MPMCI LAB MINI PROJECT on

Implementation of **DC Motor Interfacing using 8051 Microcontroller** and **L298N Motor Driver IC.**

A Mini Project Report

By

SVN. Ramakanth

1602-19-733-118

Y. Sathwik

1602-19-733-101

Under the guidance of

Smt. B. Syamala.

ASSISTANT PROFESSOR

Department of Computer Science and Engineering

Vasavi College of Engineering.

(Affliliated to Osmania University)

Ibrahimbagh, Hyderabad.

INDEX

S.No	CONTENTS	PAGE.No.
1	Acknowledgement	03
2	List of Figures	04
3	Project Abstract	05
4	Introduction to 8051 Microcontroller	06
5	Software Requirement	08
6	About L298N Motor Drive IC	09
7	Simulation using Proteus 8	10
8	Sample Code	12
9	Output Screenshots	15
10	Conclusion	18
11	References	18

ACKNOWLEDGEMENT

I would like to express our heartfelt gratitude to Smt. B. Syamala, our project guide, for her valuable guidance and constant support, along with her capable instruction and persistent encouragement.

I am grateful to our Head of Department, Dr. T. Adilakshmi, for her steady support and the provisionof every resource required for the completion of this project.

I would like to take this opportunity to thank our Principal, Dr. S. V. RAMANA, as well as the management of the institute, for having designed an excellent learning atmosphere

LIST OF FIGURES

Figure 1: Layout of 8051

Figure 2: L293D Pin Diagram

Figure 3: Figure-3: Circuit Diagram of 8051 Microcontroller interfaced with L293D and DC motors

Figure 4: Creation of HEX File

Figure 5: Clock wise and anti-clock wise motor Direction control

Figure 6: Clock wise motor Direction control

Figure 7: Anti-clock wise motor Direction control

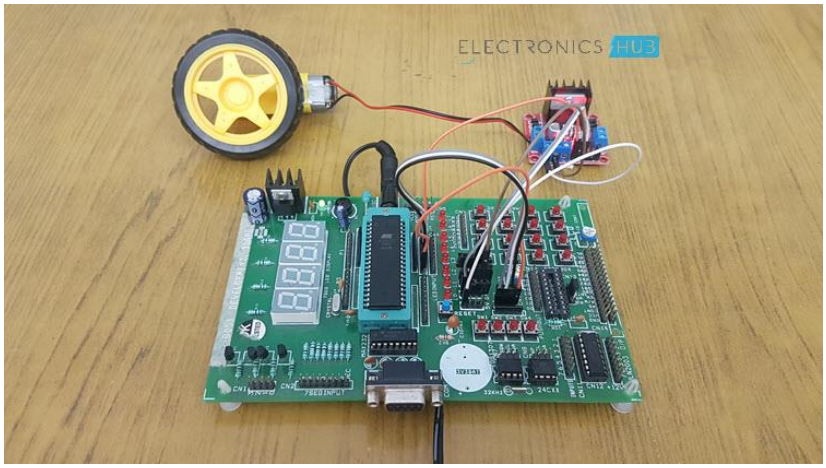
Project Abstract

In our technically evolving world robotics has a wide scope of development. Mobility of various parts of machinery is interlinked with motor interfacing. In this project, DC motor is interfaced using a microcontroller so as to manipulate the functional operations of motor such as direction and speed control. This Interfacing enables us to access complex components easily and make their usage even more easier.

A microcontroller's pin has maximum output current capacity of 15mA at 5V. Most of DC motors have power requirements that can't be matched with the microcontroller and the microcontroller may be damaged due to back EMF generated by the motor. Hence, it is not a good idea to interface DC motor directly to the controller. So, we use motor driver circuit in between a DC motor and the microcontroller. In this project we use L298N motor driver IC along with 8051-microcontroller in order to achieve motor interfacing. In case of L298N, the motor supply is up to 46V and it can provide a current of 3A which makes it suitable to run the motor.

Motor drivers have inbuilt H-bridges that reverse the direction of rotation without rewiring. Motor driver and the micro controller pins are interconnected and are programmed to achieve the interfacing. 2 buttons are used in the circuit, one for clockwise motor rotation and the other for counter-clockwise rotation.

This Demonstration has wide range of applications. Applications of this project are it can be used in robotics for Robot directions control, speed control of DC motor and in applications where we need to drive the high voltage motors.For this project we are going to use Keil MicroVision for software development environmentand Proteus for simulation.



