Optical Interactive Chess Board

Alec Barno, Nikolai Coletta, Alejandro Felix, Cassidy Phillips, and Vinicius Resende

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

***Abstract* — Our project aims to facilitate the learning and playing experience for novice chess players by providing an interactive platform that visually demonstrates the movement of each chess piece on the board. Designed to accommodate two individuals unfamiliar with chess, our innovation eliminates the need for external instruction, enabling players to engage directly with the game. Central to our design is an illumination system integrated into the bases of the chess pieces. Upon selecting a piece, the corresponding squares where it can move illuminate, regardless of the game state. This functionality relies on infrared (IR) light transmission through specially designed filters on the base, detected by photodiodes. These photodiodes then communicate with a microcontroller, activating the appropriate RGB LEDs on the board. By visually indicating possible moves, our system accelerates the learning curve for players, allowing them to grasp the dynamics of each piece swiftly and effortlessly. This immersive approach not only educates but also enhances enjoyment during gameplay. Our project distinguishes itself from existing electronic chess boards by leveraging photonics technology, offering a seamless and expedited gaming experience while preserving the inherent qualities of traditional chess.**

***Index Terms* — Analog to digital conversion, Infrared detectors, Lithium-ion batteries, Shift Registers, Operational Amplifiers, Capacitive touch screen.**

I. INTRODUCTION

The overall goal of this project is to create a physical chess board that will help teach beginners how to play chess. To teach players the basics of chess, they need to first be shown how to initially set up the board and how each piece functions on the board. In order to accomplish this goal, we first have to create smaller goals that need to be accomplished first. The first is to detect and identify the different chess pieces. The second is to light the LED array underneath the chess board properly so that it shows the player where their selected chess piece can move to and to show a player where their piece cannot move to.

A. Key Specifications

We aim to achieve three key specifications in this project. The first key specification is that we want the delay from when piece is picked up to when the move is shown on the chess board to be less than 5 seconds. The second key specification is to have 7 different voltages to

differentiate the types of chess pieces. The last key specification that we aim to achieve is a piece detection accuracy of greater than 95 percent.

B. System Hardware

To guide us in this project, a hardware block diagram served as a visual representation of the components and their connections within our system. The diagram presented below provides a high-level overview of the architecture of our product, illustrating how various hardware elements function together to achieve our desired goal.

Each of the major components and subsystems of our project are highlighted here in a simplified manner in order to convey as easily as possible the idea behind the main functionality of the design, that is, the detection and displaying of moves in the chessboard.

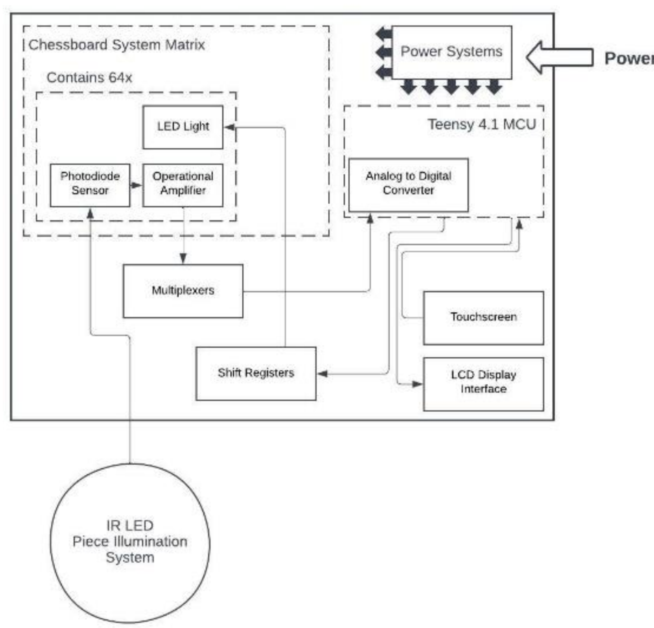


Figure 1 Hardware Block Diagram presenting major systems components.

II. OPTICAL ILLUMINATION SYSTEM

The purpose of the illumination system is to generate light so that it can be directed toward the photodiode which will then distinguish what piece type is on the square above it. In order to accomplish this, we will need a light source that will generate enough optical power to make it easier to distinguish between each piece type. The light source will be placed inside a cylindrical base which will sit under the chess pieces. This light source must be placed on top of the square in a consistent area since moving the light source could change the intensity of the light received by the photodiode. We will need 32 illumination sources since there will be an illumination source under each of the 32 pieces in the cylindrical base.

