Design and Implementation of a Wireless Remote Chess Playing Physical Platform

Divye Bhutani[1], Yusuf Ali[2], Pratyush Gupta[3]
Delhi Technological University, Shahbad Daulatpur, Delhi, India

***Abstract* - The Internet of Things is the interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data. The chess robot is an innovative implementation of remote game playing platform for chess enthusiasts. The primary motive of building the chess robot was to provide a physical, mobile and economical platform to chess players to play chess in real time against opponents anywhere in the world so long as there is an internet connection. Two complete chess robots have been designed for analysing and testing the multiplayer interface between two players. They have been constructed within a highly constrained budget of INR 8500 (approx. 130$). The full-length paper discusses the various considerations and underlying principles that have been incorporated in the development of the chess bots. The paper also draws light on the choice of the components that have been included in the chess bots and the specific rationale behind choosing the same. An exhaustive flowchart explains the various steps involved in replicating a certain player's move on the opponent's chess board. In conclusion, the several problems encountered during the assembly and development of chess engine of the chess bots have been put before the readers.**

Keywords - Robot, chess, internet of things (IoT), wi-fi shield, remote game playing platform, chess engine, microcontroller, Arduino, multiplexer

I. INTRODUCTION

Chess - one of the most widely played games in the world – originated as "Chaturanga" in ancient India. The desire to automate the game of chess dates back to the 1770s when Wolfgang von Kempelen invented the "Mechanical Turk". The Turk – as it is popularly known - consisted of a mechanical man dressed in robes and a turban who sat at a wooden cabinet that was overlaid with a chessboard. The Turk was designed to play chess against any opponent game enough to challenge him operated via a concealed operator, who controlled each movement from inside the cabinet by candlelight, pulling levers to operate the Turk's arm and keeping track of the moves on their own board. Not only did the Turk manage to play chess on its own, it did a great job at doing so. For years, people remained dumbfounded at the works of the Turk. Eventually, in the year 1840, Robert-Houdin established that the Turk was operated via a concealed operator, who controlled each movement from inside the cabinet by pulling levers to operate the Turk's arm and keeping track of the moves on their own board.

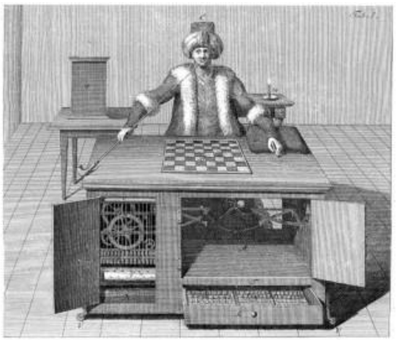

Figure 1: Mechanical Turk or Automaton Chess Player

The chess robot described in this paper has been designed to delve deeper into the world of human-robot interactions. It serves as a remote-game playing platform and eliminates the need of online chess platforms which take away the true essence of the game of chess. In addition, it greatly reduces the physical strain to the vision of the players who spend hours staring at their desktop screens while playing chess. Another important application of the chess bot would be in facilitating chess coaches to mentor players of substantial potential without missing out on the true experience of playing traditional chess.

A crucial effort has been made to build a cost-effective platform which is easily affordable to the daily customer. Clearly, utmost attention has been paid to ensuring quality and user-friendliness of the chess robot.

The paper has been segregated in the following fashion:

1) Mechanical Structure
2) Electrical Hardware
3) Software Implementation

II. MECHANICAL STRUCTURE

A. Design

Extensive research and analysis was carried out on determining the components and mechanics of the entire system. A plethora of parameters such as the exact playing area, number of stepper motors to be used, placement of pulleys and timing belts, robustness and reliability of the framework and total cost of the materials were taken into account. A few of them are discussed below:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

1. Playing area: In totality, there were two constraints which dictated the dimensions of the playing area. The need to build a compact and mobile platform necessitated the playing area dimensions to be as concise as possible. In contrast, a closely-packed area implied higher levels of accuracy in the actuation of chess pieces and minimal margins of error. On a comprehensive analysis of the aforementioned parameters, the side of each square on the chess board was set to be 5 cm – enabling us to maintain the compactness of the chess robot and reach accuracy levels well in the practical limit. Including an extra 2.5 cm of width along each edge for stationing the eliminated chess pieces, we end up with a 45 cm x 45 cm chess board. Including the extra space for the stepper motors and pulleys, the exact dimensions of the chess robot add up to 70 cm x 70 cm.

