Internet Connected Chessboard

ECE 445 Design Document - Spring 2019

Team 61 - Joel Mathews, Ritish Raje, Jeffrey Ito
TA - Thomas Furlong
2/21/19

Contents

1 Introduction

1.1 Objective

Chess is a board game that is centuries old. In the modern age of computers, there have been many online applications created to allow players to compete against one another despite being in different parts of the world. Looking at a computer screen for extended periods of time can cause fatigue in the eyes and mind, both of which are essential tools for any chess player.

To solve these problems, we plan to create a chessboard that maintains the ability to play opponents over long distances while eliminating the need for a computer screen to play the game. Our goal is to allow players to regain the physical interface of a chessboard to reduce strain on their eyes. To accomplish this, our board will interface with a PC to send and receive data about the current state of the game.

1.2 Background

Playing online chess is extremely easy. One only has to have an internet connection and a computer to play. Some people may not like staring at a screen, as it takes away from the experience. We want to change that by creating a smart chessboard, which takes away the computer, but still allows people to play each other remotely.

Players just beginning to learn the game of chess may find it difficult to find other players to compete against. Playing the game of chess online can look and feel quite different than when playing on a board. Allowing beginners to consistently improve their skills with the board will help them to learn the game faster because they will not need to learn about the interface that comes with an online version of the game.

1.3 High-Level Requirements

1. The chessboard must be able to sense the location of the player’s pieces and send this data to a PC with 100% accuracy.
2. The chessboard must be able to receive data from the PC about the location of the opponent’s pieces and display this data on the chessboard accordingly with 100% accuracy.
3. The chessboard and pieces must maintain a similar feel to existing, high-quality chess sets.

2 Design

Our design will consist of 4 main modules: the power supply, the game board, the control, and the PC. The power supply is used to ensure that the entire system is powered continuously throughout a game. The game board will be the physical interface that the player interacts with. Chess pieces will be free to move on the surface as with a regular chessboard. The reed switches are used to detect a piece. If there is a piece above a spot, the reed switch will be activated. The control module is used to process and transmit all the information from the board and send it to the PC. Similarly, the PC will send information about the game state to the control module, which will activate LEDs when necessary.