A. IR LEDs

There are many illumination sources that can provide light to the photodiode such as lamps, LEDs, lasers, laser diodes, quartz halogen lamps, xenon metal halide lamps, and other high and low powered sources. The illumination source we decided to use are LEDs since it is small, cheap, doesn't consume much power and provides enough optical power to help able to differentiate pieces. We also chose to use an IR light with a wavelength of 940 nm because it wouldn't interfere with the light coming from the RGB LEDs under the board. If we had decided to use green light on the cylindrical bases, we couldn't use green to illuminate the squares since the photodiode could pick up this green light and mess with values. Another reason why we chose to use IR LEDs is because they have the smallest forward voltage compared to other LEDs.

Our LEDs consume a current of 47 milliamps at a forward voltage of 1.28 volts. The power consumption of the LED will be 60.16 mW. We compared different types of batteries to find the balance between battery life and size. We ended up choosing 2 triple A batteries each of 1.5 volts and 1000 mAh to power the IR LEDs. We needed the batteries to have higher voltage than the LEDs however we didn't need the voltage of batteries to be too much higher. This is because higher voltage batteries usually are bigger meaning the cylindrical bases would have to be bigger causing the squares and the whole chess board to be bigger. The circuit inside the cylindrical base consists of the IR LED, the 2 triple A batteries, 56-ohm resistor, and the button. We must build this circuit 32 times for each cylindrical base.

After doing some testing we realized that over time the intensity of the IR light went down over time. This is because the voltage of the battery went down over time. We realized that the intensity of the IR light decreased faster initially after the circuit was turned on. After being turned on for a long time the rate at which the intensity decreases slow down. We tested this over 5.5 hours to see the intensity over time.

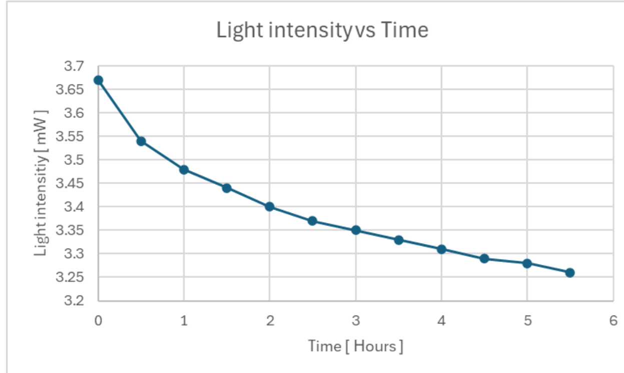


Figure 2 Light intensity of the IR LED over time

B. Cylindrical base for Optical illumination system

Each of the 32 cylindrical bases will be 3D printed using a gray filament. Each of the bases will have a button, an LED, a battery holder, and filters. The bases will be printed in 2 different parts, the top half and the bottom half. This is to reduce time and filament usage. The top half will have a flat surface overhang in inside of the cylinder that will enable the chess pieces to sit on top. The side of the cylindrical base will have cut out for the battery holder to sit in. On the other side of the cylindrical base there will be a button to turn on and off the LED. There will be half cut holes on both halves of the bases for the button to be placed into. The bottom half of the base is where the LED will be placed. We will have a 6.3 mm deep hole, with a circumference of 5.2 mm in the center of the cylinder to place the LED in. Near the bottom of the base is where the filters will be to differentiate the difference piece types. The cylindrical bases will be 2.4 inches tall and have a circumference of 1.35 inches. We wanted the height to be as small as possible so that the chess pieces don't look too tall. On each square we will have a circular ring with a diameter of 1.55 inches. The purpose of these rings is to keep the cylindrical base centered on the square so that the photodiode can get a consistent and accurate reading from the board. If we didn't have these rings the piece can be placed anywhere, and the intensity received by the photodiode would change based on positioning. We didn't need to incorporate lens in the design to make the light converge since we are receiving enough light to differentiate the different piece types.

