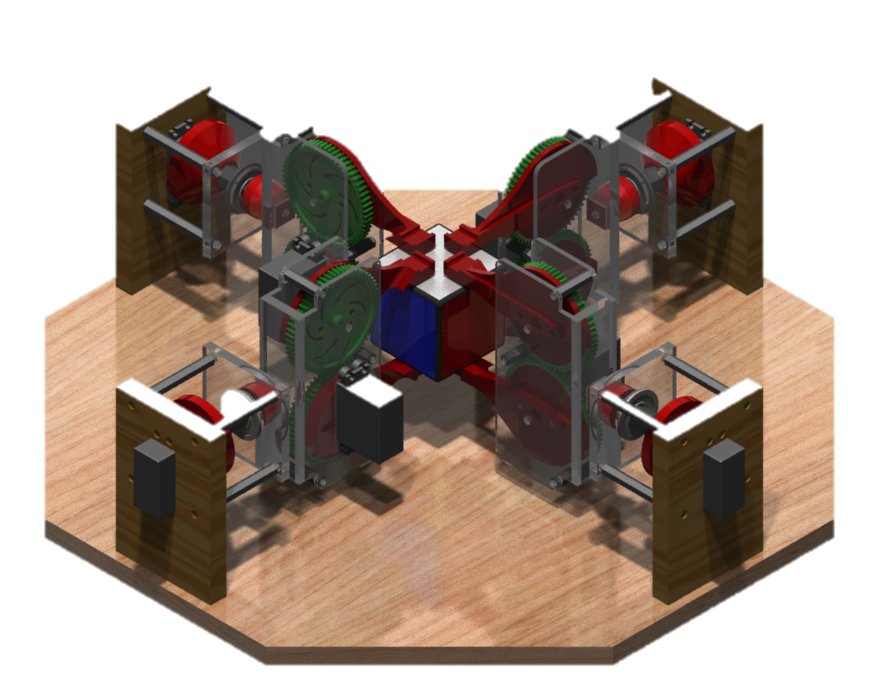
Mr. Campbell

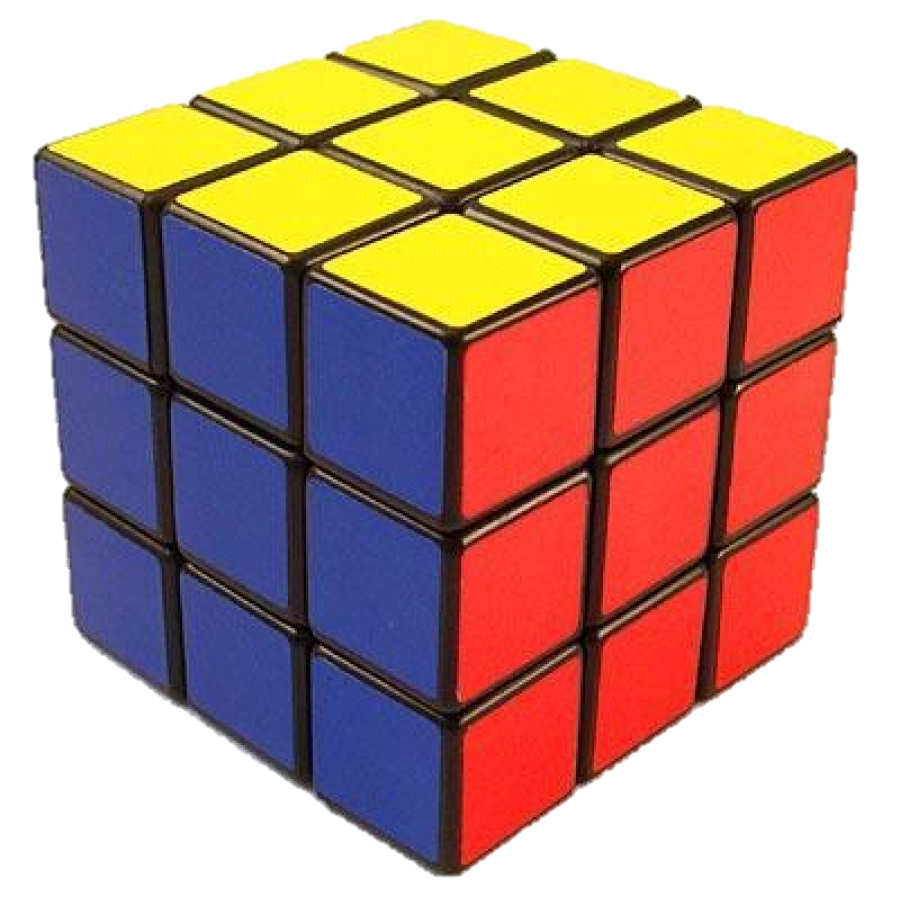
TDJ4M1-01  
1/1/2013

|  |
| --- |
|  |



|  |  |
| --- | --- |
| Akash Ryan Kavin Jigar Jasper | Rubix Cube Solver |

**Project Challenge**

The goal of our project is to solve a regular 3x3x3 rubik's cube without any human interaction. This will be achieved by manufacturing a robot which will automatically solve the rubix cube. The robot will be able to solve the rubik’s cube in any solvable state, a solved rubik's cube is when all the colours of all the sides are the same. The final goal for our project is to solve the rubik's cube no matter how long it takes and without any human interactions.

**Background**

After knowing what our goal was we started by searching if a similar goal was accomplished by another group of people. We researched starting from YouTube and we noticed many other people have had already completed the goal we wished to accomplish.

Links to cube solvers   
<http://www.youtube.com/watch?v=_d0LfkIut2M>  
<http://www.youtube.com/watch?v=bNAnUygqOYc>

<http://www.youtube.com/watch?v=5fAn5A0HbhU>

<http://www.youtube.com/watch?v=qdzzcPT0EDI>

After watching what people had done to accomplish the goal, we understood the possible ways we can manipulate the cube. We decided the type of claw we would use depending on how consistent, accurate and efficient it would be. This also gave us an idea of what type of motor we would need which was depending on how much torque is required for our function to movie. Initially we wanted to use the vex motors for our function.

Vex Motor: <http://www.vexrobotics.com/vex/products/accessories/motion>

We wanted to use vex motors because we had them in excess in the shop and they had enough torque for our function, but then we were unable to use them due to other students needing them. So we went on to another type of motor called, parallax motor.