2. Number of stepper motors: Fig. 0 denotes the X and Y axes in the chess robot framework. A minimum of two stepper motors were necessary – one each for the X and Y axes' motion. But upon closer analysis, it was observed that a single motor would not be sufficient to produce the Y axis motion as there is a lag in the power transmission from one arm of the bot to the other, thus producing a turning moment about the instantaneous centre of mass of the bot. To eliminate this complication, two synchronised motors were required for efficient movement on the Y axis. Hence, a total of three motors were incorporated in the chess robot.

3. Placement of pulleys and timing belt: The placement of pulleys in the CAD was dictated by the position of the motors. The prime motive was to ensure the minimal slack in the timing belt which would facilitate efficient transmission of the drive from the motors.

Various other parameters such as choice and design of framework are discussed in the following sections. The design phase was followed by the developing of an entire CAD model of the chess robot in the SolidWorks software. Fig. 2-6 shows the solid model of the chess robot.

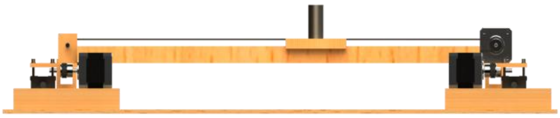

Figure 2: Front View of Assembly

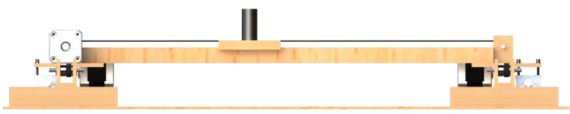

Figure 3: Back View of Assembly

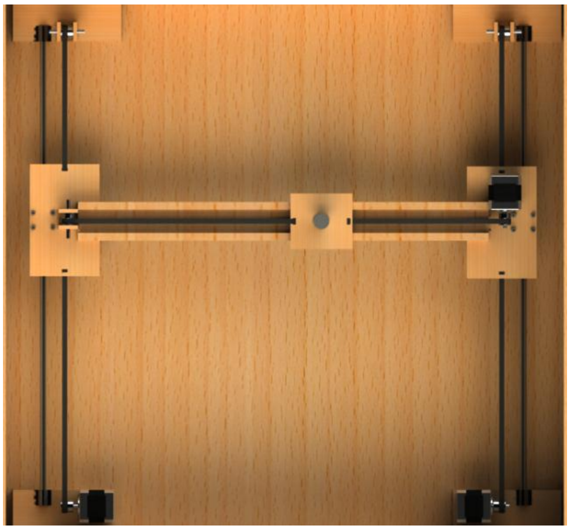

Figure 4: Top View of Assembly

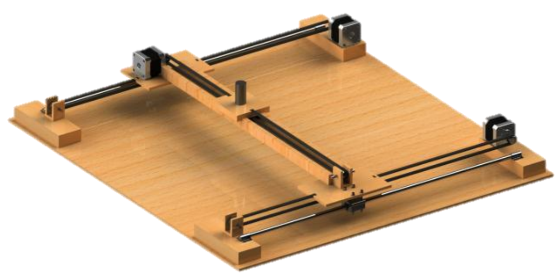

Figure 5: Isometric View of Assembly

Figure 6: Isometric View of Complete Model

B. Components Required

The essential mechanical components that were required include the following:

1. Framework: The basic purpose of the framework is to provide a sound and durable structure in which the various components can be enclosed. It is the primarily responsibility of the framework to provide support to the various working parts in the chess robot. Hence, it is thoroughly imperative to ensure that the framework is as sturdy and firm as possible. Quite a few materials such as plywood, medium-density fibrewood (MDF), aluminium were considered for the construction of the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

frame. The choice of the material to be used was dictated by two factors – the cost effectiveness and the ease of machinability. Medium-density fibrewood was chosen as it satisfied both the criteria simultaneously.