III. CHESS PIECE IDENTIFICATION SYSTEM

The chess piece identification system is designed to accurately identify and distinguish between all the different pieces on the board at any time. To do this, the chess piece identification system has two main parts: the optical sensor to receive the light from the piece illumination system and use various filters to differentiate between all the different piece types.

A. The Optical Sensor

The first half of the chess piece identification system design is to build a simple yet accurate sensing system under each square of the chess board that can detect what piece is on that square in that time. We used photodiodes as the optical sensor because we are measuring intensity of light and photodiodes can measure any wavelength. Photodiodes are also very cheap, easy to implement into the project, and small in size. The photodiodes we chose are the PD204-6B from Everlight Electronics Company sold on Digikey. We chose these specific photodiodes because the peak wavelength is at 940nm (the same as the IR LEDs used in the piece illumination system) and has a small spectral range spanning from 840nm – 1100nm. We needed a small spectral range that didn't extend into the visible spectrum as that would cause the readings to be altered because of the RGB LEDs underneath the board.

To get the photodiodes to work properly and output a strong signal, we needed to build a transimpedance amplifier circuit. A transimpedance amplifier circuit is a current to voltage converter circuit that boosts a low input current and amplifies the signal into a usable voltage value. We had to build this circuit as the photodiodes output the received signal as a very low photocurrent. After some testing, our photodiodes were only outputting a photocurrent in the range of nanoamps to microamps. We built the transimpedance amplifier circuit as shown in Figure 3 that would enable us to turn the low input photocurrent into a strong and usable voltage value that would then be sent to the microcontroller. We decided to use a 100KΩ resistor as this resistor provided more than enough amplification while limiting the decrease of intensity over time that the IR LEDs experience once turned on. The op amp that we are using is the OPA4990IDR from Texas Instruments. We chose this op amp as one of its features is rail to rail input/output. We needed this feature because the range of voltage values that the ADC on the microcontroller can measure is limited up to 3.3V. We needed to maximize this range since one of our key specifications is to have seven different voltage values to differentiate the chess pieces.

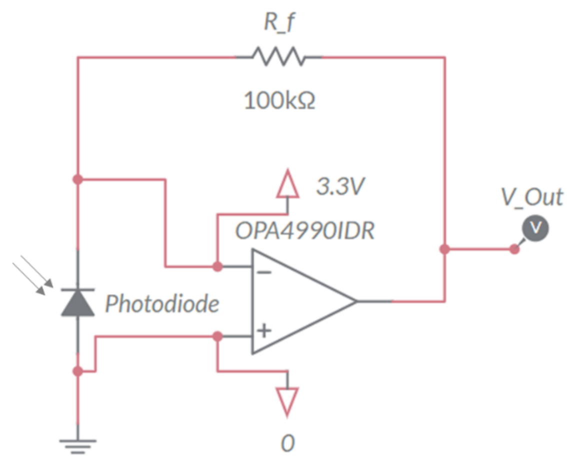


Figure 3 Transimpedance amplifier circuit that is placed under each square of the chess board.

B. Filters to Differentiate Chess Piece Types

The other half of the chess piece identification system was to create and identify a list of values to identify the different piece types. To do this, we used various ND filters. ND filters are neutral density filters and they reduce the overall intensity of light. ND filters are usually not specific to wavelength and instead reduce the overall intensity over a wide spectral range. One of the project's key specifications was to obtain 7 different voltage values related to the different chess piece types. We chose 7 values as there are 6 different types of chess pieces: king, queen, rook, bishop, knight, and pawn. The 7^{th} value is to distinguish between the other color of pawns. It was important to do this as pawns can only move forward during a game and if a white pawn were to cross the middle of the board, we need to know that specific piece is a white pawn moving forward and not a black pawn moving the opposite direction. Every other piece can move in any direction, so we didn't have to distinguish between white and black knights, rooks, queens, etc.

After extensive testing, we determined a range of values from 0V to 3.3V. We tried to separate the ranges by at least 0.4V, but some of the values are closer than that. The deviation for all values is thankfully ± 0.25V at most, while the majority of values only deviate ±0.1V. For most of the values, we needed to use a combination of filters to achieve the desired ranges, as shown in Table 1. None of the values are associated with the different piece types as that is subjective and only a personal preference for the coding section.