2.1 Block Diagram

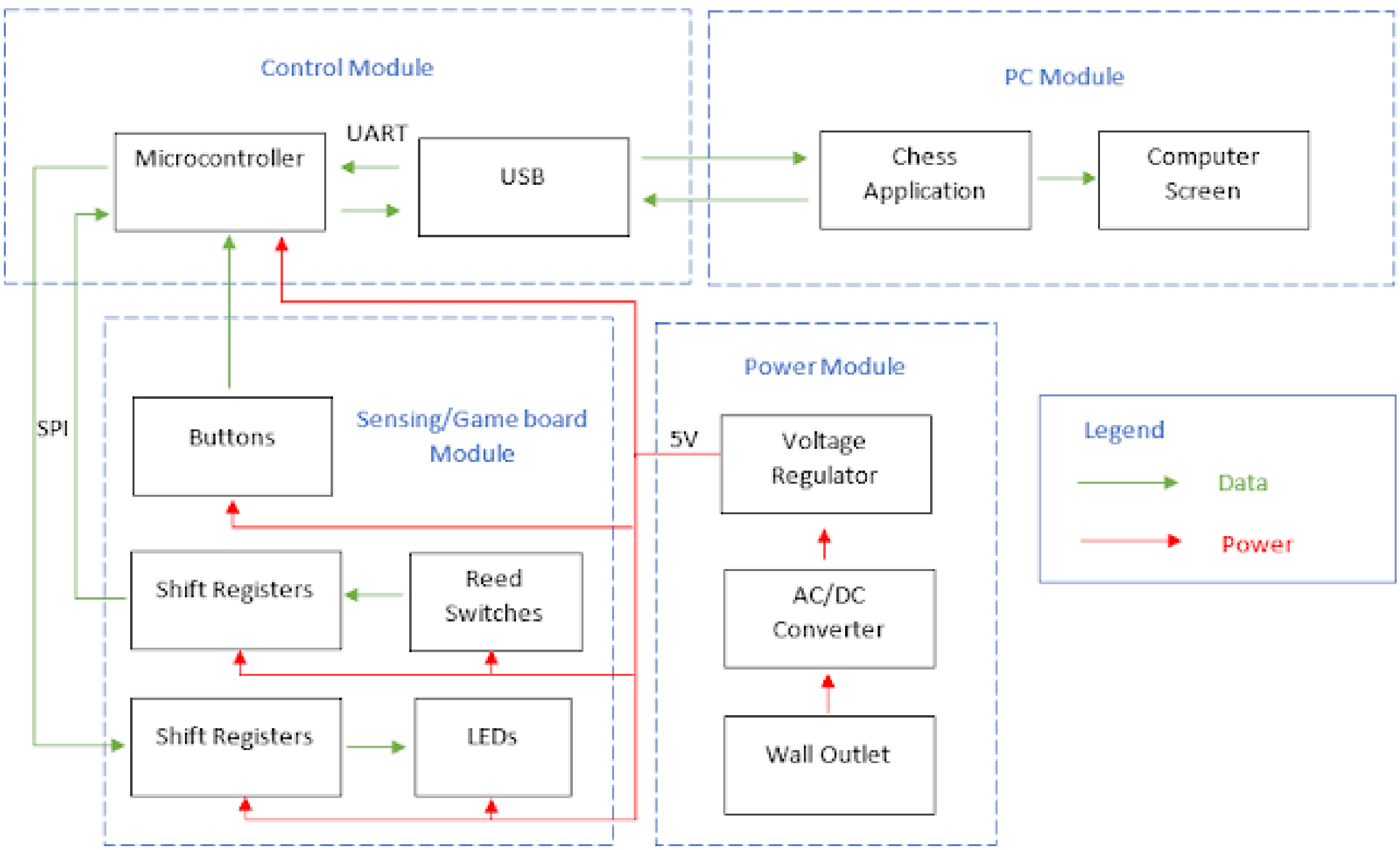

Figure 1. Block Diagram

2.2 Physical Design

Our physical design will mimic a professional chessboard as closely as possible. The recommended size of the base of a king is 75-85% the size of the square [1]. Assuming we make individual squares a size of 6 cm, that allows for a king base size of 4.5-5.1 cm. There will be two buttons to the side of the board to allow for reset and changing whose turn it is. A thin, translucent layer of acrylic will be what the chess pieces will be placed on top of. This layer allows for the LEDs to shine through while still hiding the underlying electronics.

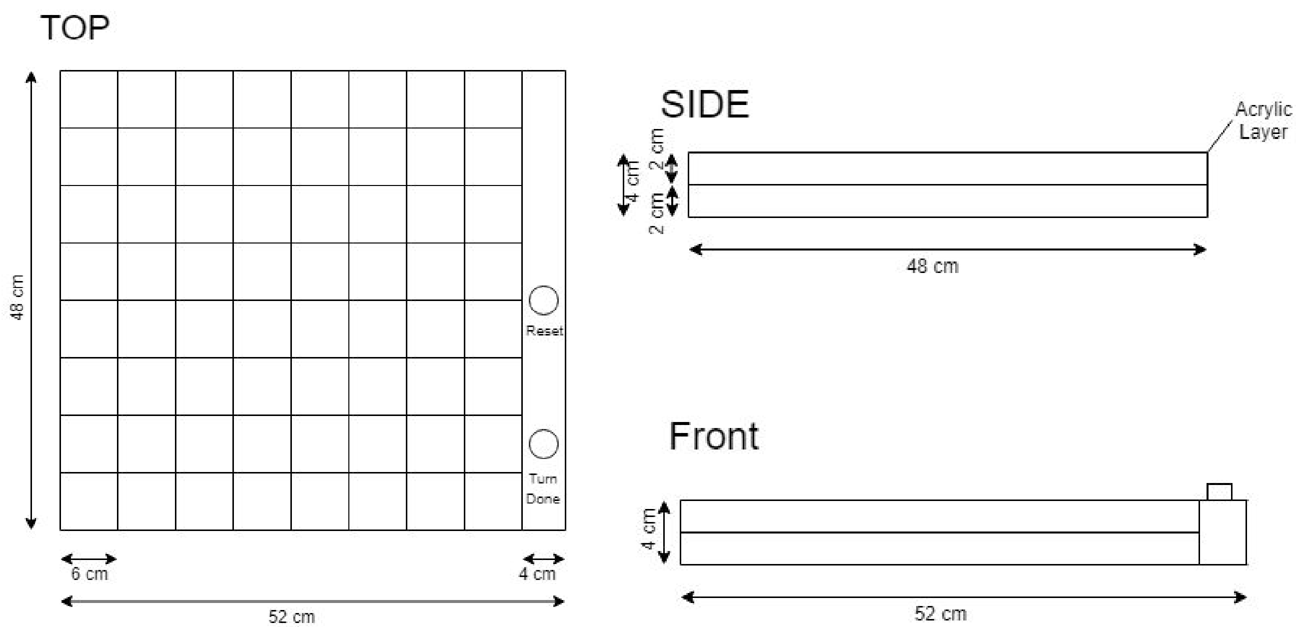

Figure 2. Physical design of chessboard

2.3 Control Module

The control module will be used to handle all the sending/receiving of data to/from the game board. Sending/receiving data to/from the PC will be handled in this module as well.

2.3.1 MCU

The microcontroller will accomplish most of the work for our project. It will take readings in from the reed switches, convert this to locations of pieces, and sends this data to the USB module. It will receive data from the USB module and translate this to lighting up certain LEDs if necessary.

Requirements	Verification

1. Can transmit data to PC through USB 2. Can receive data over SPI from shift registers 3. Can send data over SPI to shift registers	1. A. AForm a packet of data that contains the bits 10010011. B. Transmit this packet to the PC and ensure that the received packet matches the sent packet. 2. A. Load the shift registers with the bit sequence 10010011. B. Perform a shift-in on the MCU and ensure the received data is correct 3. A. Perform a shift-out on the MCU with the bit sequence 10010011. B. With a DMM, probe each parallel-out pin to ensure that the bit sequence is correct.

