1. Introduction

The goal of this project was to create a physical chessboard that is able to interact with a human player, robotic arm and a computer. The game begins when the human player makes a move. The move played by both the robot and the human are fed into a chess engine after being detected by sensors embedded into the chessboard. The chess engine replies with a move, giving the piece's current location and the location to be moved to. The robotic arm moves the piece to the desired position and returns back to its default position to await the human players' next move.

There are three main parts to this project:

1. Design of the chessboard in order to detect the move played
2. Interfacing the chess engine(Stockfish – a UCI chess engine)
3. Design and fabrication of the robotic arm

2. Problem Statement

The purpose of the project was to create a robot playing chess game where anyone can play against a robot. This is intended to be used by student visitors to the technology center to use this and learn the nuances of robotics, magnetic sensors, circuits, and chess moves.

There have been many projects that have been done on chessboards. None of the ones I have referred (highlighted in References) have addressed the project looking at the following aspects:

1. Cost effectiveness of the solution
2. Robustness of the solution
3. Ease of implementation
4. Explanation of mechanics
5. Usability by young kids – who will be interacting with is in the Technology & Science center
6. Reusability of the solution for future projects

The solution that will be designed has to address all of the 6 aspects above to make it useful for the center and for the visiting kids who will be using it.

.

3. Technologies and Techniques Used

3.1 Components

Component	Sty
1. Chess board	
2. Chess pieces	
3. 10mm x 1mm Neodymium Magnets - 32 qty	
4. Reed switch sensor - 64 sty	
5. 1N4148 diode - 64 sty	
6. Ribbon wires	
7. Arduino Mega	
8. Raspberry Pi 3	
9. 3 or 4 mg996R Servos - For Robotic arm	
10. Gripper / Electromagnet - For picking up chess pieces	
11. 12V Power Supply	
12. 10ohm resistors – 64 qty	

3.2 Sensor Selection

There are two options. Each has their pros and cons. The table below lists the pros and cons if using the sensors.

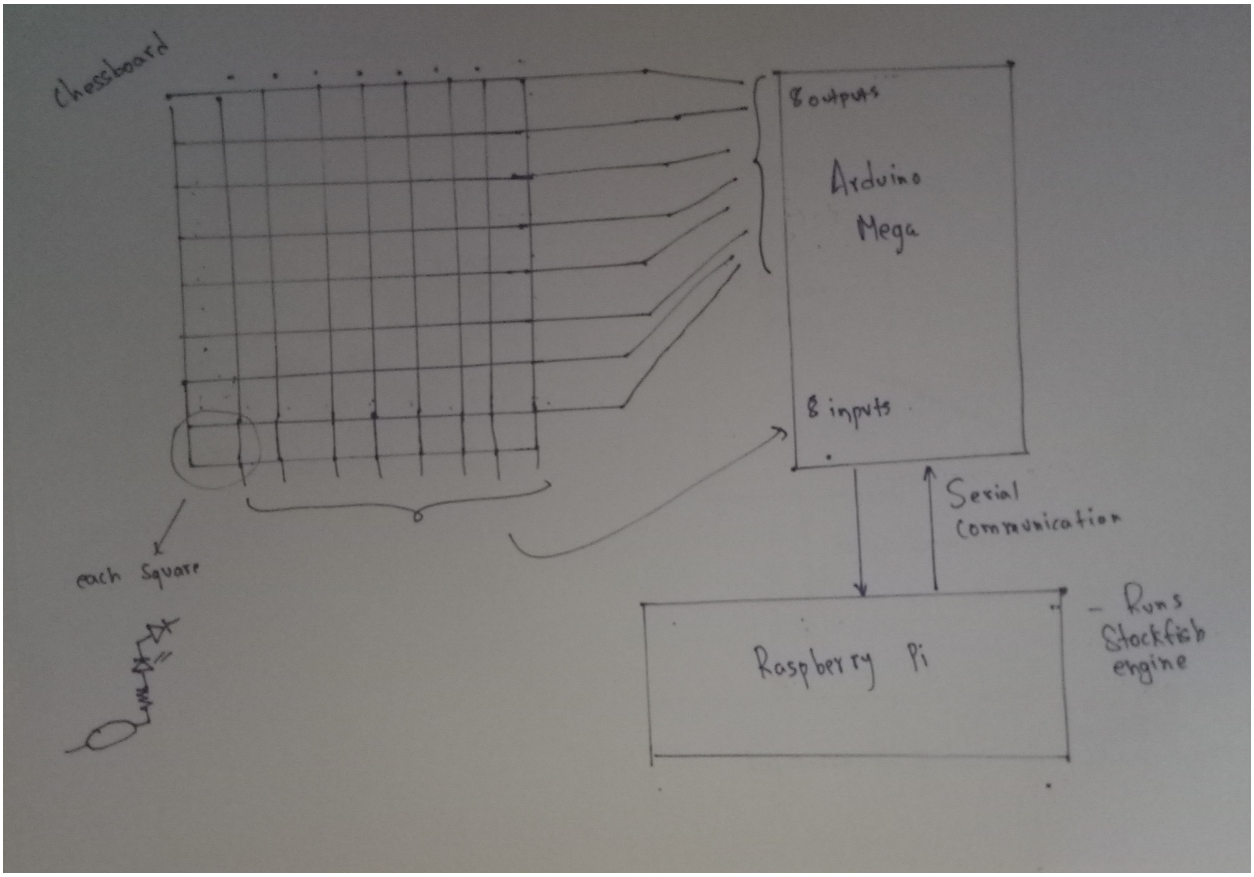
Comparison between using Reed switch sensor and RFID to identify chess piece location

