A Rubik's Cube Solving Robot Using Basic Lego Mindstorms NXT kit

Khushboo Tomar
Department of Electronics and Communication Engineering,
Amity University, Sector-125, Noida 201313 (U.P.) India
tomar2khushboo@gmail.com

Abstract -- **The Rubik's Cube usually frustrates us when we are trying to solve it, but not able to do so. Now this frustration comes to an end by designing a Robot which can solve the Rubik's cube, so that it can resolve the colorful mystery. This is possible by studying number of algorithms like Corner first method, Herbert Kociemba two-phase algorithm etc. The hardware is made by using Lego Mindstorms NXT basic kit, which makes the cube to be solved using sensors and motors, whereas the software which makes the hardware to solve the cube is based on NXC (Not exactly C) using Bricx Command Center. Lego Mindstorms NXT Intelligent Brick helps to get the desired work done by the robot.**

Keywords: Lego Mindstorms NXT, NXC, BricxCC, Corner-first method.

I. INTRODUCTION

THE Lego Mindstorms consist of a series or number of kits which contain software as well as hardware to make various robots that are programmed and customized. Every kit has its brain in a brick, which is inbuilt in it and is actually an intelligent computer that helps to control the whole system. The whole system includes number of sensors and motors as peripherals and various Lego parts to make any mechanical structure. LEGO Mindstorms can be used for building embedded system with the help of electromechanical parts that are controlled by computer. In real life, we are using a number of embedded systems that are made by using Mindstorms. For example Elevator controllers, metro luggage checking systems, industrial robots etc.

The oldest (original) Lego kit (*i.e.* Mindstorms Robotics Invention System kit) had two touch sensors, two motors and one light sensor whereas the NXT version has one light sensor, one sound sensor, one touch sensor and three servo motors. The NXT 2.0 has two touch sensors, one light sensor and one distance sensor.

Lego NXT kit was given by Lego in July 2006. It is a programmable robotics kit and comes in place of old LEGO Mindstorms kit i.e. Robotics Invention System. There are two basic version kits: (a) the Retail Version and (b) the Basic Education Set. It is embedded with programming software which is NXT-G or it may be LabVIEW for Lego Mindstorms. There are some more languages like NXJ, leJOS, RobotC, NXC and NBC.

The brain of this kit is a computer having shape of a brick known as the NXT Intelligent Brick AKA (Ciara) shown in figure 1. It has four input ports for sensors and three output ports for motors, through the modification of RJ12 cables, which are like RJ11 phone cords but not so compatible. The brick also consists of a LCD display which is monochrome having 100×60 pixels and also it has four navigation buttons to make user go through the hierarchical menu. It has a microcontroller which is of 32-bit and called as ARM7TDMI core Atmel AT91SAM7S256 having flash memory and RAM of 256KB and 64KB respectively, with addition of another microcontroller Atmel AVR ATmega48 of 8-bit. It also supports some attractive features like Bluetooth, speaker, a rechargeable battery of Li-Ion with charger and can also play sound files. These sound files are sampled at 8 kHz.

Although this kit gives us four sensors i.e. Light sensor, Ultrasonic sensor, Touch sensor and Sound sensor for four input ports, but only three sensors (viz. Light, Ultrasonic and Touch sensor) have been used to make this robot. For the output ports, all the three servo motors are used to serve the purpose.

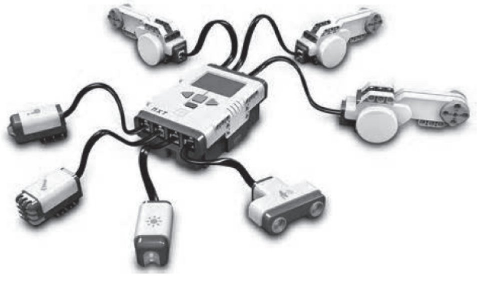
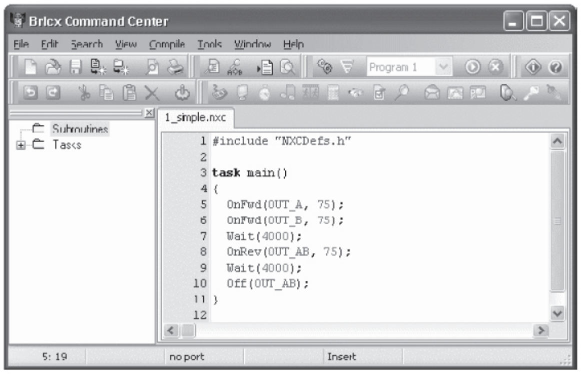


Figure 1. Lego Mindstorms NXT basic kit.