Introduction to Microcontroller

8051 microcontroller is designed by Intel in 1981. It is an 8-bit microcontroller. It is built with 40pins DIP (dual inline package), 4kb of ROM storage and 128 bytes of RAM storage, 2 16-bit timers. It consists of are four parallel 8-bit ports, which are programmable as well as addressable as per the requirement. An on-chip crystal oscillator is integrated in the microcontroller having crystal frequency of 11.0592 MHz. In case the data is larger than 8 bits then it has to be broken into parts so that the CPU can process conveniently. Most manufacturers have put 4Kbytes of ROM even though the quantity of ROM can be exceeded up to 64 K bytes. The 8051 has been in use in a wide number of devices, mainly because it is easy to integrate into a project or build a device around.

Since the basic layout of a microcontroller includes a CPU, ROM, RAM, etc. the 8051 microcontroller also has a similar layout. The following image shows a brief layout of a typical 8051 Microcontroller. It is a CISC based Microcontroller with Harvard Architecture (separate program and data memory).

8051 Internal Architecture

- 8 – bit CPU with two Registers A (Accumulator) and B.
- Internal ROM of 8K Bytes – It is a flash memory that supports in – system programming.
- Internal RAM of 256 Bytes – The first 128 Bytes of the RAM i.e. 00H to 7FH is again divided in to 4 banks with 8 registers (R0 – R7) in each bank, 16 bit addressable registers and80 general purpose registers. The higher 128 Bytes of the RAM i.e. 80H to FFH consists ofSFRs or Special Function Registers. Using SFRs we can control different peripherals like Timers, Serial Port, all I/O Ports, etc.

- 32 I/O Pins (Input / Output Pins) – Arranged as 4 Ports: P0, P1, P2 and P3.
- 8- bit Stack Pointer (SP) and Processor Status Word (PSW).
- 16 – bit Program Counter (PC) and Data Pointer (DPTR).
- Two 16 – bit Timers / Counters – T0 and T1.
- Control Registers – SCON, PCON, TCON, TMOD, IP and IE.
- Serial Data Transmitter and Receiver for Full – Duplex Operation – SBUF.
- Interrupts: Two External and Three Internal.
- Oscillator and Clock Circuit.