2. Pulleys: The pulley and timing belt work in conjunction to form the transmission system of the chess robot. They help in transferring the drive from the shaft of the motor to the movable platforms which are used to actuate the chess pieces on the board. A 20-teeth GT2 pulley with a 5 mm bore (the motor's shaft had an outside diameter of 5 mm) was used as it was quite responsive in transmitting the drive and provided just the right amount of speed as well. Fig. 7 shows the pulley that was used [1].

Figure 7: Pulley used in assembly

3. Timing belt: As mentioned before, the pulley and timing belt work together to constitute the transmission system of the chess robot [1]. A 6 mm wide GT2 timing belt with a 2 mm pitch was used as its profile seamlessly meshed with that of the GT2 pulley's, resulting in a reasonably responsive transmission system. Fig. 8 shows the timing belt used with the pulley. The pulley and belt system is shown.

Figure 8: Timing Belt used in assembly

4. Linear bearing blocks: In order to provide for a smooth movement along the Y axis, linear bearing blocks were required. Two standard aluminium linear bearing blocks with an inner diameter of 8 mm were used. These were bolted onto the movable platforms from the underside using 3 mm nuts and bolts. Washers were used as cylindrical spacers to ensure that the timing belt did not slack. Fig. 9 shows the linear bearing blocks used in the construction.

Figure 9: Linear Bearing Blocks

5. Guide rails: Two 8 mm stainless-steel rods were used as guide rails for the previously mentioned linear bearing blocks. The stainless-steel rods and bearing blocks worked in conjunction to produce smooth, non-erratic movement along the Y axis. Fig. 10 shows the stainless-steel rods used in the framework.

Figure 10: Guide rails used in assembly

C. Machining of Framework

Medium-density fibrewood (MDF) was chosen as the raw material for the construction of the framework. Keeping in view the durability and robustness that an MDF framework would provide, an additional reason for the above consideration was the cost-effectiveness and easy availability of the MDF in the market.

However, the key basis of choosing MDF was its extreme ease of machinability. MDF can be easily machined into any 2D profile by the CNC laser-cut process which itself is quite cost-effective in its own. Although, aluminium is also easily machined into diverse products through the process of CNC vertical and horizontal milling, the material is a lot more expensive in comparison to the MDF of the same quantity. Other additive manufacturing techniques such as 3D printing were also considered but on a comprehensive analysis, it was established to be less feasible in comparison to the laser-cut machining of the MDF.

As they were the most abundantly available variation of the MDF in the market, 5 mm MDF sheets were purchased which were to be machined to construct the framework. The laser cut process included the providing of the digital 2D profiles which were to be machined onto the MDF sheets.

D. Assembly of Framework

Once the laser-cut products were obtained, they were assembled according to the 3D CAD assembly originally developed in the SolidWorks environment. In the multiple locations where adhesion was required, "Fevikwik" was used as it sufficed the purpose – thus, eliminating the need of incorporating expensive adhesives like the commonly available Araldite standard epoxy resin. The CAD assembly greatly facilitated the physical assembly – allowing for smooth and effective integration of all the components. The final assembled chess robot is shown in Fig. 6.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

Table 1: Component-wise cost split-up

Sl. No	Component	Quantity	Cost
1	Framework - Raw material and machining	-	Rs.1500/-
2	GT2 Pulleys	6 units	Rs.140/- per unit
3	GT2 Timing Belt	2 metres	Rs.160/- per metre
4	Acrylic sheet	70 cm x 70 cm	Rs.400/-
5	Linear bearing blocks	2 units	Rs.450/- per unit
6	Stainless-steel rod	2 metres	Rs.60/- per unit
Total Cost			Rs.4080

E. Problems Encountered

1. Laser-cut products off-dimensions: On closer inspection of the laser-cut products, it was observed that each dimension was offset by 1-1.5 mm from their set values in the CAD. This was due to the fact that the laser beam that cuts through the wood has a finite specified diameter (approximately, 0.1 mm) – thus, refraining it to cut the work piece in a pint-like fashion. This complication was solved by making alterations in the physical assembly.

