Princess Sumaya University for Technology

King Abdullah II Faculty of Engineering

Computer Engineering Department

Microprocessors & Embedded Systems.

Project Documentation

Rubik's Cube Solver

Authors:

Abed Aljundi	20180297	Electronics Eng.	Abe20180297@std.psut.edu.jo
Omar Natsheh	20180088	Electronics Eng.	Oma20180088@std.psut.edu.jo
Zaid Hamarneh	20190095	Computer Eng.	Zai20190095@std.psut.edu.jo

Supervisors: Dr. Belal Sababha. & Eng. AbdulGhani AlJundi.

Due Date: January 24th, 2023

Abstract

The Purpose of the project is to be able to learn how to program and use the microcontroller in building a Rubik's Cube solving machine using many different parts and components that will give a result of high accuracy and speed when solving the cube.

Table of Contents

Introduction

Our Project is an automatic Rubik's cube solver, the solver is made out of six stepper motors one for each face of the cube that with the aid of their drivers are capable of moving the cube's faces in high accuracy and speed with the help of a cube solving application that requires the cube to be scanned using a camera, then the output text file Is sent to the PIC MCU serially to solve the cube. The Purpose of the Project Is to learn how we can combine Our knowledge from past Materials taken during our study in the university and to learn how to put all of it in use to build our project.

Theoretical & Background

There are 12 rotations that can be done to the cube, which are divided into clockwise and counterclockwise. The rotations are represented by Front(F), Back(B), Left(L), Right(R), Up(U), and finally Down(D). Moreover, the counterclockwise can be represented with Prime (‘) as shown in Figure 1.

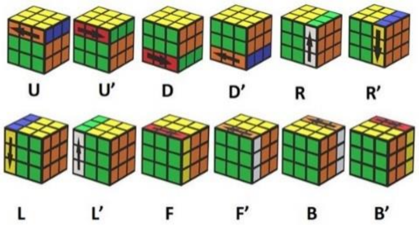

Figure 1: Rubik's Cube Possible Movement.

To begin with, a cube solving software called Cube Explorer v5.14 (developed by Herbert Kociemba). Using a mobile's phone camera, each face of the cube is scanned then the Cube Explorer applies the Two-Phase Algorithm that generates an optimum solution with a maximum of 26 moves.

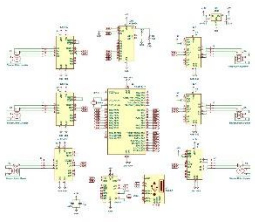

Figure 2: Schematic

After being generated, the commands then sent to the PIC16F877A MCU (using USART) which will decode the algorithm and move the stepper motors as instructed by the decoded message causing a series of movements leading to a solved cube. When done the time taken will be shown on the LCD screen.

Designs

In the first step of the design process, a schematical prototype of the project was made on KiCad to show all the components that were used as well as the connections between each component and the other with using their respectful ports. Once finished, the schematic was used to help guide us with the connections onto the breadboard that were needed to be made to thoroughly test the system to ensure that all the connections were correct before finally moving it onto the welded board.

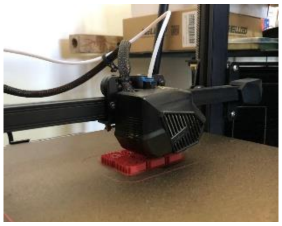

Figure 3: Printing Caps using 3D printer.

Furthermore, the stepper motors and drivers were thoroughly tested as well, which were six NEMA17 stepper motors with three A4988 and three DRV8825 motors were used. How the motor works is that once a step signal is transmitted to the drivers, the driver then causes a change in both phases of the motor that causes the gear teeth in the stepper motor to turn the gears one time (which is 1.8 degrees). To turn the motor rotor 90 degrees, 50 step signals should be sent in sequence of high and low signals(motor works on falling edge).

In addition to that, the code used mainly focuses on two things, using the USART module to translate the incoming commands into movements of the motors, as well as using Timer0 to accurately calculate the time taken to solve the Cube.

The USART takes each character and compares it to the cases specified to each motor while also checking if the prime character was sent after the character which if true causes the motors to spin counterclockwise instead of the defaulted clockwise rotation. While the TMR0 was used to cause it to overflow each 0.792ms for to main functions, first to calculate the time taken to solve the cube and more importantly, has been used as a delay in the move function between each step signal to generate the rising edge of the signal.

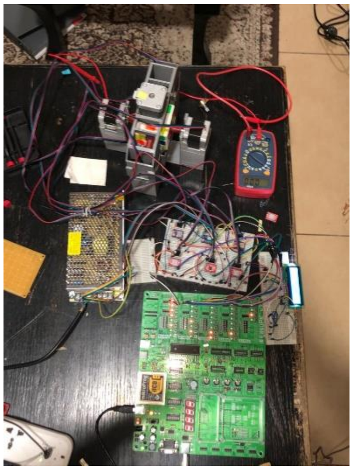

Figure 4: Testing components with EasyPic

Results

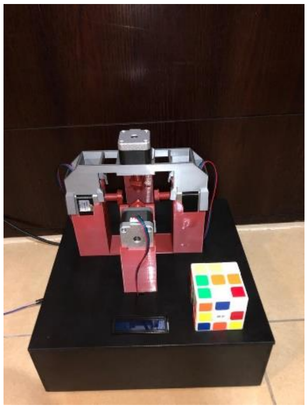

Figure 5: Qußot the cube solver

The Result of the cube being solved was very satisfying and faster than what was expected. It was able to move 24 times in less than 2 seconds! It was found that the average amount of moves needed to solve the cube were 22 moves which is the fastest in the middle east!

Problems

Furthermore, the A4988 drivers would get browned out faster than expected which caused a failure in one of the motors causing the cube to get stuck! A recommendation is trying to buy the DRV8825 drivers as they are able to endure more load on them. In addition, the noises resulting from the drivers were high so another recommendation is to buy the TMC2208 drivers as they are quieter than the other two.

Lastly, the Rubik's cube would sometimes become loose after many failures so the main recommendation is to use the magnetic cubes as they would snap into place when the rotation is made.

Conclusions

The project is a Rubik's cube solver that is able to solve the Rubik's cube in less than two seconds with the help of six stepper motors and their drivers which when solved, would print the time taken onto the LCD screen. Many problems occurred while working on the project from all parts (mechanical error, code error, electrical error) but after long hours and hard work, we were able to fix the errors and made the project successful.