	REED SWITCH/HALL EFFECT SENSOR	RFID
Prices	Sensors ~ Rs 1200 / 2560 Multiplexers ~ Rs 800 Resistors ~ Rs 50 Magnets ~ 900 Power Supply ~ 400	RFID tag ~ 6400 RFID reader ~ 400 Motors to move the reader ~ RS 500 Power Supply ~ 400

Sensitivity	Extremely sensitive, debounces are possible. (-)	Fairly sensitive. (+)
Reliability	Connections and wiring and processing is a lot	Extremely reliable, since there are not much connections
Longevity	In the long run connections will last longer.	In the long run there can be problems with the movement of the reader.
Ease of implementation	Pretty difficult to implement since there are a lot of circuits to build.	Extremely easy to implement.

After going considering the pros and cons for the two sensors, I picked to use the Reed switch sensors

3.3 Design



The chess board consists of 64 reed switches arranged in an 8 x 8 matrix with a total of 16 wires, where 8 are inputs and 8 are outputs as shown in the figure above. We have a 8x8 matrix, on

which we will have to perform continuous sampling to detect the changes which will be done as follows:

- Connect the rows to 8 pins of the microcontroller, setting them as OUTPUT, and initializing them all to '1' (High).
- Connect the columns to 8 pins of the microcontroller, setting them all as INPUT_PULLUP.
- Set to '0' (Low) one row and read the value of all columns. Do the same with the other seven rows, one by one.
- In every full sampling of the board (of the 8 rows), we obtain a matrix of 8 x 8 bits.
- Detecting changes that occur in the matrix at every sampling, we can know what piece was moved.

Once we get an 8x8 matrix of 1’s and 0’s where 1 represents that a piece is not on the square and 0 represents that a piece is on the square. We generate the move played and send the move to a computer i.e the raspberry pi which runs an UCI engine called Stockfish which has functions capable of returning the next best move for the opponent/robot to play and capable of validating the move made by the human.

The code written for interfacing the reed switch sensors with arduino was written in C and the code for interfacing the chess engine was written in python.

4. Overview of Approach

4.1 Project Phase:

This project will be developed in two phases

Phase 1: The computer chess pieces are moved by the human based on a chess engine

Phase 2: The computer chess pieces are moved by a robot based on a chess engine

4.2 Project Plan

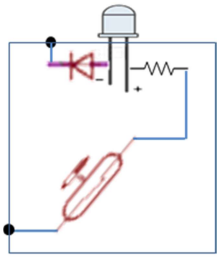
Project completion: 3 weeks

Week1	Run stockfish engine on raspberry pi and test if it can generate the next move after giving a FEN string as the input.
Week2	Construct the chessboard with the sensors and diodes and make connections between Arduino and raspberry pi.
Week3	Test and make sure phase 1 is completed, if there is remaining time 3d print the robotic arm and program the arm to move the pieces according to the stockfish engine.