Part information: ATmega328. Chosen for 1 UART, 2 SPI interfaces, 32KB flash programmable memory, and 5V operating capability [2].

2.3.2 USB driver

The usb driver will handle all the communication between the PC and the gameboard. It will send/receive data to/from the MCU and send/receive data to/from the PC.

Requirements	Verification
1. Can facilitate data transmission between the MCU and the connected PC	1. A. Form a packet of data that contains the bits 10010011. B. Transmit this packet to the PC and ensure that the received packet matches the sent packet.

2.4 Sensing/Game Board Module

The sensing/game board module will be the physical interface that the player interacts with. This module also defines how data will be collected about the location of pieces on the gameboard and how data about opponent moves will be communicated.

2.4.1 Reed Switches

The reed switches will be used to determine where the pieces are on the game board. Reed switches are magnetically activated switches. There will be small magnets attached to the bottom of each piece. When there is no piece above a space, the switch will be in an “OFF” state which means it will be sending a high voltage reading. When a piece is above a space the switch will activate to an “ON” state and send a low voltage reading.

Requirements	Verification
Activate from the effects of a magnet that is 0-3 cm away.	A. Connect read switch to test circuit, placing the switch at the 0 cm mark on a ruler. B. Slowly move a magnet closer to the switch, starting at the 10 cm mark. C. Make note of the voltage across the switch every .5 cm. D. Ensure that the measured voltage is 5 V ± 5% until the magnet is 3 cm away, at which point the voltage should drop to 0 V.

Part information: Cylewet CYT1004. Chosen for their low cost and close activation distance. The activation distance, as described by a customer, is less than 3 cm with a neodymium magnet [3].

2.4.2 Buttons

The buttons will be used to manually reset the game board and indicate that the player is done moving pieces.

Requirements	Verification
Easily pressable buttons.	A. Press the buttons to ensure that they are easily pressable.

Part information: SPST normally open 1A pushbutton. Chosen for their ease of pressing.

2.4.3 LEDs

The LEDs will be used to help indicate where the opponent has moved his piece and indicate if an illegal move has been made by the player.

Requirements	Verification
LED must be visible under opaque acrylic layer.	A. Power on LED while it is underneath the acrylic layer, visually ensuring it is still visible.

Part information: EDGELEC 5 mm RGB Tri-color 4-pin LED. Chosen for low cost and individually addressable RGB color channels [4].

2.4.4 Shift Registers

There will be two types of shift registers (SRs) used: parallel-in, serial-out (PISO) and serial-in, parallel-out (SIPO). The PISO SRs will be used to facilitate communication between the MCU and the reed switches. Each reed switch output will be connected to a parallel-in pin on the SR. The MCU will send a load signal to the SRs, which will load in the data coming in from the reed switches. Then, the MCU will begin shifting the data. 8 8-bit SR's will be connected in series, with the serial out pin from one SR connected to the serial in pin of the next SR (see figure 7). The final serial-out will be connected to the MCU. This is done to reduce the number of pins required to be attached to the MCU because only 4 pins need to be connected to MCU to control and retrieve data from all the SRs. 3 pins will be used to control the shift registers: 1 for clock, 1 for load, and 1 for shift-enable. 1 pin will be used to connect the serial-out pin from the last SR to the serial-in pin on the MCU to retrieve the data.

The SIPO SR's will be used to facilitate communication between the LEDs and the MCU. 8 8-bit shift registers will be used for each of the 3 color channels for the LEDs, so a total of 24 8-bit SR's will be connected in series to address each color channel of each LED. The serial-in pin of the first SR will be connected to the MCU, with the last parallel-out pin connecting to the serial-in pin of the next SR (see figure 8). When the LEDs need to be updated, the MCU will begin shifting in bits until all the SR parallel-out pins are the correct value. SR's are again used to reduce the number of pins required to be connected to the MCU.

Requirements	Verification
1. Allows for parallel input, serial output. 2. Allows for serial input, and parallel output.	1. A. Connect each parallel input pin to a GPIO pin on the microcontroller, setting each pin so the bit sequence is 10010011. B. Connect the serial output pin to the serial input of the microcontroller. C. Send a load signal from the MCU to the shift register.

	D. Perform a serial shift in on the MCU ensuring that the bit sequence read was 10010011. 2. A. Connect the serial output pin of the MCU to the serial input pin of the register. B. Perform a serial shift out on the MCU with the bit sequence 10010011. C. Probe each of the parallel output pins with a DMM ensuring that the final bit sequence is 10010011.

Part information: Texas Instruments SN74HC165N. Chosen for low cost, parallel-in, serial-out capabilities, and 5 V operating voltage [5].

Part information: Texas Instruments SN74HC164N Chosen for low cost, serial-in, parallel-out capabilities, and 5 V operating voltage [6].

2.5 Power Module

The power module will be used to provide power to all the components in the game board and control modules.

