**Project Final Report**

**Autonomous Chess Bot**

|  |  |
| --- | --- |
| **Project Name** | Autonomous Chess Bot |
| **Description** | Development of an autonomous chess playing bot with full features of a real chess game with add on features of virtual chess game giving the player a completely different chess experience. |
| **Project Sponsor** | Prof. Ayas Kanta Swain,  Professor at Electrical Department,  Faculty Advisor of Cyborg,  NIT Rourkela. |
| **Author(s)** | 1. Prabin Kumar Rath, B.tech 3rd year, Computer Science Department, NITR  2. Subham Sahoo, B.tech 3rd year, Computer Science Department, NITR  3. Pabitra Dash, B.tech 3rd year, Computer Science Department, NITR  4. Swostik Acharya, B.tech 3rd year, Computer Science Department, NITR |
| **Date** | June 27, 2018 |

We are developing an autonomous chess playing robot which has the capability to play chess with a human player. Currently the project is under Cyborg, the robotics club of NIT Rourkela.

Top of Form

Bottom of Form

Top of Form

Bottom of Form

***INTRODUCTION***

The aim of this project is to develop an autonomous chess playing bot with full features of a real game including add on features of virtual chess game giving the player a real chess experience. The project has been made to give the user a premium feeling of playing chess where the dots move autonomously with many abstracted tasks going on in the backend. The biggest challenge was to create a fully functional and working interface between the physical world and the available chess engines and make its mechanical unit work perfectly.

***PURPOSE OF THE PROJECT***

Autonomous Chess Bot has been constructed on a purpose to provide the player an realistic feeling of playing chess and experience. The other objectives behind the development of the project are listed below.

1. As the project involves a chess engine playing against users through a chess board, that improves the reality of the game. We are connecting the traditional chess engines (that were usually integrated with an UI) to life a chess bot to make the game more interactive and real.
2. This project can transform the way chess matches are conducted. With a similar industrial product, professional chess matches can be played remotely from two different places. If a player is somehow not able to attend a professional competition, this robot will preserve the game's authentic sense for the player who is at the competition by giving him a feel real feel of chess than a virtual one.
3. Chess being one of the most popular strategy games on this earth requires tremendous practice and training. Chess training centers around the globe can use these robots for giving a real experience to the children who are interested in the game. The best part is children don't need to stare at the monitor for hours thinking about their moves during their training sessions. Such robots will also attract young children to play chess as the robot's autonomous movements are impressive from a child's perspective.
4. Apart from that the robot is made with a cost-effective approach. If produced on an industrial scale can easily be afforded by parents who would like to train their children intelligent with the help of chess but with a more realistic manner.

***MECHANICAL DESCRIPTION***

A XY slider powered with Stepper motors has been used for on board piece movements. At first, a baseboard was made of plywood keeping the dimension 25”x25”x2.05” on which the whole mechanical system was mounted. Two parallel Y railings were placed above the base which were of length 24.63” and width 0.88” on an aim for the safe movement of stepper motors. A rack of length 24” was placed near one of these railings and was fixed with the base board.

Above it, stage made of plywood of 24” and width 5.91” was settled in such a manner that it was exact perpendicular to the Y railings. Y stepper motor was mounted on the railings through a pinion of radius 1 cm by cutting the wood stage accurately. Above that wood stage, another Railing in X direction was fixed. Rack of length 24” was placed near the X railing and was fixed with the stage with an aim for the accurate movement of X stepper motor. Another plywood stage of dimension 9.84”x3.94”x0.38’’ was created for the mounting of X stepper motor.

The stepper motor was mounted on the slider via a mounting plate such that the gear of stepper motor mate with linear gear. X stepper motor was used for any movement of chess piece over X direction and similarly Y stepper motor was used for any movement of chess pieces over Y direction whereas the movement steps were determined by our code and finally fed to the motors.

Another wood level was constructed over the slider for placing the electromagnet. The dimension of this wood level was 2” height, 0.75”x0.75” area of cross section. Over this wood level, the electromagnet was mounted. Thus, the motion of the electromagnet was guided by the motion of the stepper motor. To avoid any positional error, we used two IR feedback system each of which consisted of a circuit containing photodiode, IR led wrapped with black tape and high pass filter placed with a support of small wood piece. This feedback system played a vital role in positioning the magnet in accurate position of the chess board. The black white strip was used such that the IR system can work with minimum error. We used two IR feedback circuits on a purpose to robust the motion of electromagnet in both perpendicular direction.

