit for all three motors. All three axes use the same model bipolar stepper motors. We have three control signals connected to each motor control unit, the clock signal, the direction, and the enable. Test and adjust revealed that we had much smother operation when the motor controller was run in half step operation.

The servo gripper unit was a purchased device that is controlled with by changing the duty cycle to open and close the gripper.

VII CLAW/GRIPPER

This item was identified as an assembly that we intended to purchase and integrate into our project so as not to spend a lot of time and effort on design work. To fully design and build a griper from scratch would be good project scope for a mechanical engineering project, but not for Electrical and Computer Engineering students especially when considering the other mechanical aspects of our project. We discovered during our mechanical assembly research that limiting the vertical length of the Z axis can improve mechanical stability. Early on a grabber mechanical assembly was found that looked like it was scaled perfectly for our application; the grabber opened up to 1.3 inches, with a 90 degree rotary action, and was light weight and easily mountable to our chessboard. In this case we purchased the grabber and then chose to engineer it into our design. With the decision to purchase the grabber instead of designing and manufacturing it ourselves, the research focused on ways to drive the grabber open and closed.

The device we chose has two movable claws lined with memory foam padding. The device has a disk with two round pins which meant that we could easily install a lever arm that would mate up with the two pins as shown in the various figures above that contain pictures of the design of the claw as well as Figure 10 below.

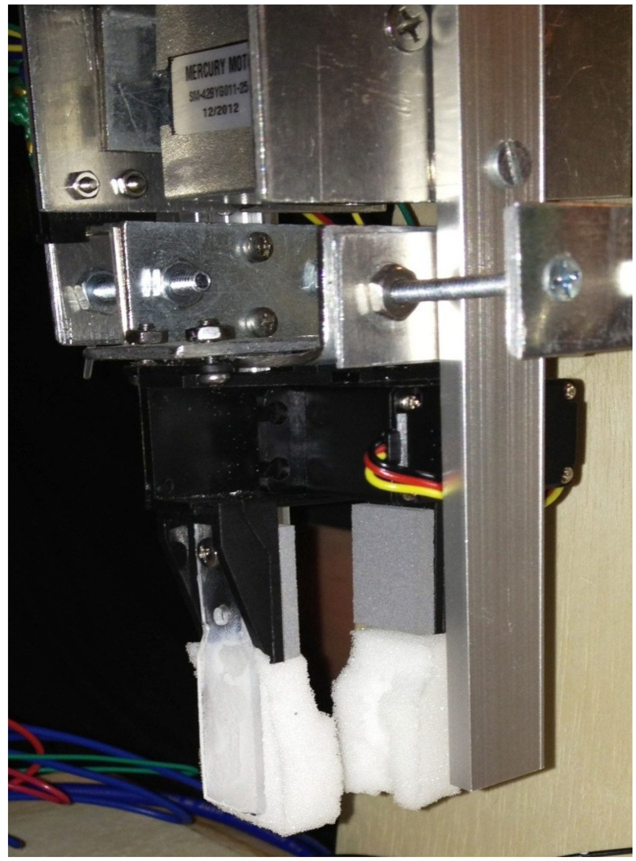

Fig. 10. A close up of the grabber with extenders and memory foam added for extra grip on the smaller and oddly shaped pieces for the Interactive Automated Chess Set

VIII POWER SUPPLY

The power supply that was chosen was store bought to reasonably ensure that the power for our project would remain stable and reliable; the chosen power supply is shown in Figure 11 below. The power source will still be wall power; the conversion from AC to DC is made by an AC to DC converter, linear power supply model number HDCC-150W-AG, 1 input and 3 outputs, with the outputs being +5V, +12V, and -15V. The 5V and 12V outputs will be used, the -15V output will be left disconnected. The high amperage needed for the motors and the relatively high amperage needed for the LED grid can be provided by the power supply, 12A max for the 5V line and 3.4A for the 12V line. The DC-DC converter will have its input linked to the 5V output of the AC-DC converter, the DC-DC converter will convert the 5V to 3.3V, which will be primarily used for various chips thought the project.

Fig. 11. The power supply chosen for our project with the connecter shown in the bottom.

IX Features that Didn't Make It

A. Hall Effect Sensors

The original Hall Effect sensor design was intended as a sort of check and balance to the computer's artificial intelligence as well as a way to insure that the pieces continued to be put in their respective squares throughout a game whether the claw succeeded to place it correctly the first time or not.