2. Strong attraction between neodymium magnets: In an attempt to actuate the chess pieces using the electromagnet, a strong neodymium magnet was attached on the base of each chess piece. On testing, it was found that the neodymium magnets under adjacent chess pieces were strongly attracted to each other – thus, disrupting the motion of the pieces. This problem was solved by using weaker ferrite magnets with a smaller radius. The use of ferrite magnets resulted in smooth and non-disturbed actuation of the chess pieces.

3. Toppling of chess pieces: The chess pieces when actuated by the electromagnet were observed to topple over owing to the gap created between the base of the chess piece and the surface of the chess board. This impediment was solved by embedding the magnets into the base of the chess piece and thus, eliminating the gap between the base of the chess piece and the surface of the chess board.

4. Asymmetric weight distribution due to X axis motor: On close observation of the setup, it was observed that the X axis motor resulted in an uneven weight distribution in the system which led to the slacking of the timing belt and inefficient drive transfer. An additional weight was added to the arm without the motor to even out the weight distribution.

III. ELECTRONIC HARDWARE

A. Hardware Design

The electronic circuitry of the robot needs to be reliable and well-built for optimal performance. The circuit needs to be elaborate and concise at the same time for proper functioning with minimal hardware dimensions.

The hardware setup for this robot was implemented on a perforated board. Even though there was an option to go for printed circuit board (PCB), perforated board was utilised due to its simplicity and ease of debugging. Breadboards offered even simpler connections but they are bulky and not used for finished circuits.

The complete hardware has two stages. The first stage is having stepper motor setup. Stepper motor drivers were the main components to be used in this circuit. Two different stepper motor drivers were used in our setup. One was used to drive the pair of Y-axis motors. The motors were used with one driver in order to have synchronous movement. X-Axis motor was driven by a different driver.

Second stage is sensor circuitry. Hall sensors (WSH 130) were utilised for this purpose. The main challenge in this stage was the sheer number of sensors to be interfaced. 64 hall sensors were fixed in each square of the board. The sensor outputs were multiplexed in order to reduce the number of GPIOs required for processing purpose. The whole stage was neatly assembled beneath the chess board. The microcontroller used was further interfaced with ESP8266-01 to connect our robot to the internet and enable gameplay with the remote player.

The key components of the chess robot:

1. Motors

The variety of motors available in the market has different operating parameters, dimensions and affordability which are crucial for selection. In our case we needed affordable motors which had less complex control mechanism and high angle precision. Unlike most of the other DC motors, stepper motors can be positioned exactly as required. Combined with low cost and low power requirements, this motor suffices the criteria for this robot. Although servo motors are way more accurate given they work on a feedback mechanism, but they don't provide 360-degree rotation. Servo motors which do provide full rotation are beyond the scope of this project given the cost and availability.

Stepper motors find widespread application in the field of mechatronics. The stepper motors are precise as they can easily perform micro stepping.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

Figure 11: Stepper Motor

2. DRV8825 (Polulu High Current Motor Driver)

Stepper motors are highly efficient motors as they are ideal for high torque applications with precise angle control. But the motor requires higher voltages and currents for operation compared to the TTL logic (Transistor-Transistor Level), which makes its interfacing with any microcontroller an issue. The stepper motor driver here acts as a translator which makes control of motor quite simple. Driver selection also presented a hurdle. The stepper motor is usually a high current operating device which makes use of heat sink for the driver mandatory. Also the driver gets fried due to over currents while performing high current operations. We were presented with two choices, Polulu DRV8825 [4] and A4988. A4988 [3], though cheaper and widely used, is having complex wiring requirements and is usually not capable of handling current spikes which is one of the major reasons of circuit failures. Henceforth, DRV8825 was selected given its thermal shutdown feature and overcurrent protection. The driver is easy to interface and properly electrically isolated, ideal for long operating hours.