ParallaxMotor:<http://www.parallax.com/Store/Accessories/MotorServos/tabid/163/CategoryID/57/List/0/SortField/0/Level/a/ProductID/101/Default.aspx>

These motors were also in excess to us and had enough torque for our function, so we ended up using this instead. The function we plan on designing requires 8 motors, the motors are powered and programmed through a PWM cable, so we knew we need a micro controller which allows PWM motors to be used and has 8 free ports for motors.

Now that we had the mechanical part figured out, we researched about the programming part. First we needed a way to detect colours on the rubix cube. Many of the people had previously done it used the camera from a smart phone, since none of us had a smart phone, we decided on using a webcam. We needed to research on which type of code would allow us to use the webcam.

Sites used for webcam:

<http://webcamxtra.sourceforge.net/index.shtml>

<http://playground.arduino.cc/interfacing/processing>

<http://marvinproject.sourceforge.net/en/index.html>

<http://www.oracle.com/technetwork/java/javase/tech/index-jsp-140239.html>

<http://stackoverflow.com/questions/276292/capturing-image-from-webcam-in-java>

Using the sites we learned what we can use to track and detect the colour on the rubix cube with a webcam. The program to track the colour is written in java. Now all we were missing was a microcontroller, we were informed by our teacher that he can access Adriano microcontroller, so we decided on using them since they have PWM ports and it can be coded using java, meaning the webcam will be compatible with it.  
 In conclusion we researched three main things, how the rubix cube will be manipulated mechanically, how the colors will be recognized on the program, and what microcontroller will be used.