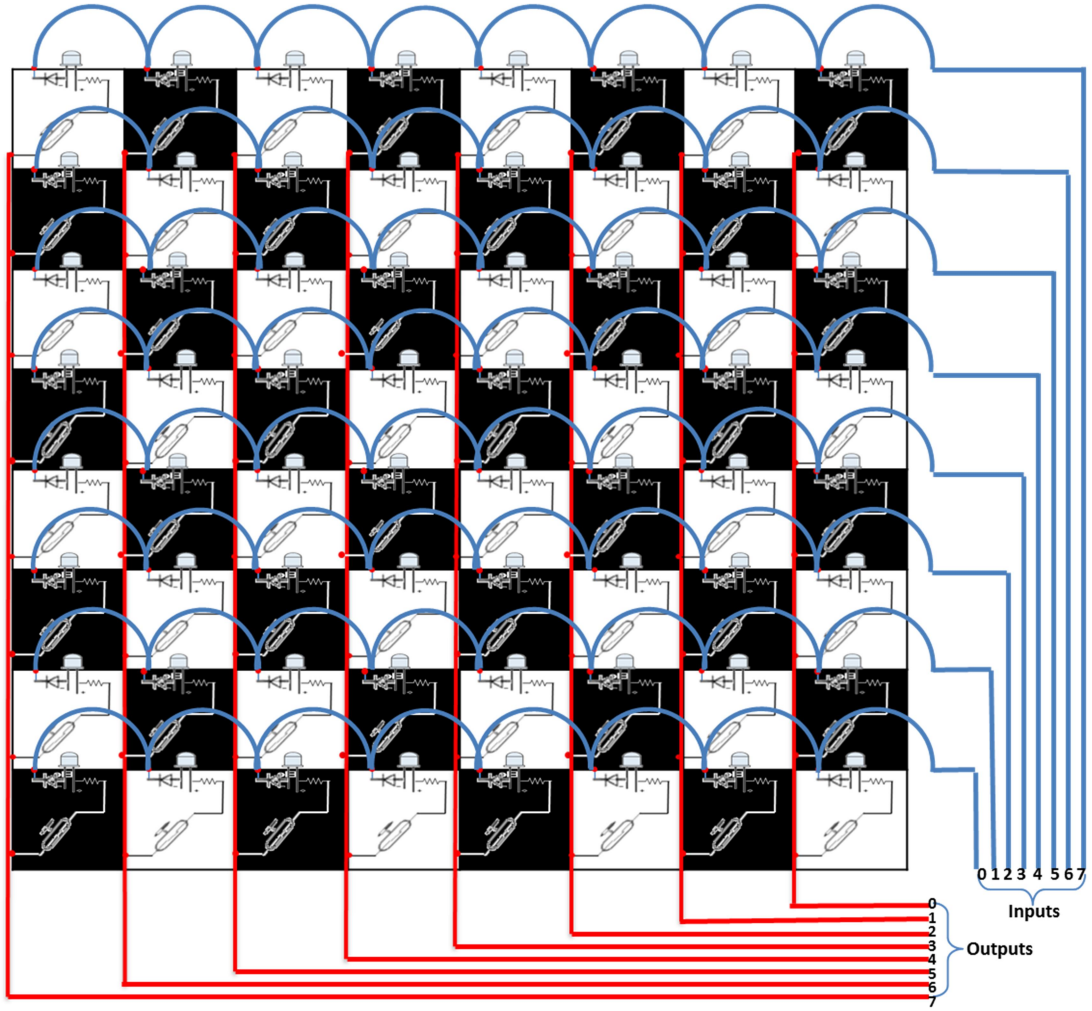
4.3 Project Progression

4.3.1 Design the circuit layout

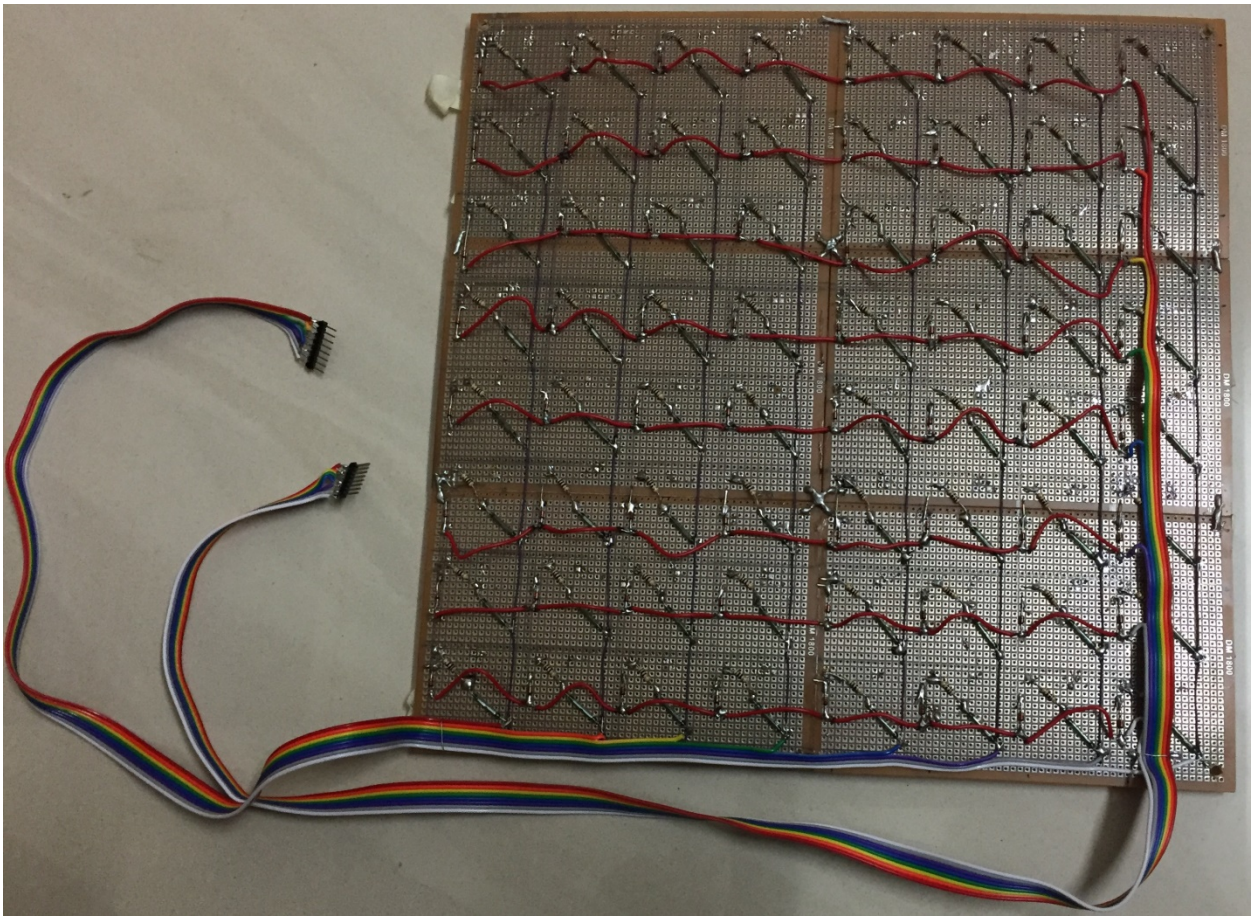
Single Square Unit



4.3.3 Build the circuitry



4.3.4 Circuit Board



5. Conclusion

It has been a great learning doing this project from many aspects. Some of the key learning's from this project are:

Plan every detail – Plan everything ahead. Be it designing the circuit layout or buying components or putting the components together or testing the circuit. Having a good plan that progressively takes you to the goal is the best way to achieve success. I was able to utilize it in this project for success.

Start Small - I started with a small 2X2 matrix and not only tested the components but also the code and the logic behind this. I was able to iron out all issues and faced very little issue when I did the complete board

Prototype - Even with the a large board, I tested every row before proceeding with the next row.

Make it easy to maintain – We all know that electronic components do fail when exposed to moisture and dust. Since this project will not be enclosed in an enclosed case, there is a chance that components may fail. I soldered the components in such a way that it will be easy to replace the component if any component fails.

Understand the logic behind everything you do. – It would have been easier to follow some of the similar projects made by other people, whether it was laying out the components or the components to use. Instead I tried it with various resistors,diodes and LED's and noticed the change in behavior and picked the right combination of component's to make it very robust. It was a good learning for me to understand the electronics and do the project.