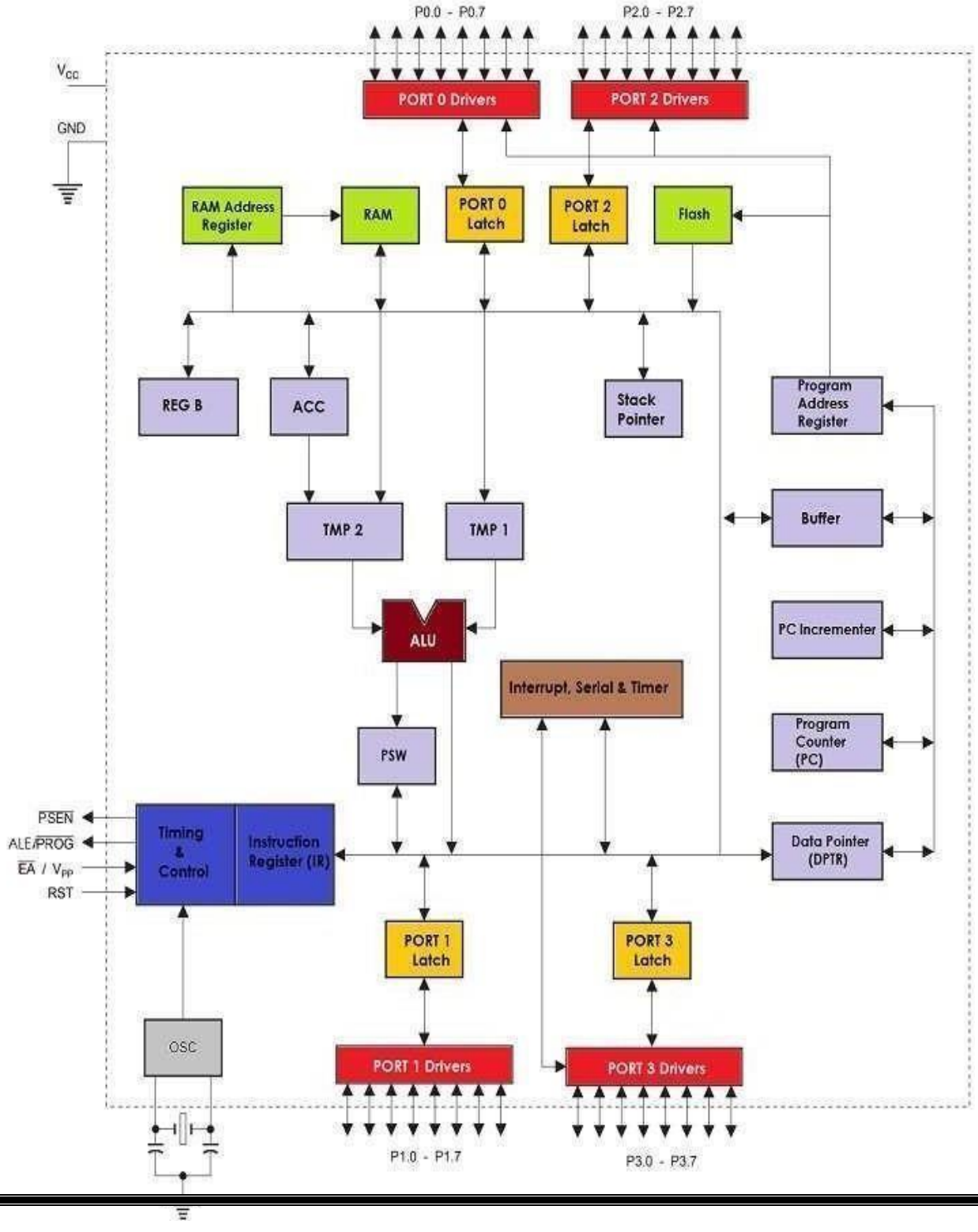


Figure 1: Layout of 8051 Microcontroller

SOFTWARE REQUIREMENT

Keil MicroVision5

For this project, we have used Keil MicroVision5 as the software development environment. Keil MDK is the complete software development environment for a wide range of Arm Cortex-M based microcontroller devices. MDK includes the µVision IDE and debugger, Arm C/C++ compiler, and essential middleware components.

The µVision IDE combines project management, run-time environment, build facilities, source code editing, and program debugging in a single powerful environment. µVision is easy-to-use and accelerates your embedded software development. µVision supports multiple screens and allows youto create individual window layouts anywhere on the visual surface.

The Debugger provides a single environment in which you may test, verify, and optimize your application code. The debugger includes traditional features like simple and complex breakpoints,watch windows, and execution control and provides full visibility to device peripherals.

Proteus 8

And for simulation, we have used Proteus 8 Professional (Schematic Capture). It allows us to develophardware implementation of the project by providing a wide range of components to build circuits inworkspace and thereby allows us to attach "hex" code to the simulation and run the simulation at real-time.

Proteus 8 Professional is a software which can be used to draw schematics, PCB layout, code andeven simulate the schematic. It is developed by Labcenter Electronic Ltd.

Features of Proteus:

Schematic Capture – One can easily draw the schematic using proteus. We can select thedevices by clicking ‘Pick Devices’ button and select the desired component.

Simulation – Many components like microcontrollers can be simulated in proteus.

Designing PCB

3D Visualization

About L298N Motor Driver IC

L298N IC Features:

- Operating voltage range: +5 to +46V.
- Maximum supply voltage:50V.
- Maximum Input and Enable Voltage: +7V.
- Maximum current allowed to draw through each output: 3A.
- TTL control inputs.
- Total power dissipation:25W.
- Operating temperature: -23°C to 130°C.
- Storage Temperature: -40°C to 150°C.

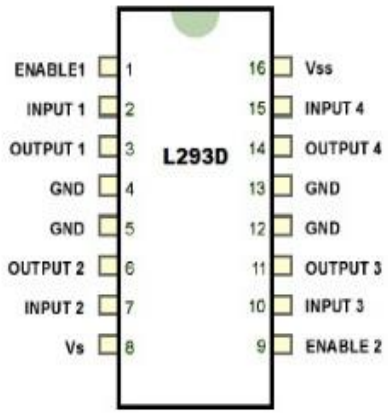


Figure-2: L293D Pin Diagram

SIMULATION USING PROTEUS 8

The microcontroller used for this simulation is AT89C51 (ATMEL) microcontroller

COMPONENTS:

1)AT89C52 Microcontroller - 8051 microcontroller.