2.5.1 Voltage Regulator

The voltage regulator will be used to convert the DC voltage to the appropriate voltage needed by the rest of the components. It will also contain an isolating circuit to prevent any power surges from damaging components on the gameboard.

Requirements	Verification
Converts DC input to 5 V ± 5% output.	A. Connect input of voltage regulator to a power supply. B. Power on the supply and adjust it to provide 12 V ± 5%. C. Measure output with DMM to ensure it remains steady at 5 V ± 5%.

Part information: Linear Technology/Analog Devices LT1084CT-5#PBF. Chosen for input voltage up to 20 V with a steady output of 5 V and a maximum current of 5A [7].

2.5.2 AC/DC Converter

The AC/DC converter will be used to convert the wall outlet voltage (120 V 60 Hz AC) to a DC voltage that can be used by our system.

Requirements	Verification
Converts 120 V 60 Hz AC to >5 V DC.	A. Plug converter in to wall outlet. B. Measure output voltage using a DMM to ensure it is >5 V.

2.5.3 Wall Outlet

The wall outlet will be used to provide power to the components of our system. This is a typical wall outlet in the United States (120 V 60 Hz) [8].

Requirements	Verification
Outputs 120 V ± 5% 60 Hz ± 5% AC.	A. Connect wall outlet to oscilloscope to ensure output waveform is 120 V ± 5% RMS voltage and 60 Hz ± 5% frequency.

2.6 PC Module

2.6.1 Chess Application

This application will process the data it receives from the Bluetooth controller, it will send data to a cloud so that the opponent will receive the move registered on the smart chess board.

Requirements	Verification
1. Chess application receives the correct information regarding the game state every time it is changed. 2. Chess application sends the correct data over the cloud	1. Change the game state. Print out the packet of information right before it is sent to the cloud 2. Connect a second computer to the cloud. Print out the data it receives from the sender.

2.6.2 Computer screen

The screen will display the chess board in a digital format. It will receive information on what the opponent has done. Tentatively, we want to connect the smart chessboard to another smart chess board, but due to the costs involved, we believe we will have to stick with displaying information to a computer screen.

Requirements	Verification
1. Display the current game board clearly, player should be able to understand the general layout of the game. The location of every piece in the game board should be known to the player. 2. Chess board screen visually shows moves made by player on the smart board 3. Chess board screen takes input from opponent and correctly sends the information back to the chess board	1. The computer screen displays a chess gameboard when connected to the smart chess board. The gameboard visually shows the pieces of all in the chess board 2. Make a move on the smart chessboard. Make sure that it is registered visually on the screen. 3. Have player make a move on the computer screen. See if the move is registered on the smart chess board

2.7 Program Flowcharts

High level flowchart:

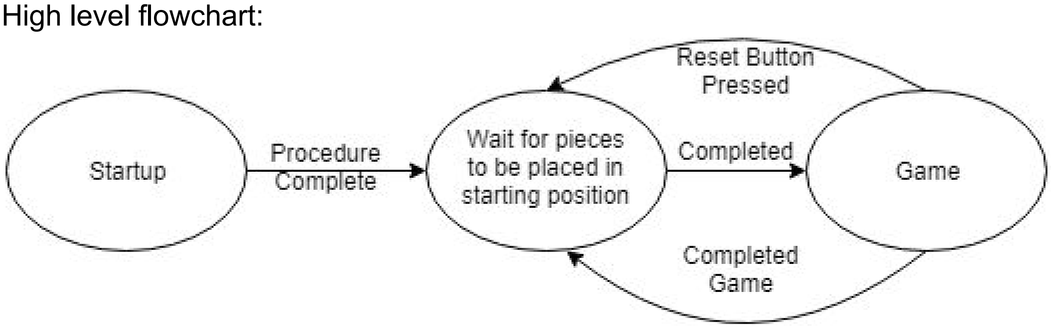

Figure 3. High level flowchart of MCU programming

Flowchart of Game block:

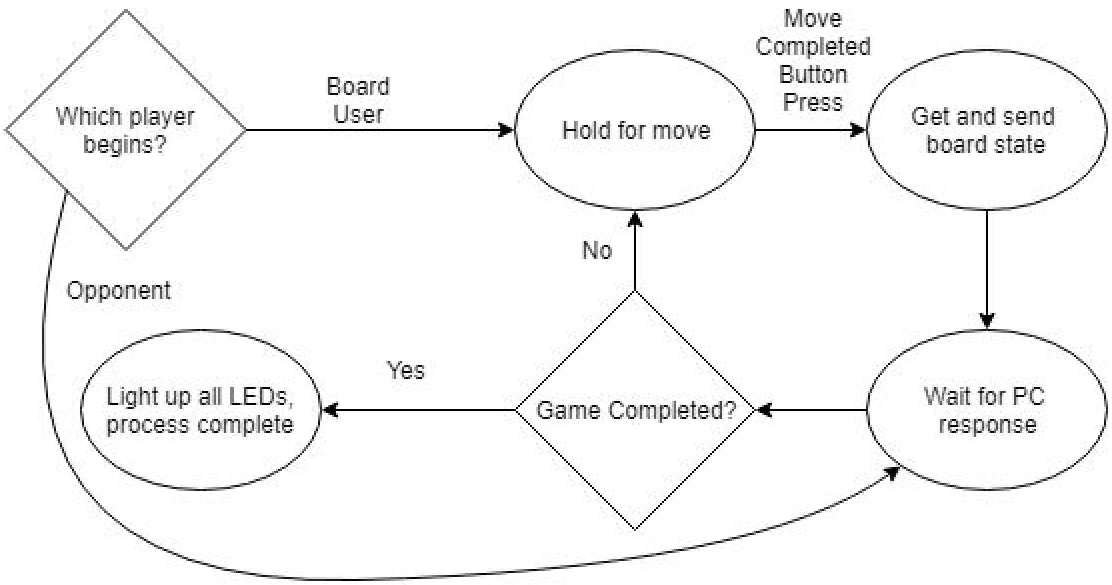

Figure 4. Flowchart breakdown of Game block from high level flowchart

Flowchart for getting and sending the board state:

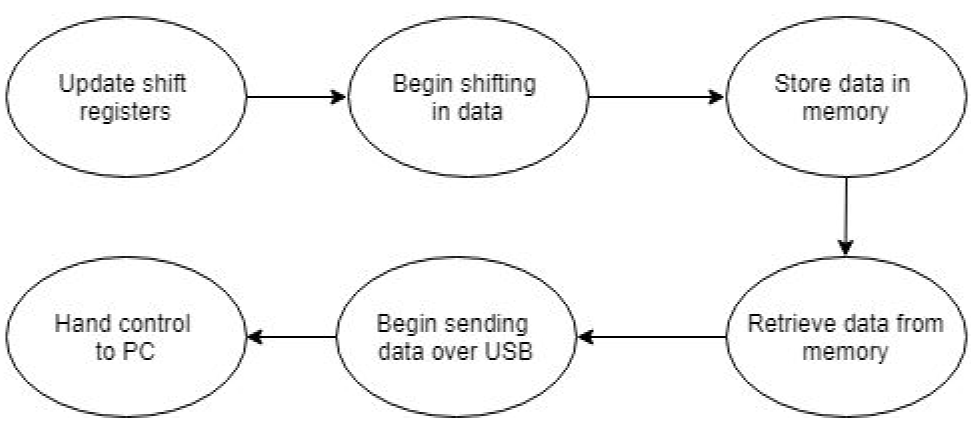

Figure 5. Flowchart breakdown of retrieving data from game board and sending it to PC