BricxCC stands for Bricx Command Center which provides an environment for writing, compiling, downloading, editing and running the NXC and Next Byte Codes (NBC) programs for Lego NXT. Thus, it is an IDE (Integrated Development

Environment) and makes NXT able to work on different firmware. BricxCC has number of features like NEXT-Screen which is used to view the contents on LCD display such as images, videos of NXT and NEXT-Explorer which is used to do different work on file and its contents such as upload and download files, hex viewer of files, defragment of NXT.

NXC (Not eXactly C) is a simple high level and open-source language like C. This language is based on Next Byte Codes (NBC) language and thus built on its compiler. To program Lego NXT kit, NXC is used and this is similar to NQC for Lego brick. Video making, games making can also possible with NXC and is a mostly used third Party language for programming for Lego NXT. To make using NXC more easier, Bricx Command Center (BricxCC) is used which provides an IDE for writing programs, downloading programs to robot, starting and stopping programs, browsing flash memory of NXT, converting sound files etc and thus works equivalent to a text processor with some extra features. Figure 2 shows a simple program which is written in NXC language but compiled on BricxCC. Once the robot gets connected to the computer, then the BricxCC will show four buttons to compile, download, run and stop the program in the robot, as shown in Figure 3.



```
#include "NXCDefs.h"

task main()
{
  OnFwd(OUT_A, 75);
  OnFwd(OUT_B, 75);
  Wait(4000);
  OnRev(OUT_AB, 75);
  Wait(4000);
  Off(OUT_AB);
}

```

Figure 2. Simple program of NXC on BricxCC.

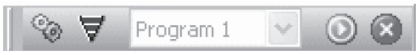


Figure 3. BricxCC main buttons to Start and stop program.

II. METHODOLOGY

While making any Robot, Hardware and Software are equally important for its success. So starting with the hardware of the Rubik's Cube solving Robot, it requires Sensors and Motors given in Lego Mindstorms NXT kit and the connections made are:

- Touch sensor: connected to Port 1
- Light sensor: connected to Port 2
- Ultrasonic sensor: connected to Port 3
- Turntable Motor: connected to Port A (output port)
- Tilter Arm Motor: connected to Port B
- Light Sensor Motor: connected to Port C

Whereas the things required for making the basic building for Rubik's Cube solving Robot, are given in the Lego Mindstorms NXT kit and the practical structure formed is shown in figure 4.

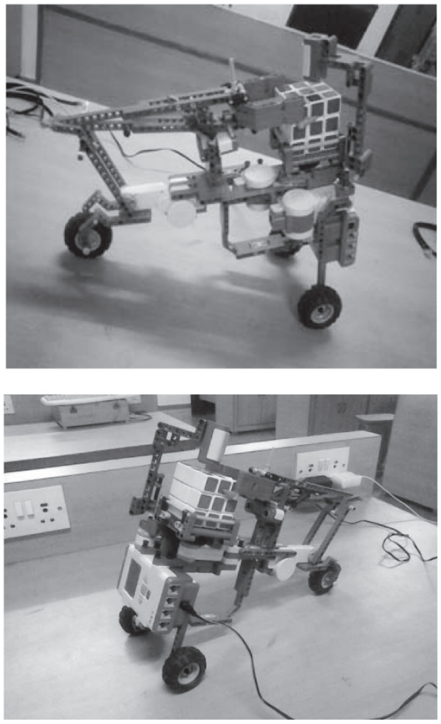


Figure 4. Actual Hardware.

The hardware shown by figure 4 requires the software part which is based on the algorithm of Corner- first method and its program will be written in NXC on BricxCC. This algorithm helps to solve the Cube using the methodology i.e. the solution of this method rotates the cube before posturing it. In this method, the solution first starts solving the Corners of the cube by orienting top and bottom corners. Then the solution will start solving the Edges of the cube by number of orientations. The ideas and sequences it is using, are taken from the solutions of some other algorithms/ methods. The steps of solving the cube by this method are as follows:

Step 1:Solving Corners: a) Top Corners Orientation

Figure 5. Start to solve corners.