2) L293D - Driver IC

3) Capacitors (2) – 33 pf

4) Resistors (2) – 300k Ohms

5) DC motors (2)

6) Switches (2)

7) Batteries (2) – 5V and 12V

NEED FOR USING DRIVER IC

It is not good to connect DC motor directly to the microcontroller. Since the maximum current that can be sink from 8051 microcontroller is 15 mA at 5v. But a DC Motor needs much more currents. It also needs more voltages as 6v, 12v, 24v etc. depending upon the type of motor used. Another issue is that the back EMF produced by the motor may affect the proper functioning of the microcontroller and reversing the direction can damage the controller. Due to these reasons we can't connect a DC Motor directly to a microcontroller. To overcome the problems in their interfacing, a motor Driver IC is connected between microcontroller and DC motor. Motor driver is a little current amplifier.

It takes a low current signal and gives out a high current signal which can drive a motor. It can also control the direction of motor. We can use any Driver IC like L293D.

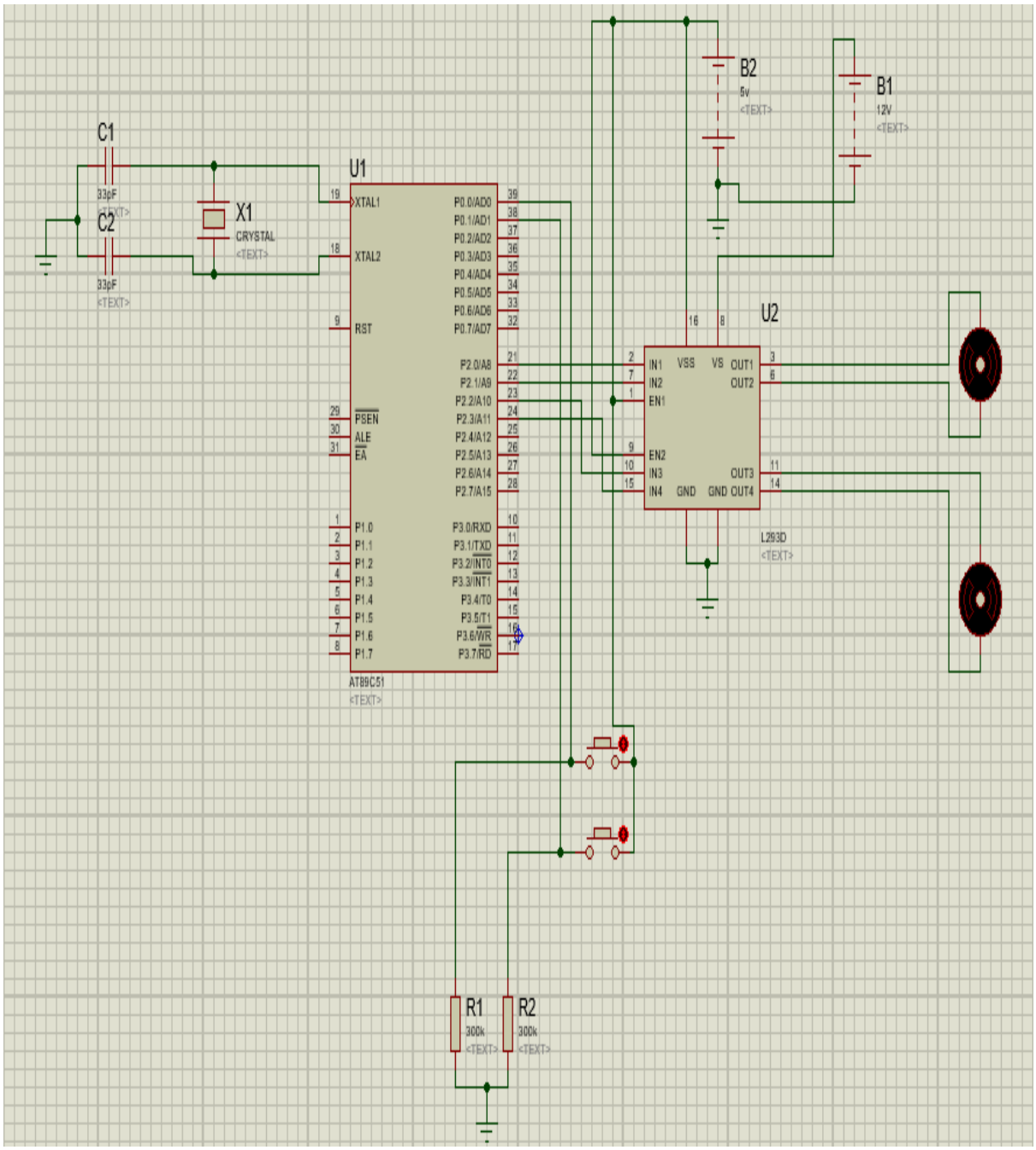


Figure-3: Circuit Diagram of 8051 Microcontroller interfaced with L293D and DC motors