Table 1 Intensity Values Based on Various Filters

Value #	Filter(s)	Voltage	±
1	No filter	3.3V	±0.1
2	ND9 + ND4	3.126V	±0.1
3	ND9 + ND16	2.569V	±0.2
4	ND6 + ND2	2.206V	±0.25
5	ND6 + ND8	1.272V	±0.2
6	2x ND6	0.76V	±0.1
7	ND3 + Blackout	0.294V	±0.1

IV. SOFTWARE DETAIL

With the large variety of hardware being used from the project, every driver that could help make this project easier on the software side was used. The Photo Diodes will be set in a system of activation and reading an analog signal, so using the ADC built into the microcontroller will be done to process and store values from all the photo diodes in real time. This analog value will then be converted into a piece value and depending on the board state and the legality of the move, it will yield different outputs to the peripherals. The LED array will be a primary way to display output to the user and will be controlled through shift registers. The touch screen on the display will serve as way to interact with the display to control settings, the chess clock, and game setup.

A. Analog To Digital Conversion

To be able to process the input data from the photo diodes, an Analog to Digital Converter (ADC) would have to be used. Fortunately, with the microcontroller selected for this project, the Teensy 4.1, there was an ADC already built into the device. To be able to work with the ADC is extremely simple with the Teensy, as it is a one-line command that already is preconfigured with the Teensy's setup to run. The resulting value comes at a 12-bit resolution, but this value will be trimmed to just 10 bits of resolution because of noise. Each piece on the board will have unique analog values associated, as set up in configured by the piece identification team and will simply translate that read value into the resulting piece it should be.

With using these analog values, one issue with reading the pieces is that the values for the piece would fluctuate as the user is lifting or placing a piece. As a result, a system was put into place that would only "lock-in" what the piece should be once the value became stable, where it wouldn't change for some period (about 50ms). This removed many immediate piece misdetections as the user interacted with the board.

To be able to read from every single piece on the board simultaneously, 8 ADC values will be read at once. Using a multiplexer, a row selection will be done to be able to isolate the 8 ADC input signals to a single row. The program will quickly oscillate between these 8 different rows to monitor the state of all 64 photo diodes under the board.

B. Shift Registers for LED Array

For the control of an 8x8 array of LEDs, shift registers were used from data sent by the microcontroller. After the ADC effectively registers what piece has lifted, the corresponding legal moves will be displayed on this array of LEDs, color-coding different actions. Valid moves are shown in blue, and illegal moves made that need to be un-done will be shown with red, along with a few other colors for other niche cases.

Each of the four shift registers control either RGB values or what row will be selected, so the output to these shift registers will be an unsigned 32-bit integer. Each of the groups of eight bits will go to a different shift register and are passed around in a daisy chain style to be able to communicate to all four. The software takes its 8x8 matrix of color output stored in memory and converts that into the corresponding shift register values and sends out that signal through a simple iteration. Because this process does require the CPU for constantly oscillating between the groups of LEDs being turned on, this process is setup on a clock to interrupt the code and run the latest updates several thousand times per second. This frees up the CPU to do other calculations without worry about keeping the LED oscillation smooth.

C. Display and Touch Screen

The way the user will interact with the system beyond just the lifting and places of the pieces will be done through the mounted display on the side of the board. The initial design of the board had all interaction with the display done through a series of buttons. However, with the display we selected, a capacitive touch screen was a feature built into that device and proved to be very accurate and useful. Thus, the buttons were scrapped for a fully touch screen integrated user interface.

Both the display and touch screen on this device is communicated with the SPI protocol. With SPI, you can share MOSI, MISO, and clock signals, so the same SPI port was used on the microcontroller for both the display output and the touch screen input. There did have to be separate ports used for the chip select (CS) component, to be able to distinguish between which part of that device to use with those lines. Much of the interfacing with this device was done through the library TFT_eSPI when configured specifically for the ST7796 driver- which is the driver inside the display. This library allowed for a lot of standard screen drawing, including different shapes, texts, and even sprites when stored as bit maps.

The user interface for this project will be fully integrated with the touch screen as control. The main menu page will consist of buttons that will activate when tapped, sending the user to different menus. There will be an options menu, which will serve as a way for the user to decide how the game will run. When starting the game, the display will guide the user on how to correctly set up the board piece by piece and will not proceed in the game until the board is setup completely correct. From there, the game will begin, and each user will use the touch screen to pass there turn to the next player, as well as pause the game if needed.