In this step, top face corners of the cube are oriented in maximum 6 moves in figure 5 and then the same orientation is done for managing bottom face corners, in figure 6. In this step, the average no. of turns requires are 5.

b) Bottom Corners Orientation

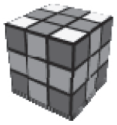


Figure 6. bottom corners.

Rotate the cube by 180º such that top face will become the bottom face and vice-versa. The corners should be oriented by matching any of the seven patterns shown below and then rotate according to its solution given with it.

- T-Pattern with Solution: RUR'U'F'U'F

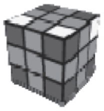


Figure 7. pattern T.

- L –Pattern with solution: FR'F'U'R'UR

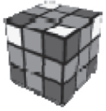


Figure 8. pattern L.

- Pattern: μ with solution: RUR'URU2R'

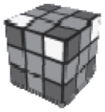


Figure 9. pattern μ.

- Pattern: Inversed μ (mu) with solution: R'U'RU'R'U2R

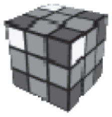


Figure 10. Pattern Inverse μ.

- Pattern: π (pi) with solution: RUR2F'R2UR'

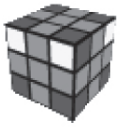


Figure 11. π (pi) pattern.

- Pattern: U with solution: R' F' U' F U R

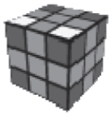


Figure 12. Pattern U.

- Pattern: H with solution: R2 U2 R' U2 R2

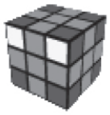


Figure 13. Pattern H.

For this step, the average no. of turns required is 7

c) Positioning of all Corners

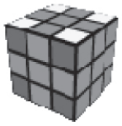


Figure 14. Solved corners.

Two adjacent corners (either on top or on bottom layer) are called as a Pair. If the position of two corners on a layer is relatively correct and their colors are same, then this means that such pair is correctly solved, as shown in figure 14. There can be only 0, 1 or 4 correct pairs on a layer.

The correctness of the pair can be easily identified by watching the 2 adjacent faces of any side. For any particular layer, if one pair is correct whereas other is incorrect, then it means that layer only has one correct pair. If two pairs are correct, then it means all corners on that layer are correct. If no pair is correct and also do not have the same color, then it means that layer don't have any correct pair.

Step 2: To Solve Edges

Now at this step, the corners are needed to be aligned whereas the centers are needed to be positioned. At this point, the cube should be completely symmetric except the edges. Take a new top and a new bottom face for further solving the cube and easily also.

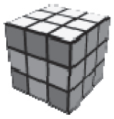
i) Solving Three (3) Top Edges

In order to solve the edges efficiently, the first top edge needs to be solved by making 1 or 2 turns without considering centers. For this step, the average no. of turns is nine (9).

ii) Solving Three (3) Bottom Edges

In order to reduce the required no. of turns, there is a need of combining this with the next following step while third bottom edge is being solved. For this step, the average no. of turns is twelve (12).

iii) Solving 1 more either Top or Bottom Edge



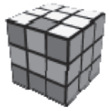


Figure 15. Bottom or top edge.

In this step, the last edge (top or bottom) may be in the middle layer, but not in orientation or solving. Continue the solving by matching the following and then go through the solutions given with them for solving the last edge during the orientation of middle layer edges.

Step 3: To position the Midges

- Pattern 1: Solution: R2 E' R2: Procedure: by sending front--right cubie to back--left cubie, back--left cubie to back--right cubie and back--right cubie to front--right cubie.

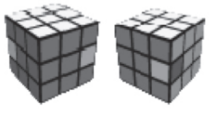


Figure 16. Pattern 1 midges.

- Pattern 2: Solution: M2 E' M2

Type: By exchanging centers with opposite centers.

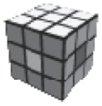


Figure 17. Pattern 2 midges.

- Pattern 3: Solution: R2E2R2

By exchanging, front--right cubie with back--right cubie and front--left cubie with back--left cubie.

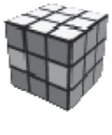


Figure 18. Pattern 3 midges.

For this step, the average no. of turns is four (4) and for this method, the average no. of turns is fifty-eight (58).