Utilizing a piece of Ferris metal, and a magnet embedded into the bottom of each chess piece, the Hall Effect sensors were to be mounted under the board, in the same fashion as the LED grid employing its own controller. The purpose of this was so that each round the computer could compare the board in its memory with the physical board in the real world. If an error was detected the computer could either instruct the claw to pick up and move the piece to its correct destination or it could prompt, through the use of the LCD screen, for the user to move the piece as a last resort. This would also prevent either user from moving a piece to their advantage when it was not their turn.

The piece of Ferris metal was to ensure that the piece would magnetize to the center of each square so that we could minimize placement error and prevent a compounding error from occurring especially during a long game.

Due to the precision with which our claw is able to set a piece, as well as pick it up again, time, money, and space constraints, it was decided that the piece of Ferris metal and the Hall Effect sensors could be left off of the final project.

B. Audio System

The original idea for the audio system was to have speakers mounted inside of the Chess Set that would project any number of sound bites including, but not limited to, "check", "checkmate", and "illegal move" as well as a hidden mode that would have the computer taunting one or both human Players.

The audio system was always considered as an optional feature to be added on at a later date if the time and resources for it were proved to be readily available and since they were not, it was left off of the final product.

C. Modes of Game Play

As stated in Section II there are three modes of game play that we intended at the inception of this project (in order of importance):

1.) Player versus Computer
2.) Player versus Player
3.) Computer versus Computer

Most of our focus has been on the first mode as it would ultimately demonstrate the ability to do the other two. Due to the complexity of getting not only the Player mode working but the Computer mode as well, we felt as a group that our time was better spent perfecting the Player versus Computer mode rather than adding in Player versus Player and Computer versus Computer and having all three modes work at a sub-par level.

X. Conclusion

Given the amount of time and resources at our disposal we were able to create a project that utilized not only the skills we have each learned over the course of our academic careers, but real world experience and determination to learn a new set of skills in a field not directly related to our own. Through trial, error, and expert guidance we have been able to create a viable project, though not with as many features we would have preferred, but a fully functioning Interactive Automated Chess Set.

XI. THE ENGINEERS

Paul Conboy began his electrical career in the Electronics Technology program at Mid Florida Tech. This training provided him with the opportunity to fill an Electrician position in the Monorail Shop at Walt Disney World at age 21. He was promoted to a Sustaining Engineering position in August of 2002. This promotion led to the Electrical Engineering degree path. Paul plans to obtain a PE over the coming years. He also wants to advance in his career with an interest in ride and show control systems. Paul has experience working on large scale DC motor drives, Linear Synchronous Motor launch systems, PLCs and VFDs.

An Engineer with a focus in control systems for industrial use, Brett Rankin began his degree open to different fields of electrical engineering and then began focusing on what interested him. After taking a class focusing on digital systems, as well as an internship at Walt Disney World doing sustaining engineering, his focus on control systems began to be refined. The internship he did gave him experience with large scale electrical control systems. He joined robotics club in 2009 to get hands on experience with electronics and small scale power systems, doing several robotics projects ranging from autonomous underwater vehicles to autonomous land vehicles.

Samantha Lickteig is currently a senior pursuing a B.S. in Electrical Engineering at the University of Central Florida with a minor in Mathematics. She has been fascinated with space and technology since her Dad, an Electronics Technician, first introduced her to science fiction at an early age and they have remained each other's "geeky" inspiration for many years. In her off time she enjoys playing video games, doing martial arts, or spending time with her husband and the animals they have hand raised.

Stephen Bryant is a computer engineer with a focus on software development and intelligent systems. Stephen participated in the UCF Research and Mentoring program during his sophomore year, engaging in smart house research with Dr. Ladislau Boloni. During his junior year, Stephen entered the UCF-Lockheed CWEP program, developing inventory control software. Currently, Stephen is a research assistant at the Institute of Simulation and Training, where he assists with human-robot interaction research.

ACKNOWLEDGEMENT

The authors wish to acknowledge the advice and review of the mechanical system and support of Alan Skaggs MME Walt Disney World Co.

IGUS for donation of the linear slide rails.

Allied Electronics for their assistance with parts.