**Solutions**

|  |  |  |  |
| --- | --- | --- | --- |
| **CLAW** | | | |
| DESIGN | IDEA | PROS | CONS |
| 1 |  | -simple  -fast  -easy to build  -light  -robust  - angular motion | -have to be perfect with measurements  -have to have great precision in order to work |
| 2 |  | -fast  -accurate  -parallel motion | - hard to CAD  -not as robust  -heavy |
| 3 |  | -linear motion  -amazing accuracy and precision | -slow  -weak  -heavy |

|  |  |  |  |
| --- | --- | --- | --- |
| **CLAW LAYOUT** | | | |
| DESIGN | IDEA | PROS | CONS |
| 1 |  | -light  (2 claws)  -small | -slowest  -in accurate  -less precession |
| 2 |  | -fast speed  (4 claws) | -heavy  -big |
| 3 |  | -medium speed  (3 claws ) | -medium heavy  -medium big |

**Best Solutions**

**Mechanical Aspects**

After deciding on the possible designs we can manufacture, we concluded on 3 possible designs. Design ‘1’ is a simple rotating claw that is easy to manufacture and fairly accurate. Design ‘2’ is a complex design which is harder to manufacture but allows for a higher accuracy. Design ’3’ although being the most accurate and robust of all three designs it is the slowest and most time consuming design to manufacture.

We decided to continue with design ‘1’ because the pros of this design out weight the cons of all the other designs. Design 1 is simple to manufacture and is less likely to break. It allows us to manipulate the cube with maximum accuracy and minimal effort.

We also considered how many claws the contraption should have , minimum of 2 and maximum of 6. We briefly considered having 6 claws but after looking at our hardware limitations we realized this design was not possible and two claws although being light and material conserving it will require too long to solve the cube. So we decided that 4 claws was the optimal solution for the task. 4 claws allow us to manipulate the cube fairly efficiently with accuracy and minimal effort.

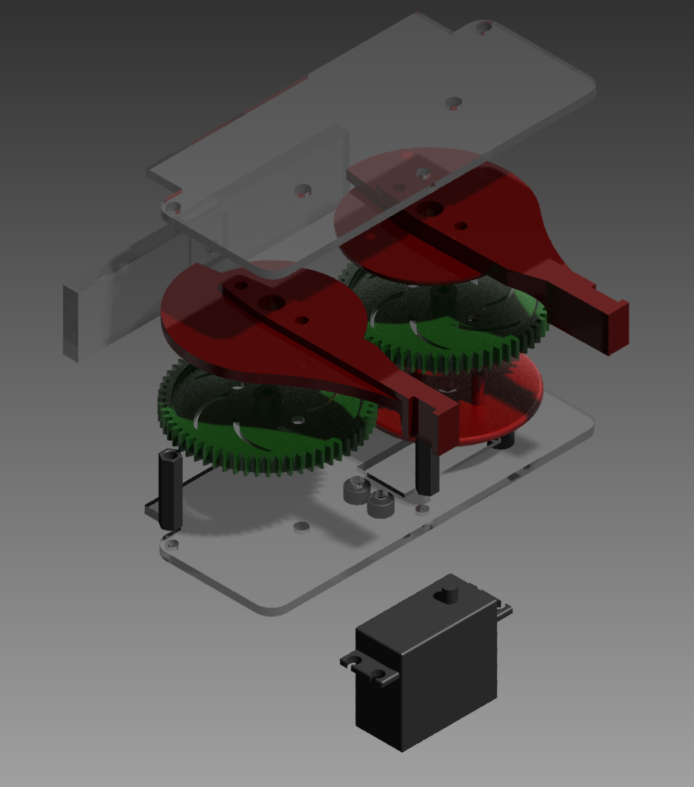
**Programming Aspects**

We first thought about designing our own algorithm to solve the rubix cube with minimal steps required, similar to some of the more advanced rubix cube solving algorithm. But then we realized that we are not mathematicians and within the time frame we have this is not possible. So we decided to approach the solution from a much simpler perspective. Solving the cube as if we were solving it by ourselves. Although this requires many more steps in solving the rubix cube, it is the best we can do. The process in which we use to solve the cube will be detailed in the flow chart.