III. WORKING

By combining both hardware and software i.e. by burning the program into the Lego Mindstorms NXT brick, it will work as:

- Initially the NXT asks for giving the Cube which is displayed as "GIVE ME A CUBE" and receiving the cube, it will then display "Thank You".
- Now the sensor starts scanning the faces of the Cube.
- After scanning the faces, it starts resolving the colors and the NXT display says "Resolving colors".
- At this stage, Tilter Arm and Turntable helps to rotate the cube to sense the color on each face.
- Then the brick tries to calculate three solutions according to the programming.
- While calculating the solution, the brick makes a ticking like sound having pitch increasing with time, for 10 seconds or more.
- Then it selects the shortest solution out of the three solutions.
- Then it starts executing the selected solution and also displays the no. of twists left to solve the Cube.
- One of the three things can happen, if the brick unable in reading the colors, one of the following things will happen:

(A) While calculating a solution, it sounds continuously with increasing pitch with time.
(B) It will stop doing anything i.e. no sound nothing.
(C) It tries to calculate the solutions, but can't solve the Cube correctly as it was failed in resolving the colors.

IV. CONCLUSION

The work on Rubik's Cube solving Robot using basic Lego Mindstorms NXT kit is somewhat successful in which an ultrasonic sensor is used to detect the presence of the Cube and after getting the Cube, a Light sensor starts reading the faces of cube and its colors. With the help of Tilter Arm of the robot which tilts the Cube and Turn-table in which the Cube is fitted and thus helps to turn the Cube, light sensor reads all the faces of Cube. After scanning and reading of faces, three solutions have been calculated to solve the Cube and out of them, one shortest solution is selected to solve the cube. Finally, the robot starts following the shortest solution and solve the Cube by twisting, turning and tilting it using the Arm and the table. But some difficulties have been faced during solving the cube like light sensor caused difficulty in distinguishing between colors having close intensities (e.g.: red & orange, white & yellow, green & blue), which can be removed by either using vibrant color stickers or using latest Lego kit which has Color sensor that can easily differentiate colors (like red and orange). Also, some points need to be kept in mind while working on this project is:

- The initial position of the light sensor is very important, including the distance between the sensor and the surface of the cube, which is also crucial.

- If the sensor is too close to the surface of the cube, it may not give inaccurate readings. Also we can adjust the height of the light sensor and the distance should be 3 mm (approx) between the sensor and the top surface of cube.
- The gap of the tilter arm should not be very tight for the cube, so that arm will drag it back.
- We need to lubricate the cube with silicon spray, if the cube is very stiff to move.
- The turntable should be aligned at the initial position before getting the program run.
- The tilter arm should be completely at its initial position i.e. fully retracted before getting the program run.
- There must not be a strong (bright) light-source above/ on the Robot.

V. ACKNOWLEDGEMENT

The Author is grateful to Amity School of Engineering and Technology for providing facilities.

VI. REFERENCES

- **Research Paper**

[1] Hassan Samadi and Mohd. Razi Daliri, "Solve Rubik's Cube with Robot using Computer Interface", IEEE Transaction and ICIS, 2014.

- **Books**

[2] W.Fotheringham, "Fotheringham's Sporting Past-times", Anova..Books. p.p 50; ISBN 1--86105--953--1, 2007.

[3] Michael William Dempsey, "Growing up with science: The illustrated encyclopedia of invention", London; (Marshall Cavendish); p.p. 1245; ISBN 0--87475—841--6, 1988.

[4] Daniele Benedettelli, "Programming LEGO NXT Robots using NXC", Version 2.1, 9 April 2007

- **Technical Report**

[5] Zongheng Wang, Ifeoma Okereke, Yin-Chieh Wang and Heather Fisher, "Rubik's Cube Solver", Embedded system Design, CSEE W4840, 16 May 2013.

Khushboo Tomar obtained B. Tech in Electrical and Electronics Engineering from Vishveshwariya Institute of Engineering and Technology (V.I.E.T.), Gr. Noida (U.P.) in 2012. Currently, pursuing M.Tech (Embedded Systems & Technology) in the Department of Electronics and Communication Engineering at Amity University (AUUP), Noida. She has given Term paper on "High Level Synthesis" as a part of course curriculum. Her areas of interest are Robotics, Embedded System, SoC and ASIC (FPGA). After completing M.Tech, she plans to pursue career in Education sector.