V. LED ARRAY

A significant portion of this project relies on giving feedback to players as they play a game of chess. This feedback is provided by the LED array that sits under the acrylic chessboard. The LED array will light up and inform players where their selected chess piece can go and while also displaying information about illegal moves and captures.

A. Configuring the LED Array

Since we wanted to use multiple colors of LEDs to provide feedback to players, we decided to use RGB LEDs. By using these LEDs, we will have access to 7 different colors. The specific type of RGB LEDs that we will be using are common cathode LEDs. To configure these LEDs and to minimize the number of bits needed to control all 64 LEDs, we designed an array where the anodes of each column are connected to a single wire while all the cathodes in a row are connected together. By wiring them this way, we can turn on a specific LED by setting the column to "high" and the row to "low". Essentially this matrix is a grid system.

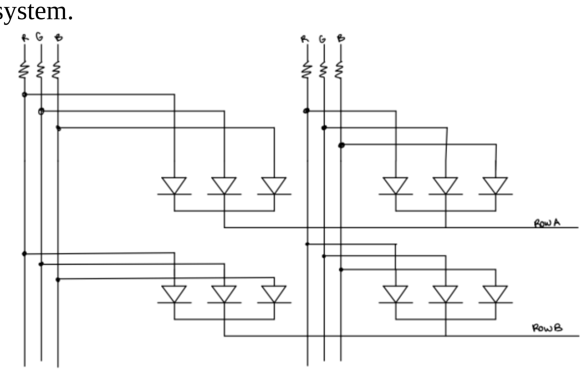


Figure 4 2x2 configuration of common cathode LEDs

There are issues with this configuration though, we cannot turn on LEDs that are diagonal to each other. If we turn on multiple LEDs, the brightness will also be affected. Our solution to overcome this issue is to alternate quickly between all the LEDs so that they look like they are all on, but truly only one is on at a time.

B. Controlling the LED Array

We looked into two options on how we could control the LED matrix: LED drivers and shift registers.

With LED drivers, they have built in features to help prevent the LEDs from burning out and even allow for pulse width modulation to further access to more colors than the initial seven.

Shift registers are an integrated circuit whose main benefit is the ability to save pins on the microcontroller. Depending on how many bits the shift register can control, the shift register can take in *n* amount of bits and send each bit to a designated pin. Another benefit is the ability to create an even larger shift register by daisy-chaining multiple shift registers together. This means that any leftover bits will be sent to the input of the next shift register. Compared to the LED drivers, shift registers have a quicker switching speed which is important for making our project successful.

Based on our initial testing of the shift registers, we decided to stick with using them for this project. In total we used 4 8-bit shift registers to control all the columns and rows of the LED array. In total there are 24 columns and 8 rows, so 32 bits will be needed which can be accomplished with 4 8-bit shift registers. We selected the SN74HC595N shift registers which are an 8-bit shift register that have a switching speed of 24 MHz. These shift registers can both sink and source current. Initially we planned to use transistors to control the rows of the array, but after testing we were able to implement shift registers in rather easily.

Each of the four shift registers are used for a specific purpose: one to control the red columns, one to control the green columns, one for the blue columns, and the last one to control the rows. The shift registers for the columns will turn the pin designated for the specific column to high and the others to low while the shift register for the rows will work in the opposite way.

When ordering the PCBs for the LED array, we created a design where the boards could be connected to each other like puzzle pieces. We ordered 5 copies of a singular board but it was easy to connect 4 4x4s together to create the full 8x8 array. The one challenge with this project is ensuring all the pieces that go under each square are assembled correctly 64 times.

VI. OTHER HARDWARE ELEMENTS

There are other hardware elements that had to be researched and selected for this project. Each section will cover the specific characteristics we looked into when selecting components.

A. Operational Amplifiers

As part of the transimpedance amplifier, operational amplifiers are an essential component. Since we are only using a 5 Volt source for the entire project, we want to select an op amp that can be biased using this voltage.