SAMPLE CODE

```
#include <reg51.h>
void delay(void);
sbit m1_pin1 = P2^0;
sbit m1_pin2 = P2^1;
sbit m2_pin1 = P2^2;
sbit m2_pin2 = P2^3;
sbit B1 = P0^0;
sbit B2 = P0^1;
void delay()
{
  int i;
  for(i=0;i<1000;i++)
  {
  }
}
void main()
{
P2=0x00;   //output port
P0=0x03;   //input port
if (B1==1)
```

```
{
   m1_pin1 = 1; m1_pin2 = 0;  //Rotates Motor Clockwise
        delay();
}
else if (B1==0)
{
   m1_pin1 = 0; m1_pin2 = 0;  //Stops Motor
   delay();
}
if (B2==1)
{
   m2_pin1 = 0;m2_pin2 = 1;  //Rotates Motor Anticlockwise
        delay();
}
else if (B2==0)
{
   m2_pin1 = 0; m2_pin2 = 0;  //Stops Motor
   delay();
}
}
```

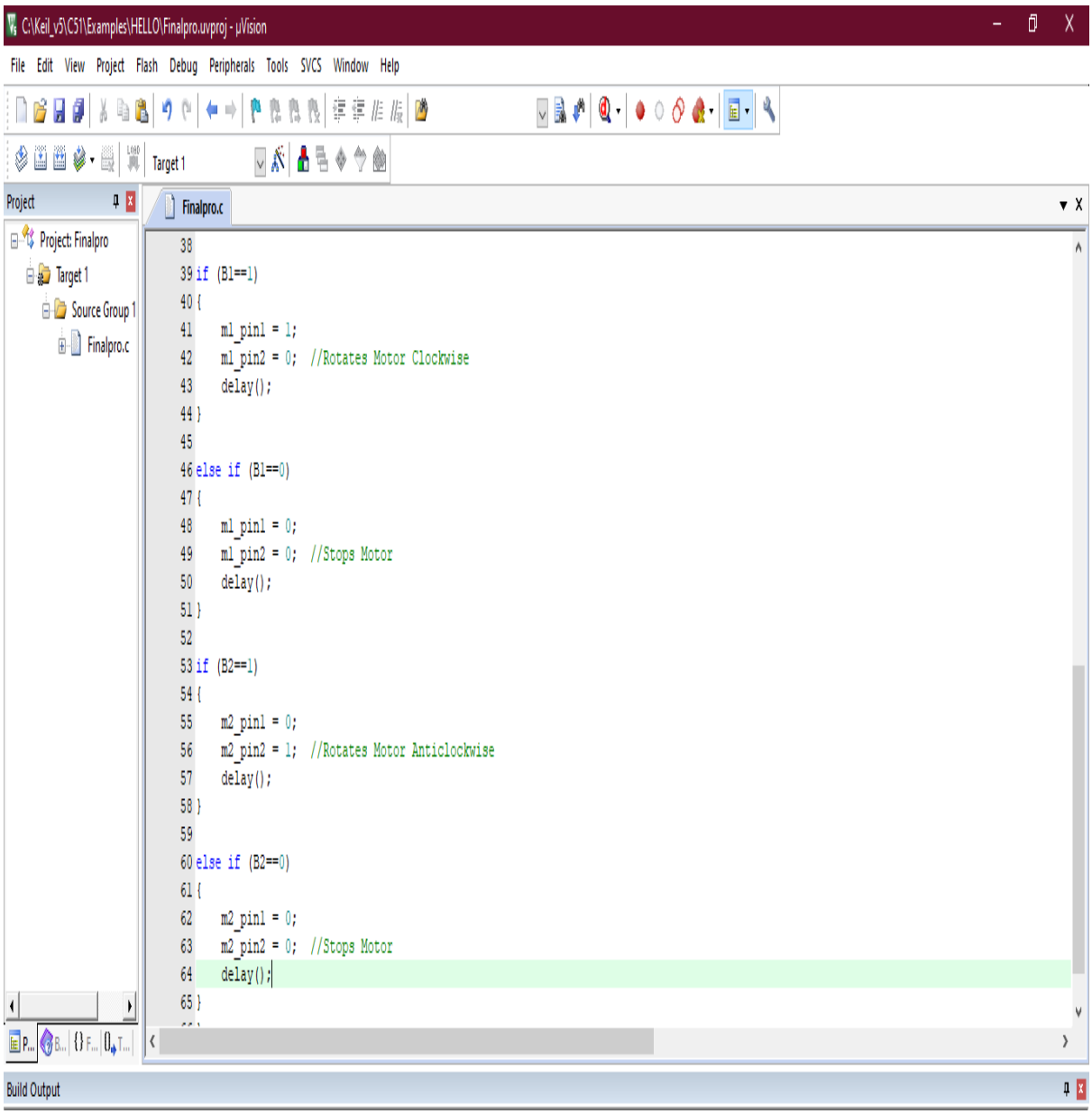


Fig 4: Creation of HEX File

Outputs

Figure-5: Clock wise and anti-clock wise motor Direction control