Figure 12: DRV8825 motor driver

3. Power Supply

The two sets of motors (X-Axis and Y-axis) are powered from the same source. A separate relay circuit is used for proper isolation This helps in providing sufficient current to the operational motor. The electromagnet circuit is powered separately in this robot.

4. Arduino Mega 2560 R3

The microcontroller is the main digital processing unit of any robotic hardware. The main challenge in determining the right controller is the cost, the availability and ease of code execution [5]. Arduino is having an open-sourced IDE which makes wide range of libraries available for hardware support. A strong community support also encourages innovation and provides first hand user knowledge.

Figure 13: Arduino Mega 2560 microcontroller

5. Hall Sensors

Sensors were required in the execution process to determine the exact location of the chess piece. The pieces had magnets fixed below them in order to be carried around by the electromagnets. In order to make a contact less sensor that is reliable and affordable at the same time, it turned out that by sensing the presence of magnetic field we can determine the position of the piece.

Hall sensors are sensors which can sense the change in magnetic field [2]. WSH 130 was the sensor chosen in particular. Cost-effective and reliable digital output made it the perfect match making processing and circuitry simple yet effective at the same instance. Other sensors were either expensive or weren't giving digital signals which made them seemingly unattractive choices.

Figure 14: Hall sensor WSH130

6. Multiplexer

Due to limited GPIOs (General Purpose Input/Outputs) of the microcontroller being utilised, multiplexers were incorporated in the hardware setup of the robot [5]. Multiplexers can be utilised where there are multiple inputs and outputs which cannot be linked in the system together due to a number of reasons. So the multiplexers are simple shift registers where these inputs are provided in the system by means of digital switching.

74HC4051 fulfilled all the requirements and was reliable and affordable. Other multiplexers available with larger number of channels had a steep increase in price. Though they provided us with concise interfacing, the shoot up in the cost was drastic in comparison to the utility offered.

7. Electromagnet

The choice of using an electromagnet came quite naturally as the pieces had magnets attached below them. It seemed natural to utilise this for actuation. The pieces can be easily slid across the new board to their new position without much mechanical expertise. The electromagnet was designed using a soft iron core which was chosen due to its high permeability. The wire of 40AWG gauge was chosen to make the windings.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

8. ESP8266-01Wi-Fi Module

This tiny yet powerful device enables our robot to connect to the internet. ESP 8266 installed in one of the devices can be used to provide the actuation stats of player to another one over internet. In this way our one user can communicate with a remote player using Wi-Fi and plays the game of chess physically in real time.

Figure 15: ESP8266 Wi-Fi Module

Table 2: Electronic Component-wise cost split-up

S No.	Component	Quantity	Specification	Cost(Per Unit)
1.	Stepper Motor	3 nos.	Nema 17 Stepper Type,12V, 1.5mA	Rs.350
2.	Arduino Atmega 2560	1 no.	Operating Voltage 5V	Rs.800
3.	Power Supply	2 no.	12 Volts, 2 A adapter	Rs.120
4.	DRV8825 (Polulu High Current Motor Driver)	2 nos.	Operating Voltage 12V	Rs.200
5.	Hall Sensors (WHS 130)	64 nos.	Operating Voltage 5V	Rs.15
6.	74HC4051 (8 Channel MUX)	8 nos.	Operating Voltage	Rs.20
7.	ESP8266-01	2 nos.	Operating Voltage 3.3V	Rs.200
Total Cost				Rs. 4030

B. Component Selection Parameters

1. Stepper Motor Selection as per Torque

Stepper Motors have a torque v/s rotational speed graph which represents the inverse relationship between the two quantities. Torque provided by the motor is inversely related to the rotational speed of operation. As the speed of the motor increases the torque output of the motor decreases. The torque required is dependent on the mass of assembly and the drive system. It is suggested that motors should operate at maximum torque in order to have a smooth operation.

Motors which provide higher torques tend to be heavier and also require higher power for optimal driving. NEMA 17 were found to fill all the desired characteristics of being lightweight and robust enough at the same time.