***SOFTWARE DESCRIPTION***

Software part of the project primarily uses well known applications and frameworks like Arduino, ROS ,Qt and OpenCV. Basically, Arduino is an open source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board often referred to as a microcontroller and a piece of software or integrated development environment that runs on computer, used to write and upload computer code to the physical board. Arduino does not need a separate piece of hardware in order to load a new code on to the board it simply needs a USB cable. ROS stands for Robot Operating System which is a flexible framework for writing robot software. ROS is a collection of tools, libraries and conventions that aim to simplify the task of creating complex and robust robot behaviour across a wide variety of robotics platforms. ROS is multilingual i.e. it supports development in multiple languages. ROS helps in establishment of communication between two or multiple program files each of which is written in same or different programming language. The sensor or actuator message from one program node is transferred to the other by a default or customized message file through ROS. Thus, ROS plays essential role in the intercommunication of different program nodes. The nodes we used basically in our projects here are namely chess\_ai(Python), board\_handler(Python), frame\_pub(CPP), frame\_handler(CPP), UI stack(CPP), Ardsimul(Python), Arduino code(Embedded C).

***ELECTRONICS DESCRIPTION***

We supplied 12 volts from a SMPS source which was connected in parallel to buck converter, LM 7505, Arduino UNO and the electromagnet. Buck converter provides a voltage output which varies from 0 to 12v with a high current. In our case we used it to get a 5v high current output. This 5v output was connected in parallel with two ULN 2003A, with 0.1uF capacitors connected across each of them. The output pins of these ULN2003A were input to the stepper motor along with 5volt supply from buck converter. Stepper motor requires a high current to work which was a reason to use Darlington transistor ICs(ULN 2003A) here.

The output of LM7805, a 5v voltage regulator was provided as power input to LM324, HC05 and IR Pair. 1uF capacitor and 10k ohm resistor were used for the high pass filter for allowing only the low frequency signals from the photodiode and switches to be received by Arduino.

***Project Performance***

Top of Form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***On Schedule*** | | | | | |
| **Planned Finish Date** | **Actual Finish Date** | **Variance**  **(in days)** | **On Schedule** | **Ahead of Schedule** | **Behind Schedule** |
| 01-09-2018 | under development |  | **yes** |  |  |
| The team has been working on this project for over 6 months with dedication. This project is still under development and we are working on it for making the bot more robust and user friendly. We are expecting to finish it within 3 months with full functionality. | | | | | |
| ***On Budget*** | | | | | |
| **Approved Budget** | **Spent Budget** | **Variance** | **On Budget** | **Under Budget** | **Over Budget** |
| 10,000/- Rs. | 8,000/- Rs. | 2,000/- Rs. |  | **yes** |  |

The total amount of money spent on building the autonomous bot till date is 8000 INR. We had requested for 10,000 amount as per the requirement of our project out of which 5000 has been reimbursed and rest is yet to be released by next academic session.

***CONCLUSION***

All in all, the project has been made to give the user a premium feeling of playing Chess. We are working on the project keeping in mind the new changes that can be appended in it to make it’s working more robust. Further features will be added to it to make it more user-friendly. It was really a great project to work on and the dedication and perseverance shown by the team members really paid off making the project a success.

[***https://drive.google.com/open?id=1dATDm1BuBCA2hZYPpm-zD\_PY2i7zmTAA***](https://drive.google.com/open?id=1dATDm1BuBCA2hZYPpm-zD_PY2i7zmTAA)

Link to complete Documentation of the project along with pictures and detailed implematation description.

***REFERENCES***

1. <https://python-chess.readthedocs.io/en/latest/> Python-chess is an open source python module used for handling chess engines.
2. <https://stockfishchess.org/> An open source chess engine we used as default engine in our project.
3. <http://wiki.ros.org/> ROS official Tutorials.
4. https://www.dribin.org/dave/keyboard/one\_html/ Similar projects.
5. <http://www.instructables.com/> DIY community for electronics projects.
6. <https://playground.arduino.cc/> Arduino Official forums and tutorials.
7. <https://www.qt.io/> Official Documentation for using Qt with C++.
8. <https://opencv.org/> Open source Computer Vision Library for image processing.