When selecting an op amp, we had two main characteristics that we looked at: rail-to-rail input and output as well as if it is a low power op amp. We ended up selecting the OPA4990 which is a quad packaged op amp that works with small rail voltages. The OPA4990 is capable of working with small rail voltages and its output maximum will be that rail voltage minus 0.2 V.

B. Multiplexers

Since our microcontroller only has 18 ADC pins and we have 64 values we need to be able to continuously read, we looked into multiplexers. Multiplexers will be used to take in multiple inputs and condense them to one pin. We selected 8-bit multiplexers so we will need to use three pins in order to switch between all 8 pins. We configured the multiplexers so that three pins will control all eight 8-bit multiplexers. Each multiplexer will be in charge of a row on the board so we will continuously alternate between 64 squares. We ended up selecting the CD74HC4051. This 8-bit multiplexer is specifically for analog signals and it features a quick switching speed.

VII. BATTERY AND POWER SYSTEMS

Since we envisioned the project with an emphasis on mobility and flexibility, we knew right away that we would need to have some form of power storage so that the board wasn't always confined by the need for a power outlet.

This section guides the reader through our group's thought process while determining the ideal type of battery for our design. During this, we were largely presented with two options, lithium polymer pouch cells (LiPo) and lithium-ion cylindrical cells (Li-ion). Other types do exist, such as nickel metal hydride batteries, which do not meet our energy needs, and lead acid batteries, which are unnecessarily heavy, bulky, and expensive for use in this application.

A lot of research was done to analyze and compare the key characteristics of both LiPo and Li-ion cells and ascertain which of them best aligned with the interests of our project, aspects such as a cell's physical and electrical properties, battery safety, as well as the purchasing cost of the cells for construction of our battery were all considered, and weighing each format's pros and cons caused us to settle with the use of Li-ion batteries for this design.

A. Battery Pack Design

Once the voltage and capacity requirements of the system are established, it is possible to arrange a group of Li-ion cells in a specific assembly configuration to create a custom battery pack that meets our desired specifications. By connecting cells in parallel, we can reach a desired battery pack capacity, with each connected cell adding its nominal capacity to the total capacity of the battery. Afterwards, connecting such parallel groupings in series will achieve the same effect, but this time, raising the voltage value between the positive and negative terminals of the battery by a value equivalent to a single cell's nominal voltage.

These two properties are easily visualized in the equations below:

$$\#\ of\ Parallel\ Cells = \frac{Desired\ Pack\ Capacity}{Cell\ Capacity}$$

$$\#\ of\ Serialized\ Modules = \frac{Pack\ Nominal\ Voltage}{Cell\ Nominal\ Voltage}$$

Figure 5 (A) Equation For Number Of Parallel Cells (B) Equation For Number Of Serialized Modules

The battery will be providing the system with a constant 5V supply and our estimations for the current draw of the load lands us at a projected total power consumption of less than 10 watt-hours. Thus, we set the requirements for the design of the pack to a minimum battery rate of 40 watts, for a comfortable period of 4 hours of play. To achieve this, we will be utilizing a 2S3P design, meaning that the pack will hold two groups of three cells each as described previously in this section. Considering each cell has a nominal voltage of 3.6 V and a capacity of 2500 mAh, this approach will create a 6-cell battery pack with a nominal voltage of 7.2 Volts and a total capacity of 7500 mAh, which can deliver up to 54 watts of power under optimal conditions. This gives our group plenty of leeway in an already conservative estimate. We expect that our final product will be able to operate for at least six hours on a fully charged battery.

B. Battery Management and Peripherals

Connecting the battery directly to the remainder of the system would be both dangerous and inefficient. To ensure the proper operation of our design on battery power, we need some additional components to monitor, control, and safeguard our battery, and the systems therein connected.

The primary component in this protection system is a Battery Management System (BMS). These boards essentially act as an interface between the battery and the remainder of the system and are responsible for the balancing of charge across all battery cells, making sure they charge evenly throughout. Without such a system in place, some cells could possibly charge at a rate different from the others due to minor chemical differences within the cell, which could lead misalignment in the voltages of the cells, and ultimately to the destruction of the battery.