2. Stepper Motor Selection as per No. of Steps

Stepper Motors are pretty accurate when it comes to angle precision. A stepper motor takes 200 steps to complete one revolution in full step operation. This provides a resolution of 1.8 degrees per step. But stepper motors can be far more accurate in micro stepping mode. In Micro Stepping technique, we provide variable controlled current to the motor in sine waves. This allows smooth functioning of rotor, decrease the stress of the parts and increase the accuracy of motors.

Table 3: Micro stepping Resolution [4]

Operation	Steps Available	Resolution(Degrees per Step)
Full Step	200	1.8
Half Step	400	0.9
Sixteenth Step	3200	0.1125

C. Sensing

1. Hall Sensing Principle

To guide the chess pieces, we required a sensor which would detect the change in the magnetic field. Hall sensors are the sensors which can reliably detect the change in magnetic field [2]. Hall sensors work on the Hall Effect phenomenon. In this effect basically the presence of external magnetic field produces a voltage which is perpendicular to the applied voltage. This sensor measures that voltage. In this way we can measure the relative strength of magnetic field present.

In the absence of magnetic field, the sensor outputs a low whereas as soon as it detects the field, it triggers a high.

2. 74HC4051 (8 channels Digitally Controlled Analogue Multiplexer)

64 sensors together pose a challenge when it comes to interfacing. Multiplexers are basic shift registers and can multiplex multiple inputs into just one output. This particular IC (Integrated Circuit) allows us to use such a large number of sensors even with limited GPIOs. This is crucial in order to assure enough processing memory to the microcontroller also. Here outputs of eight different hall sensors are reliably multiplexed into one output making wiring and processing more efficient.

D. Problems encountered

1. While setting up the connections, unsoldered ends presented a tough time. Shorting of free ends of wires eventually lead to fried components. One driver was rendered redundant due to this blunder. So tinned ends of wire are mandatory while making connections.
2. Proper current regulation is a must in this case for overcurrent protection for the motor driver. To avoid frequent shutdown of the motor driver, Polulu (Driver Manufacturer) provides a current regulation technique which can be appropriately set using the on-module potentiometer.

While powering the motors, the two sets of motors were being powered at the same time. This led to reduced availability of current for the respective motors making smooth functioning of the robot an issue. To overcome this, a relay circuit was used for the two motor supplies which ensure current isolation for the two sets separately providing equal current, sufficient for smooth driving.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 09, September - 2017

IV.SOFTWARE IMPLEMENTATION

Chess robot is realization of Chess game software on a physical platform which is achieved by calibrating the mechanical setup, mapping the chess pieces on the chess board, validating the move performed by the player followed by transfer of data over Wi-Fi to the remote player and replicating the move on remote chess board.

A. Calibration of Mechanical Setup

The traversable area (45cm x 45 cm) on the mechanical platform incorporates the chess board (40cm x 40cm) in middle and peripheral area is used as parking spaces for the captured chess pieces. The stepper motors have to be calibrated in order to actuate the precise movement of the chess piece. This is done by assuming one corner of the traversable area as the reference point. The distance traversed by the guide rail attached to the stepper motors with its unit step rotation is measured and recorded, henceforth a mathematical formula is derived which helps to program the motors to get precise access to any location on the chess board.

Let x be the distance traversed by the slider on the guiderail with unit step rotation of motor and n be the number of steps required to cover the distance of half of edge length of a cell on chess board.

Let x1, y1 be the initial coordinate of the chess piece on the chess board and x2, y2 be the final coordinate of the chess piece on the chess board.

An iterative function is formulized for generating the path of chess piece.