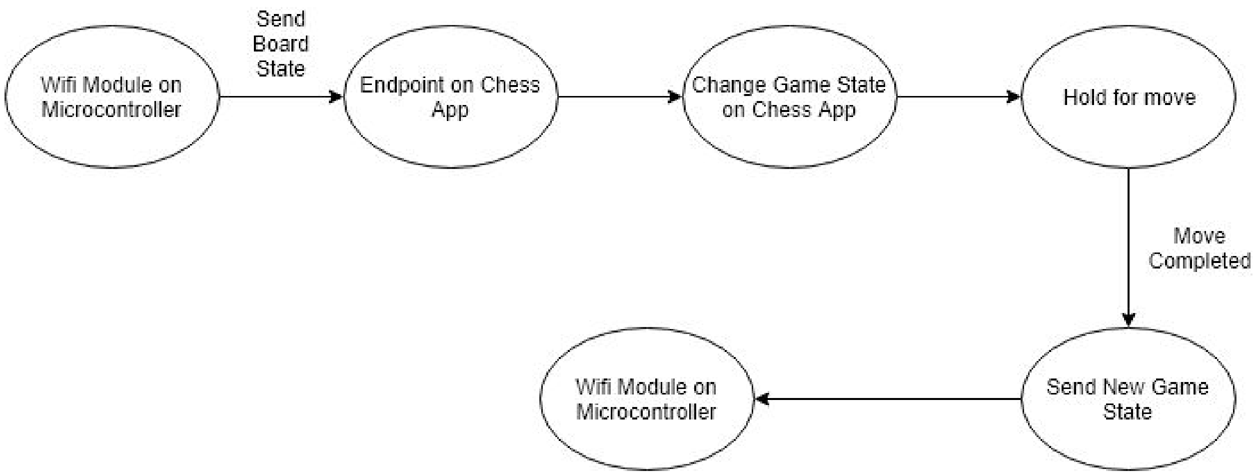

Figure 6. Flowchart of the communication to/from website application

2.8 Schematics

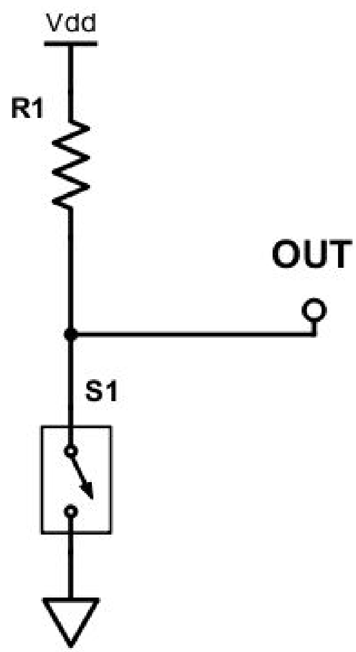

Figure 7. Reed switch schematic

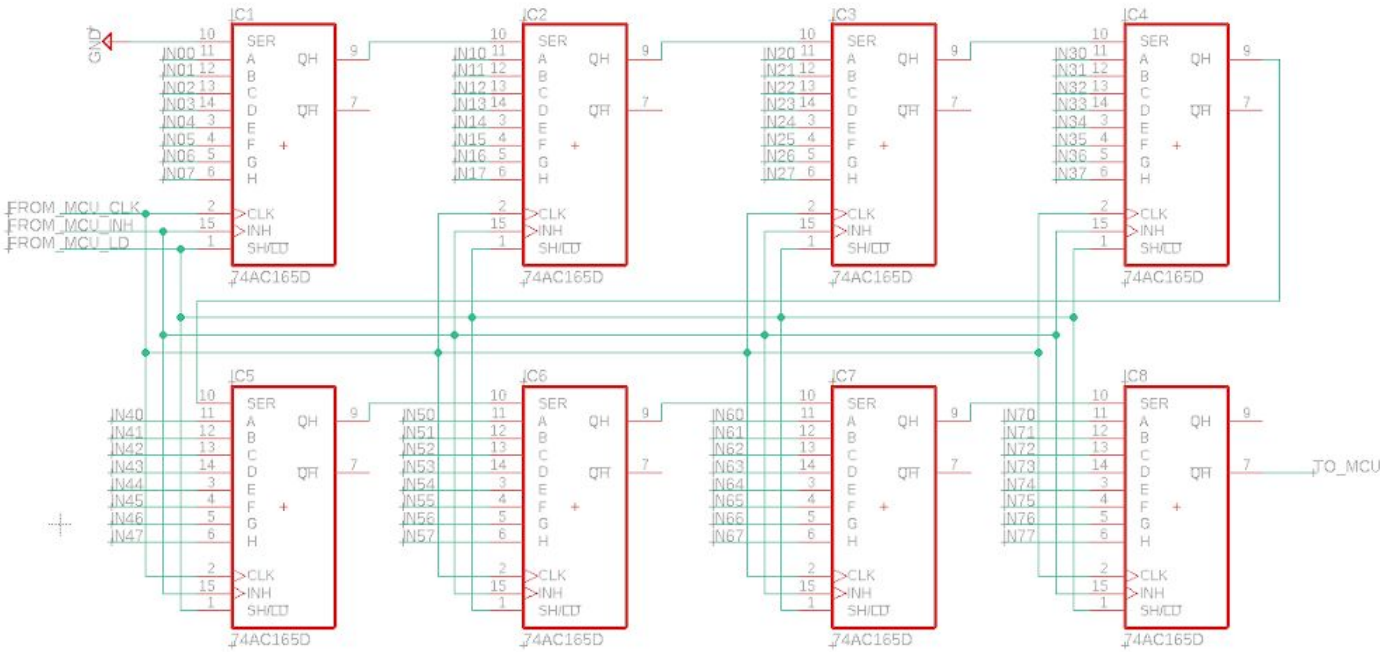

Figure 8. Shift registers connected to reed switches schematic

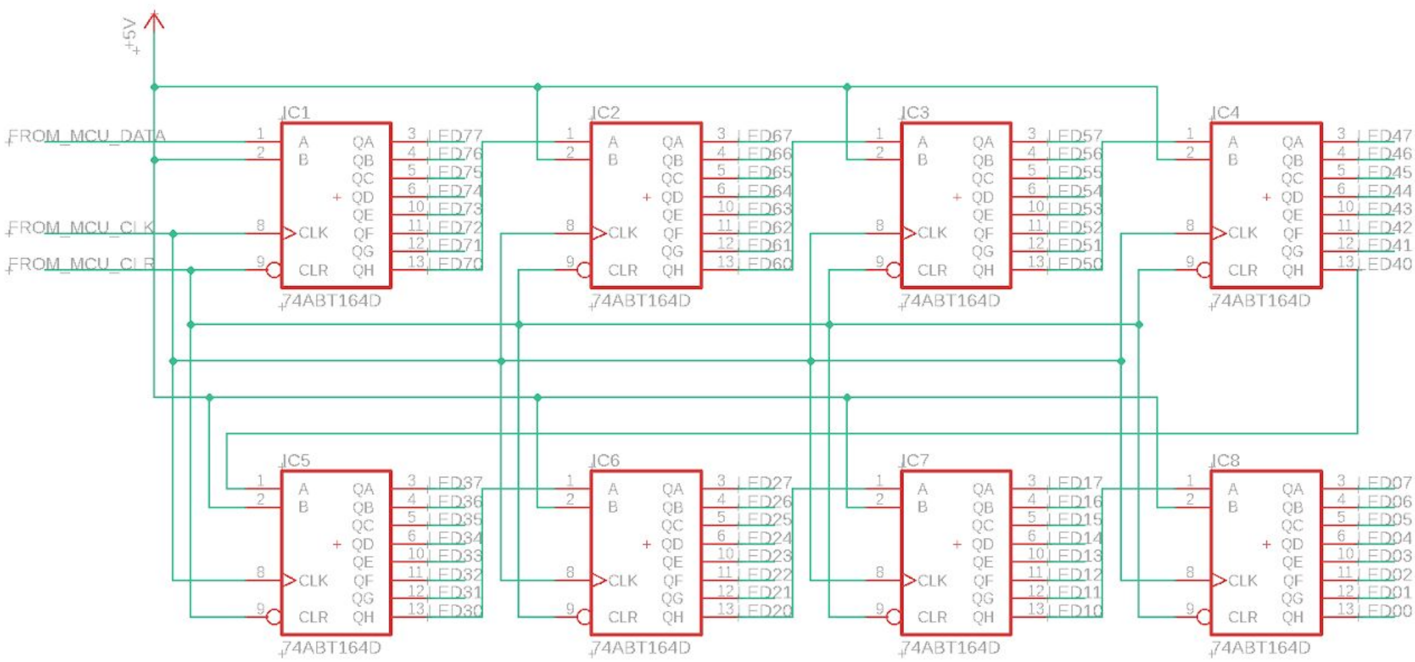

Figure 9. Shift registers connected to LEDs schematic, one color channel

2.9 Tolerance Analysis

The control module is the most crucial module for successful completion of this project. One of the main goals of the project is to gain successful communication between the gameboard and PC so that the player can compete against others from across the world. To successfully accomplish this goal, the control module must be able to take the readings from the sensors and transmit this data to the PC and vice versa. The USB communication between the gameboard and PC does not need to be instantaneous, but it needs to be fast enough to allow a player to make moves as quickly as they wish.

The latency between the player pressing the 'end turn' button and the opponent receiving can be calculated with the following equation:

$$\tau_{total} = \tau_d + \tau_t \qquad (1)$$

Where τ_{total} is the total time it takes from pressing the button to the PC receiving the data, τ_d is the time it takes to collect the data, and τ_t is the time it takes to transmit the data. Td can be calculated by analyzing the architecture of our design and the datasheet given for the PISO SRs. There is a total of 64-bits to be shifted into the MCU. Due to the architecture of our design, only 1-bit can be shifted into the MCU in a single clock cycle. Therefore, the total time taken to collect the data is limited by the clock speed that the SRs are running at. According to the SN74HC165N datasheet, at a Vcc of 4.5 V, the maximum clock frequency that the SR can be run at is 25 MHz. However, this is not the limiting clocking factor. The MCU we chose, the ATmega328, has a maximum clocking frequency of 20 MHz running at 4.5 - 5.5 V. Since we are running the MCU at an input voltage of 5 V, we can safely assume we can get near the 20 MHz maximum clocking frequency. In order to calculate the total data collection time we use the following equation:

$$\tau_d = 66 * 1/f \qquad (2)$$

Where f is clocking frequency in Hertz. A factor of 66 is used because it takes 1 clock cycle to load the data into the shift registers, 1 clock cycle to allow the values to settle, then 64 clock cycles to shift all the bits. Assuming the 20 MHz maximum clocking frequency of the MCU is achieved and passed on to the SRs, this gives us a total data collection time of 3.3 μs. Assuming we are not going to run the MCU at maximum clocking frequency, the data collection time still remains fairly quick. If we were to lower the clocking speed to the MCU internal oscillator frequency of 128 kHz, we would get a data collection time of 515.6 μs.

Estimating τ_t is a much more difficult task. Since we have not currently written our transmission protocol, there is no easy way to determine the number of instructions that the MCU must execute in order to transmit a single bit. Assuming this time is limited by the baud rate of serial transmission and a conservative estimate of this rate at 9600 bits-per-second (bps), it would take 6.67 ms to transmit 64-bits [9]. This will likely be the dominating time factor in determining latency.

A similar calculation can be made for estimating the latency between when the opponent makes a move and when this data is displayed on the LEDs. Using a similar calculation for determining the time it takes for data to be transmitted from the PC to the MCU, it will take approximately

6.67 ms to transmit 64-bits. The MCU will translate this data into information that the LEDs will use. This consists of 24 * 8 = 192-bits of RGB information for all the LEDs. Again, assuming the MCU is running at the max clock speed and the SRs match this clock speed, it would take 9.6 μs for the data to propagate through all the SRs. Assuming a lower clock speed of 128 kHz, this number would change to about 1.55 ms. Transmission of data from the PC to the MCU is still the dominating factor in determining the latency for data transmission.

Even in the scenario where a significantly lower clock speed is used to clock the MCU and SRs than the maximum clock speed, the time it takes to transfer data between the board and the PC is much faster than the speed that an average chess player would make a move.