Initially it was proposed that we built our own BMS control board. However, we can find several of these boards in the general consumer market today and, with some careful research, we can identify models such as the HX-2S-JH20, which we picked for use in this project. In addition to being really low-cost solutions, ready-made modules such as this one have the added benefit of including additional protection mechanisms. The JH20, for example, includes a system for short circuit protection and is capable of protecting the battery from overcharge, over discharge, and overcurrent conditions.

We will also be using a voltage regulator to stabilize the output of the battery and prevent damage or improper operation of the device. This module will provide the system load with a constant 5V, which would be impossible to maintain otherwise, since the voltage of the battery changes overtime proportional to its state of charge. We will be using a buck switching regulator since these much more energy efficient when compared to their linear regulator counterparts. The buck converters we will be using in this project are found readily available in the market in the form of small self-contained and ready-to-use modules. They utilize LM2596 series chips, a collection of integrated circuits that provide all the active functions necessary for a step-down switching regulator. These chips can drive a 3 Amp load, which should be well over our load requirements for this chessboard.

Finally, for battery recharge, we had initially settled on USB-C to be our system of choice for the recharging of the batteries. However, we ran into some problems that led us to swap to a more traditional 9V power supply. First, the nature of USB-C is bidirectional, so we had to develop a method to stop the connected battery from discharging when the device was plugged in. But the biggest problem was that, although the system worked flawlessly for the smaller battery we created during prototyping, when connected to the larger battery used in our final design, our power delivery boards would cease to function. Overall, the change hasn't affected much of the design, our new supply has the same power characteristics as the one we had planned with USB-C, supplying the system with a 2A current at 9V.

we will be using a common 9V adapter as our power supply system of choice. These are commonly bought in stores for general use and are easily integrated in our project.

VIII. PROJECT EVOLUTION

Our project has been changed many times to fix concerns we had noticed and that were pointed out and to make sure we could achieve our key goals. The optical illumination system has been the system that has changed the most over the entirety of the project.

At first, we planned on using fibers to illuminate the bottom of each piece. Each piece would have had a different reflective material that would have reflected the light back to the photodiode. In this iteration the pieces would have been distinguished by the reflectivity instead of intensity. We moved away from this idea after realizing how hard it was to keep the fibers positioned in place and at the same angle for all 64 squares.

The next idea was to have a modulating IR LED under each square. We again would have different types of reflective material that would reflect the light to the photodiode. The advantage of this method was that ambient light could be removed from our readings. We moved away from this idea because we didn't believe that ambient light would be a big problem in our project, and it would make this project more software intensive.

After some more thinking and discussion we landed on the current idea of putting the light under the chess pieces and having different filters to change the intensity of the light.

IX. CONCLUSION

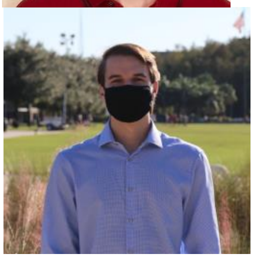
All the subsystems mentioned above combine to form a chess board that would help beginners learn how to play chess. This project not only simplifies the complexities of the game but also introduces an interactive element that strengthens the understanding for novice players. Through its design, this project serves as both a mentor and a playground, inviting those new to the game to explore the world of chess with confidence and curiosity.

X. ACKNOWLEDGEMENT

The authors wish to acknowledge the assistance and support of Dr. Weeks and Dr. LiKamWa for their help making this project a reality.

XI. BIOGRAPHY

Alec Barno is a 23-year-old graduating with his bachelors in photonic science and engineering. Alec wishes to pursue a career within the optics community and is currently pursuing jobs in the field.



Nikolai Coletta is a 22-year-old graduating with his bachelor's in computer engineering with a minor in mathematics. Nikolai wishes to pursue a career in software design and implementation and has a job at Fast Enterprises to begin this career in July.

Alejandro Felix is a 23-year-old graduating with his bachelor's in photonic science and engineering. Alejandro hopes to pursue a career in optics and photonics and is currently exploring job opportunities in the field.

Cassidy Phillips is a 22-year-old graduating Electrical Engineering student. Thanks to this project, Cassidy hopes to pursue a career in electrical design and fabrication.

Vinicius Resende is a graduating Electrical Engineering student at University of Central Florida. They plan to complete their bacca laureate at 23 years old, in May of 2024. Vinicius hopes to pursue a specialization in mobile robotics and a career working in research and development of such agents.