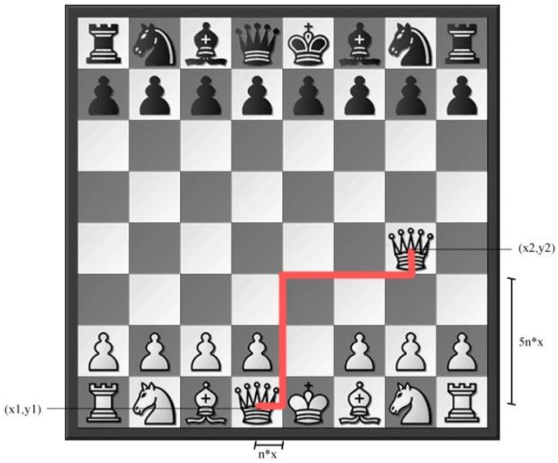

Figure 16: Calibration of motors

B. Mapping the Chess Pieces

Chess robot is designed to deliver traditional chess playing experience to players unlike online gameplay on personal computers. Every chess piece is embedded with a magnet and under each cell on the chessboard a Hall Sensor is fixed which senses the presence of a chess piece in that particular cell. The initial configuration of the chess pieces is known and any changes made in the configuration during the game is recorded and updated in the memory. s

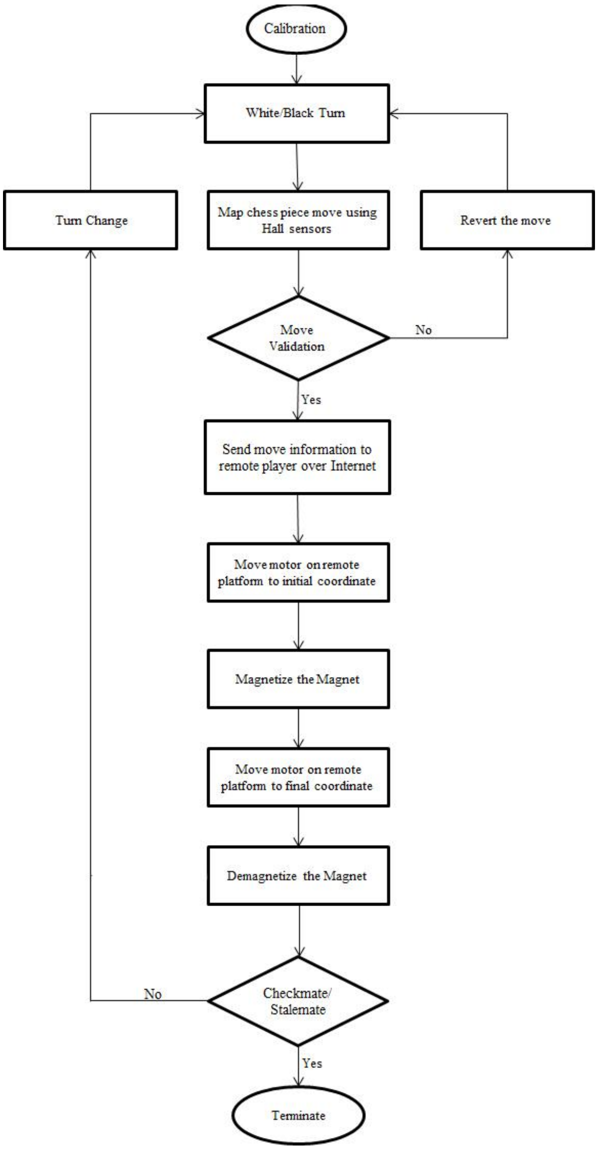

Figure 17: Working Algorithm Flowchart

C. Validation of Chess Moves

Every chess piece has precise rules for its movement. In addition to movement, rules governing Piece Capturing, Castling, En Passant, Pawn Promotion, and Checkmate are incorporated in the algorithm. The software was initially made as independent project on computer as a Console Application on C++, later it was integrated with mechanical setup using Arduino microcontroller. The Console Application also served as major savior in debugging the flaws in mechanical integration of software through its visual feedback on screen.

The source file contains a “main.cpp” * and “pieces.h” ** header file.