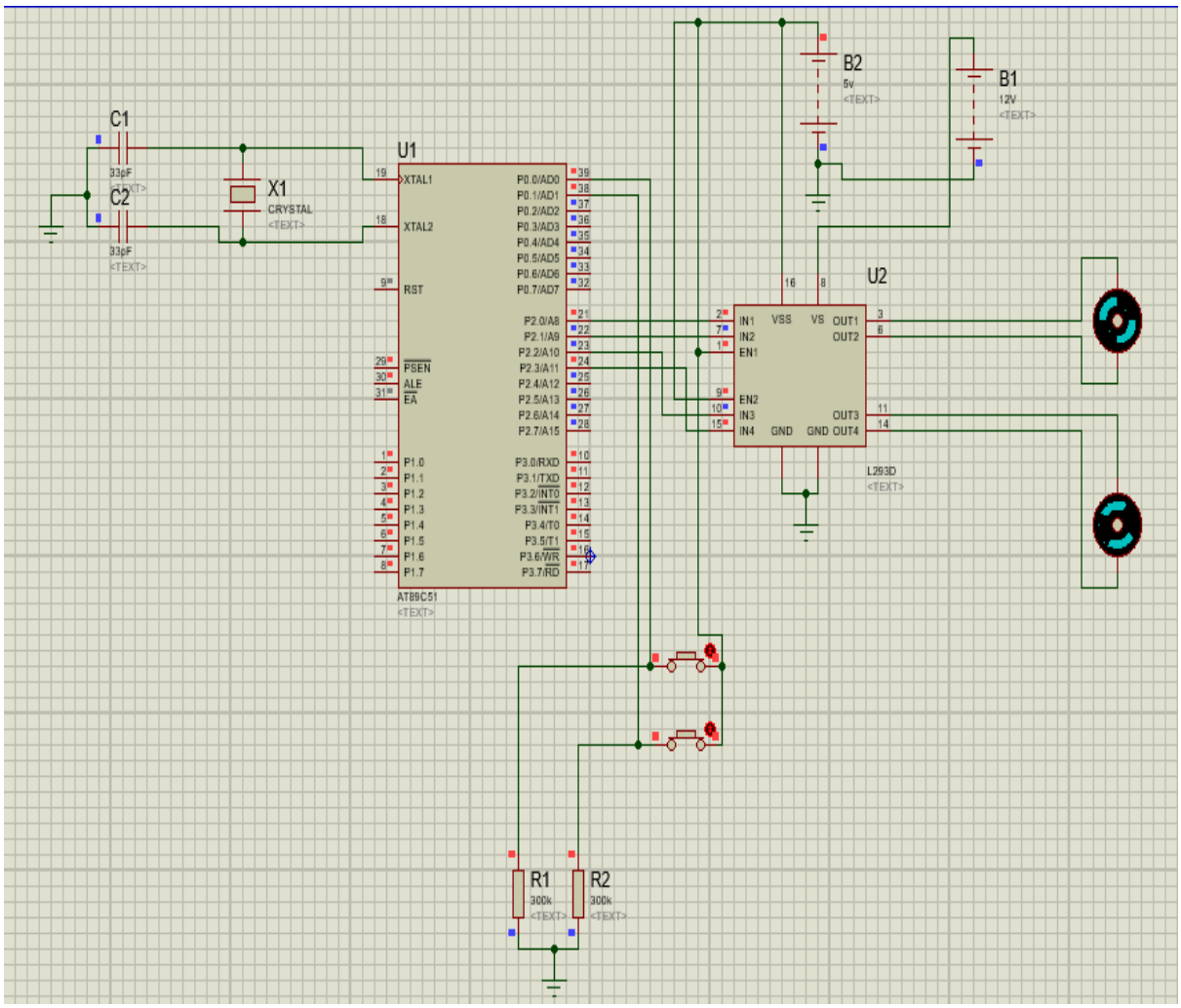


Figure-6: Clock wise motor Direction control with second motor stationary

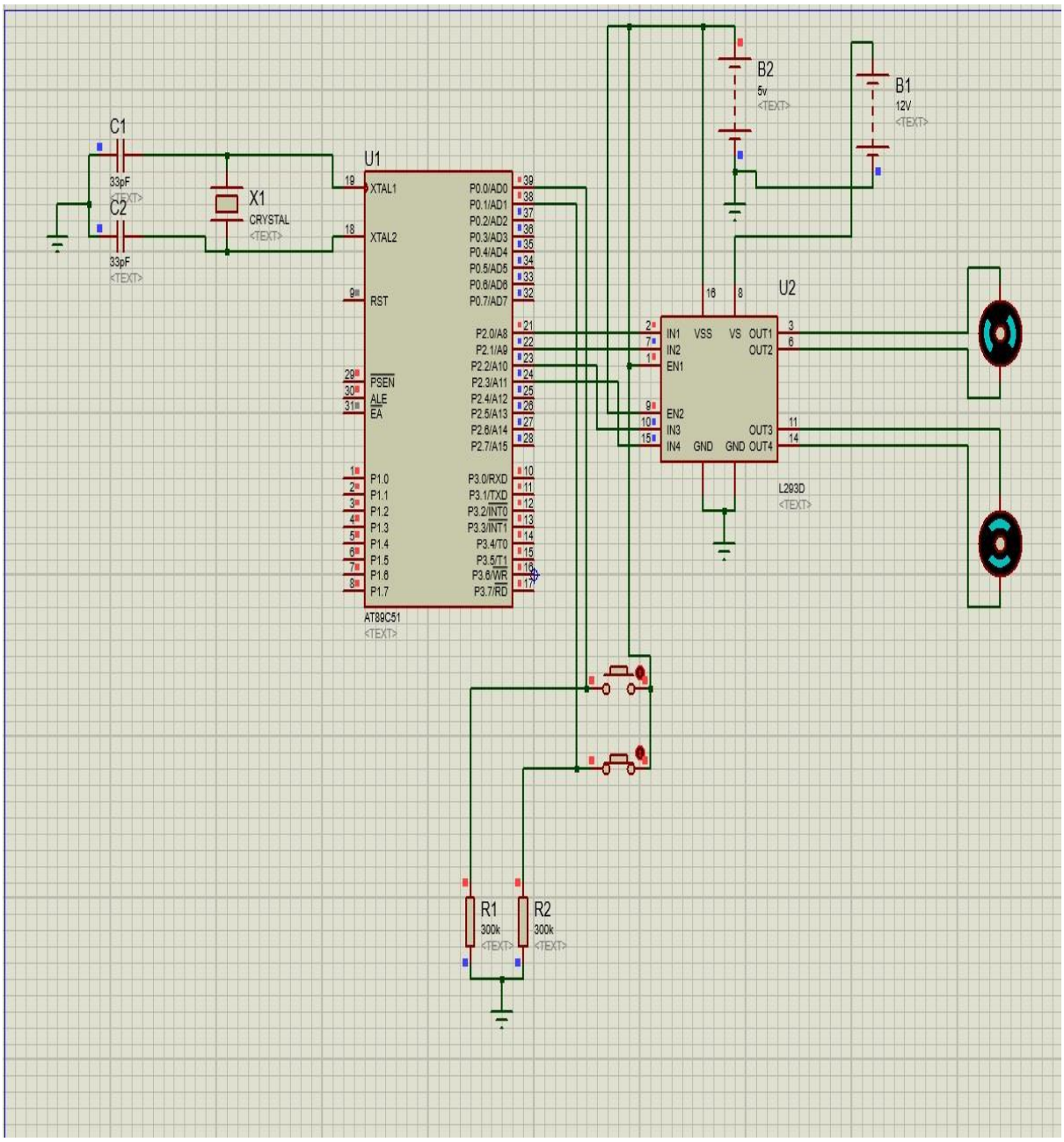
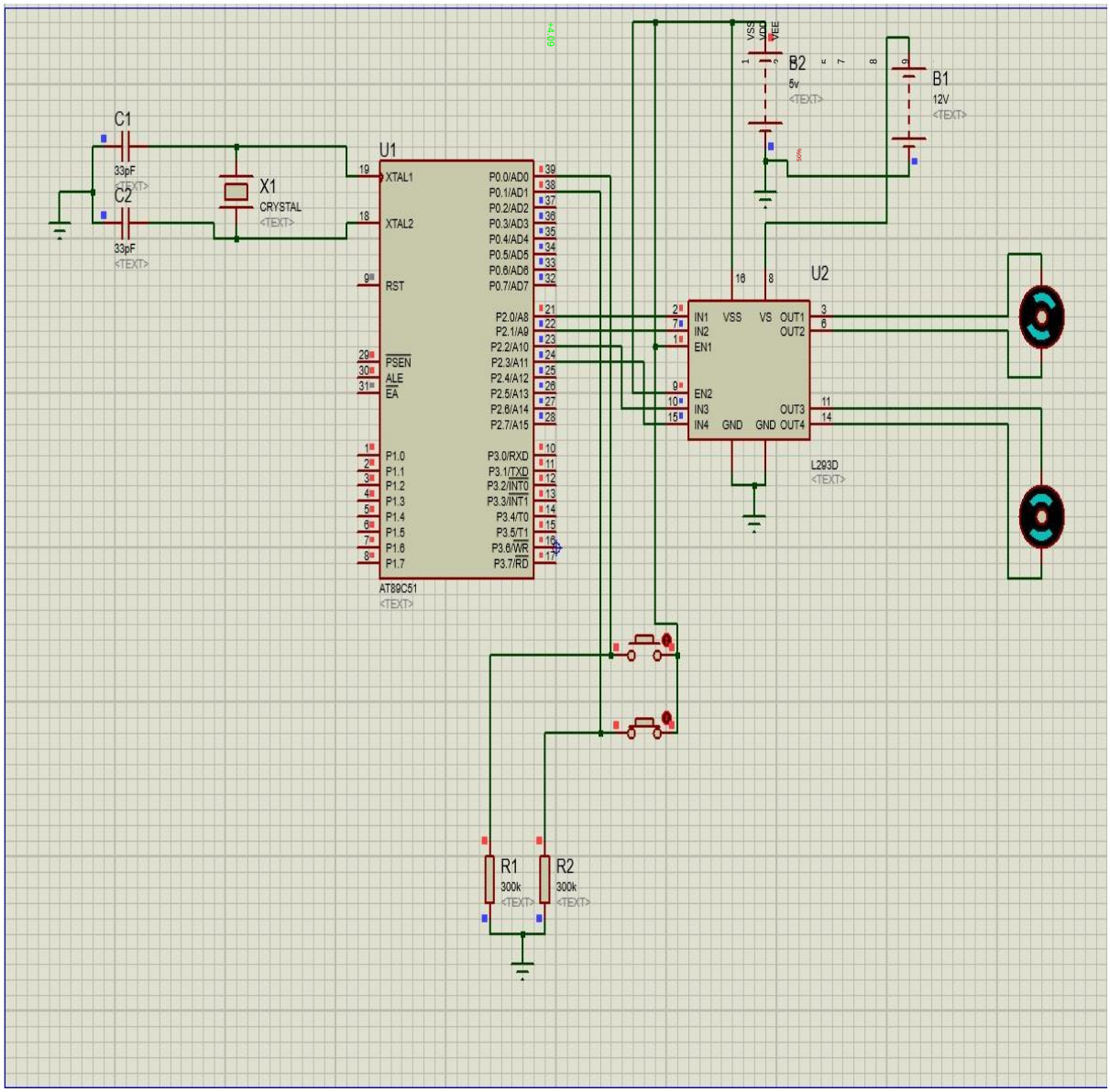


Figure 7: Anti-Clock wise motor Direction control with second motor stationary



CONCLUSION

Hence we can interface the DC motor using 8081 microcontroller and L293D motor Driver IC. This implementation has wide range of applications in Robotics for direction control and at applications where we need to control high power Dc motors.

REFERENCES:

- http://www.keil.com - KEIL MICROVISION
- https://youtu.be/kTtt09uX894
- https://www.electronicshub.org/8051-microcontroller-introduction/
- https://www.labcenter.com/ - PROTEUS