**Plan It**

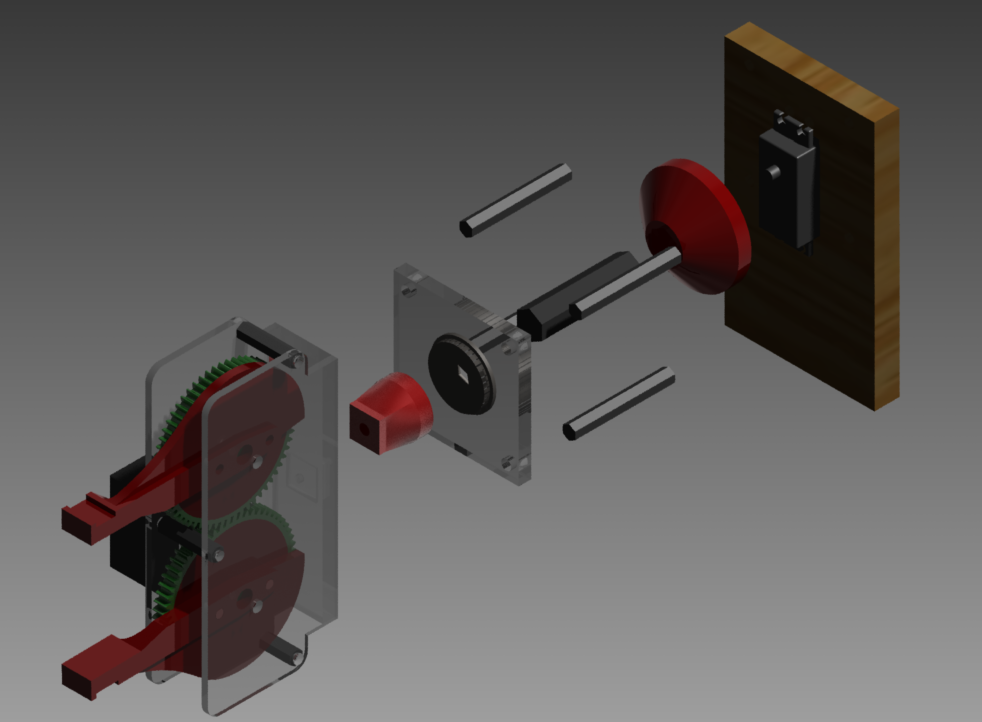
**Claw**

The final design has 4 claws which each are powered by one motor. The one motor is linked to two arms which are linked by gears. The ratio on the gears is 1:1 so the arms move opposite direction at the same speed. This allows us to grab the cube.



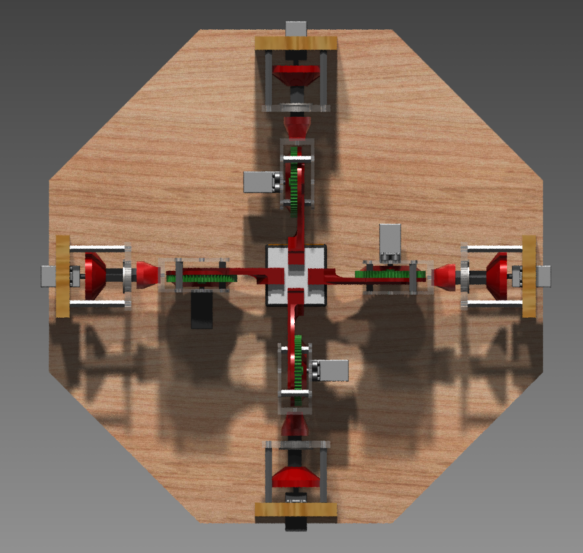
**Pivoting Claw**

The four individual claws are attached to fix hub which allows it to rotate along an axis. It has an extended support outside the main frame, supported by standoffs to make it robust. The extended support has a bearing for smooth rotations. This allows us to turn the grabbed cube.



**Final Product**

The four functions are all attached to one hexagon shape base made with wood. They are precisely placed to handle the cube accurately. The whole function is well braced to keep a high accuracy of rotations and grappling of the cube.



**Materials Required to build the product**

* 3D Printer
  + CADed files
* CNC Machine
  + Lexan pieces
  + Wood pieces
* 4 Bearings
* 4 Shafts
* 16 Stand off
* 12x15 inch area of Lexan (1/8)
* Wood
* 8 Vex gears
* 8 Motors
* 1 camera
* Adriano Microcontroller
* Rubix Cube
* Foam
* Double sided tape

**Flow Chart**

**Go to the directory of :**

:Rubicks\_Cube\_Solver\Report\Rubicks Cube Solver Flowchart\_Web\_PNG\index

The flow chart explains how the program will solve the rubix cube. To open it go to the directory given above open and click “Go inside level” for detail on the certain criteria.