3 Cost and Schedule

3.1 Labor

30/hour * (50-100 hours to complete)= 1,500 per person *3 =4,500 total

3.2 Parts

Part	Manufacturer	Part #	Quantity	Cost
MCU	Microchip Technology	ATMEGA328P-PU	1	2.14
Reed Switches	Qianxin	Cylewet	7	8.88/10 switches
Buttons	ECE Supply Shop	-	2	3/pc
LEDs	EDGELEC	ED_YT05_RGB-4P-C_100Pcs	1	8.99/100pc
Shift Register (PISO)	TI	SN74HC164N	8	.43
Shift Register (SIPO)	TI	SN74ALS259DR	24	2.71/pc
Voltage Regulator	Ti	LT1084CT-5#PBF	1	8.20
AC/DC Converter	Vacplus	Converter-Vacplus-110V-Power-Adapter(Amazon)	1	14.99

NEED TO ADD MACHINE SHOP LABOR COSTS?

3.3 Schedule

Week	Joel	Ritish	Jeff
2/25/19	Set up endpoints to send/receive data	Begin figuring out what libraries to use, which lanuages to use. Have logistics of chess computer screen figured out	Order parts, verify that parts meet requirements. Finalize mechanical design of board

3/4/19	Finish endpoints and start chess application	Start to program chess board backend logic.	Begin, debug, and test version 1 of MCU programming. Test that each component communicates with MCU properly
3/11/19	Continue chess application	Continue to work on back end	Complete and order version 1 of PCB design
3/18/19 (Spring Break)	Finish chess application	Finish back end logic	Possibly revise version 1 of PCB design to version 2. Begin, debug, and test version 2 of MCU programming with PCB.
3/25/19	Integrate chess application to web	Start working on visually displaying the chess board	Finalize MCU programming, begin integration testing of all hardware with MCU
4/1/19	Start testing with microcontroller	Finish working on visual implementation	Finish testing and debugging of hardware components
4/8/19	Debugging/testing	Integrate computer screen application with chess application	Test and debug chess application integration
4/15/19	Final touches	Finalize project	Finish testing and debugging of chess application integration
4/22/19			Begin Final Presentation
4/29/19			Work on Final Paper

4 Ethics and Safety

Our smart chess board is limited by its sensors on board, and its interface with the internet. As a result, there is not much room for misuse. One minor problem which could arise through intentional misuse if our project were to go out into the industry would be manipulation of the game state. A player could potentially make illegal moves through manipulation of software. This would violate ACM's code of ethics section 1.3 “Be honest and trustworthy” [10].

We aim to adhere as much to ACM's code of ethics especially 1.2 “Avoid harm to others”, so the device is made very safe, there are no sharp edges, dangers of fire hazards or any other major harmful side to the project [10].

Another unlikely, but of our project could be hackers attacking our website. Hackers could potentially put malicious code on our website and use it to their benefit, such as access user's web cameras, or put malicious downloadable links giving access to even more of the user's personal information. This would violate ACM's coffee off ethics section 1.6 “respect privacy”. This would however only be a problem of our product went into the real world after this class. If our product does become more we would make sure to have extra precautions.

References

[1] “Chess Size Guide,” The Regency Chess Company [Online]. Available: https://www.regencychess.co.uk/size_guide.html. [Accessed Feb. 7, 2019].

[2][Online]https://www.digikey.com/product-detail/en/microchip-technology/ATMEGA328P-PU/ATMEGA328P-PU-ND/1914589

[3][Online]https://www.amazon.com/Cylewet-Normally-Magnetic-Induction-Electromagnetic/dp/B01NBPDU04?ref_=fsclp_pl_dp_1

[4][Online]https://www.amazon.com/EDGELEC-Tri-color-Multicolor-Resistors-Included/dp/B077X95LRZ/ref=asc_df_B077X95LRZ/?tag=hyprod-20&linkCode=df0&hvadid=241869999306&hvpos=1o2&hvnetw=g&hvrand=5041759219732655778&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9022196&hvtargid=pla-426004915125&psc=1

[5][Online]https://www.digikey.com/product-detail/en/texas-instruments/SN74HC165N/296-8251-5-ND/376966&?gclid=EAIaIQobChMIl6mNgrXL4AIVEbnACh3Q5gQAEAQYASABEgIK8fD_BwE

[6][Online]https://www.digikey.com/product-detail/en/texas-instruments/SN74HC164N/296-8248-5-ND/376946&?gclid=EAIaIQobChMIkeOW2rbL4AIVkLfACh1rhANSEAYYASABEgIHePD_BwE

[7][Online]https://www.digikey.com/product-detail/en/linear-technology-analog-devices/LT1084CT-5-PBF/LT1084CT-5-PBF-ND/888532

[8][Online]http://www.quick220.com/page/cp-voltage.html

[9][Online]https://learn.sparkfun.com/tutorials/serial-communication/all

[10]Acm.org. (2016). *ACM Code of Ethics and Professional Conduct*. [Online] Available at: http://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct [Accessed Feb. 6, 2019].