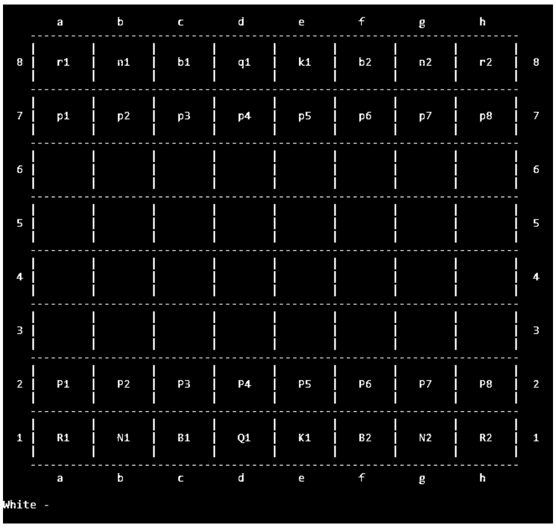

Figure 18: Chess game on console application in C++

Header file contains a Base Class defining the common attributes of each Chess Piece and classes are derived from it for every kind of pieces containing their specific movement rules. Every kind of piece has a white set and a black counterpart each piece is associated with its value of strength and in memory a piece is represented by its positive value if it is a white piece and a negative value for a black piece.

Functions of Piece Capturing, Castling, En Passant and Checkmate are defined in main.cpp file.

D. Update Chess Move to Remote Player

Chess move made by a player on its chess board by hand picking has to be reflected on the opponent's chess board. For this, information regarding the piece moved, its initial position and final position sensed from Hall Sensors are sent over internet. For establishing the internet connectivity ESP8266 Module is used with Arduino. There is a pre-defined library "ESP8266Wifi.h" for Arduino which provides an easy to use interface to transmit and receive data over Wi-Fi. Once the data is received on the remote player the motors are actuated to move the demagnetized electromagnet from its current position to the initial position of the chess piece across the shortest path computed. Then the electromagnet is magnetized to attract the above lying chess piece and is dragged to final position through a path it does not collide with other chess pieces and takes the least time. Once it reaches the final position the electromagnet is demagnetized again.

E. Problems Encountered

Arduino Mega Rev3 offers the maximum memory (256 Kb) in Arduino product range which was not sufficient for our purpose. Other standard alternative development board for the project is Raspberry Pi. Although it has 1 GB of RAM but much lesser number of GPIO pins compared to the former consideration and in addition to that higher complexity in usage. The solution was to use Laptop as the processor and Arduino as the controller for the hardware. The C++ Console Game validates the moves being played on the board and it communicates with Arduino via Serial Communication bridged by "SerialClass.h".

V. FUTURE PROSPECTS

Although the size of the current chess platform has been ensured to minimal dimensions given the cost and hardware constraints but a smaller size of the chess board can be attained by using sophisticated hardware and laying out the circuit on a printed circuit board (PCB). Emergence of sophisticated Artificially Intelligent Systems has revolutionized the gaming experience. Hence incorporating an artificially intelligent computer side to the Chess robot will help players train against a computer on the physical board eliminating the need for a human companion for practice. Furthermore, adding a computer generated recommendation system on the chess robot for the best upcoming move by the human player will let them improve at chess and this requires a development of Chess Engine. Using refined Natural Language Processing frameworks, voice controlled chess board can be built which will make this chess playing experience all the more exciting and will also empower the handicaps to play chess on this platform.

REFERENCES

[1] Javed, F. and Javed, S., "Design Methodology and Development of an Economical 3D Printer," SAE Technical Paper 2016-01-0325, 2016, doi:10.4271/2016-01-0325
[2] https://www.allaboutcircuits.com
[3] http://howtomechatronics.com/tutorials/arduino/how-to-control-stepper-motor-with-a4988-driver-and-arduino
[4] http://www.ti.com/lit/ds/symlink/drv8825.pdf
[5] Varun Gupta, Amit Kumar, Shilpi Agrawal, Saumya Jaiswal, "Autonomous Chess Playing Robot", International Journal of Engineering Research and Technology, ISSN: 2278-0181 Vol. 4, Issue 3, March 2015

AUTHORS

(1) Divye Bhutani Software Engineering undergraduate student at Delhi Technological University
(2) Yusuf Ali Mechanical Engineering undergraduate student at Delhi Technological University
(3) Pratyush Gupta Electronics and Communication Engineering undergraduate student at Delhi Technological University

*www.github.com/matrixgenez/chess_robot/main.cpp
**www.guthub.com/matrixgenez/chess_robot/pieces.h