AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Engineering

Lab Report

Experiment # 9

Experiment Title: Implementation of a motor control system using Arduino: Digital input, outputs, and PWM.

Date of Perform:	Date Month 2023	**Date of Submission:**	7 August 2023
Course Title:	MICROPROCESSOR AND EMBEDDED SYSTEM LAB		
Course Code:	COE3104	**Section:**	N/A
Semester:	Summer 2022-23	**Degree Program:**	BSc in CSE/EEE
Course Teacher:	**Prof. Dr. Engr. Muhibul Haque Bhuyan**		

Declaration and Statement of Authorship:

1. I/we hold a copy of this Assignment/Case-Study, which can be produced if the original is lost/damaged.
2. This Assignment/Case-Study is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgment is made.
3. No part of this Assignment/Case-Study has been written for me/us by any other person except where such collaboration has been authorized by the concerned teacher and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared, and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the Faculty Member for review by any internal/external examiners.
7. I/we understand that Plagiarism is the presentation of the work, idea, or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offense that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual forms, including electronic data, and oral presentations. Plagiarism occurs when the origin of the source is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or copy my/our work.

** Student(s) must complete all details except the faculty use part.*
** Please submit all assignments to your course teacher or the office of the concerned teacher.

Group #

Sl No	Name	ID	PROGRAM	SIGNATURE

Faculty use only

FACULTY COMMENTS	**Marks Obtained**	
	Total Marks	

Table of Contents

Experiment Title: Implementation of a motor control system using Arduino: Digital input, outputs, and PWM.

Objectives:
The objectives of this experiment are to-

- Familiarize the students with the PWM signals generated by the Arduino.
- Control the speed of a DC motor using the PWM signals generated by the Arduino.
- Change the direction of rotation of a DC motor using the input push switch.

Equipment List:

1) Arduino UNO board
2) L298N Driver
3) 12 V High Torque DC Motor with Fan Blades connected to it.
4) A DC Power Supply
5) Breadboard
6) Jumper Wires.

Circuit Diagram:

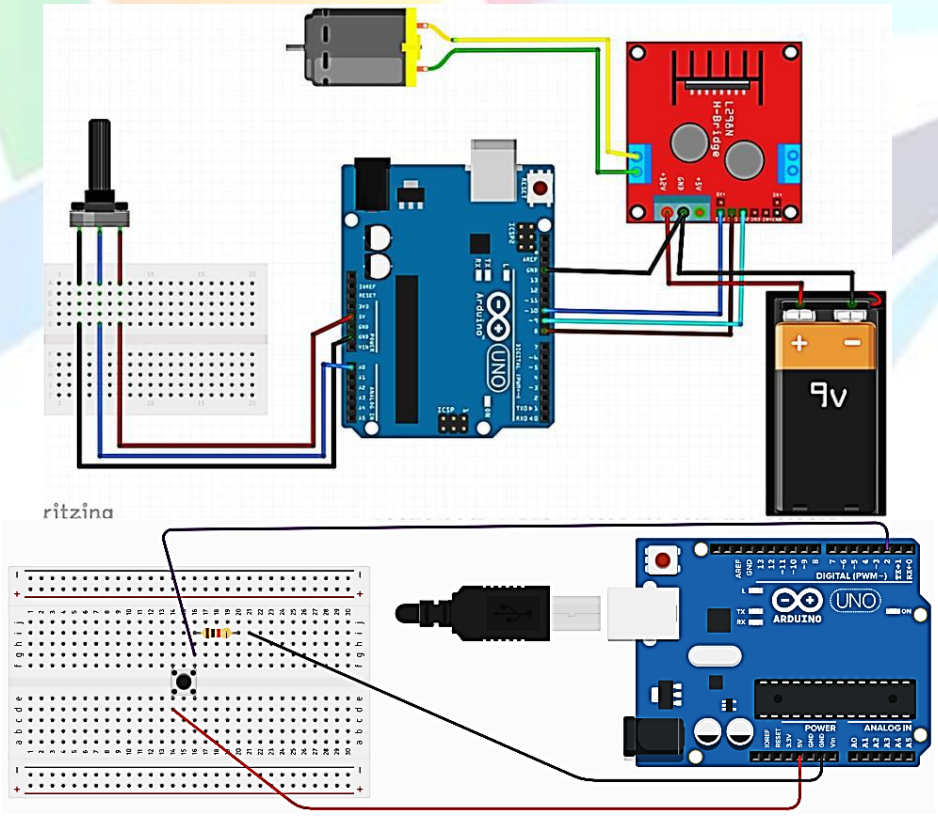

Fig. 1 Experimental setup of a motor control system using Arduino

Code/Program:
The following is the code for the implementation of a motor control system using Arduino with the necessary code explanation:

```
int switchrotation = 2; // Input pin to switch the direction of rotation
int in1 = 8; //Declaring where our module is wired
int in2 = 9; //Declaring where our module is wired
int ConA = 10;// This is a PWM DI/DO
int speed1; //Hold variable of speed

void setup() {
      Serial.begin(9600); //Serial begin for serial operation at 9600 baud
      pinMode(switchrotation, INPUT);
      pinMode(in1, OUTPUT);
      pinMode(in2, OUTPUT);
      pinMode(ConA, OUTPUT);
}

void TurnMotorA1() {
      //A function to control the direction and speed in one direction
      digitalWrite(in1, LOW); //Switch between these HIGH and LOW states to
      change direction
      digitalWrite(in2, HIGH);
      float analogvalue = analogRead(A0); // declaring and reading an analog
      voltage value from the pin
      int PWMvalue = map(analogvalue, 0, 1023, 0, 255); // mapping the analog
      readings to change
      // range from 0-1023 to 0-255 to divide the value by 4 to get a PWM value
      analogWrite(ConA, PWMvalue);// To activate the DC motor
      Serial.println("The motor is running in the clockwise direction."); //
      May need to change
      Serial.print("Digital Value = ");
      Serial.print(PWMvalue); //print digital value on serial monitor
      //convert digital value to analog voltage
      float analogVoltage = (PWMvalue *5.00)/255.00;
      Serial.print(" Analog Voltage = ");
      Serial.println(analogVoltage);
}

void TurnMotorA2() {
      //A function to control the direction and speed in another direction
      digitalWrite(in1, HIGH); //Switch between these HIGH and LOW states to
      change direction
      digitalWrite(in2, LOW);
      float analogvalue = analogRead(A0); // declaring and reading an analog
      voltage value from the pin
```

```
      int PWMvalue = map(analogvalue, 0, 1023, 0, 255); // mapping the analog
      readings to change
      // range from 0-1023 to 0-255 to divide the value by 4 to get a PWM value
      analogWrite(ConA, PWMvalue);// To activate the DC motor
      Serial.println("The motor is running in the anticlockwise direction.");
      // May need to change
      Serial.print("Digital Value = ");
      Serial.print(PWMvalue); //print digital value on serial monitor
      //convert digital value to analog voltage
      float analogVoltage = (PWMvalue *5.00)/255.00;
      Serial.print(" Analog Voltage = ");
      Serial.println(analogVoltage);
}

void loop() {
      if (digitalRead(switchrotation) == LOW) {
            TurnMotorA1(); // function that keeps looping to run the motor
            continuously in one direction
      }
      else if (digitalRead(switchrotation) == HIGH) {
            TurnMotorA2(); // function that keeps looping to run the motor
            continuously in another direction
      }
}
```

Hardware Output Results:

Here is the hardware implementation of a motor control system using Arduino and the necessary explanation of the implementation:

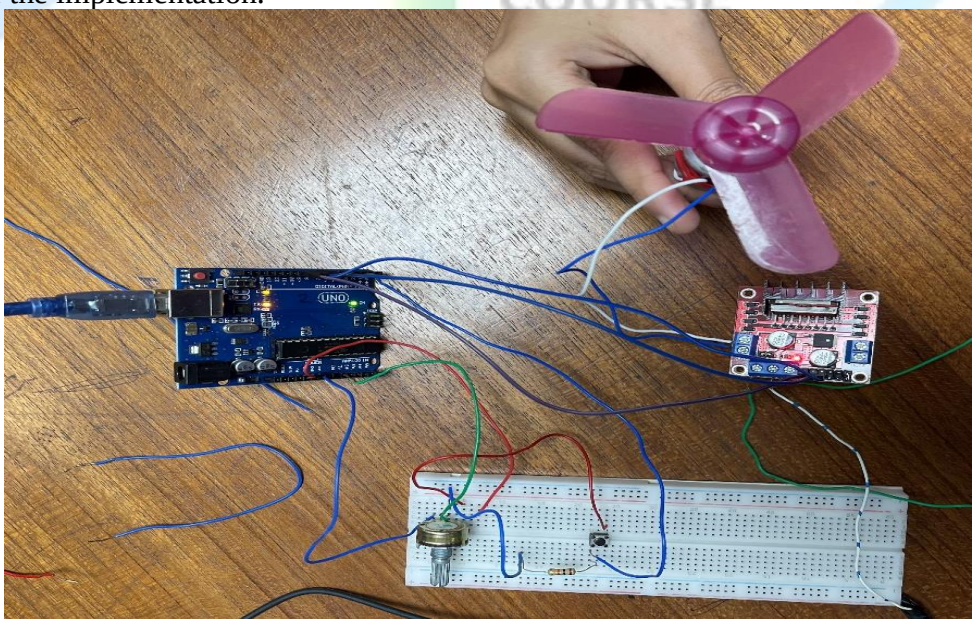

Fig. 2 Hardware implementation of a motor control system using Arduino

Explanation:

In this experiment, an Arduino Uno board was taken to perform the experiment. The potentiometer, resistor of 10 kΩ and push switch was set on the breadboard. Also, a L298N Driver and 12 V High Torque DC Motor with Fan Blades were used in the experiment. The DC power supply was connected with the L298N driver. The Vcc was connected with the +12V socket and the ground was connected with the GND socket. The DC motor with fan blades was connected with the output port of the L298N driver. The anode port was connected with the out1 port and the cathode port was connected with the out2 port of the L298N driver. Wires were connected with the port on the Arduino Uno with the following ports : port 2 was connected with the push switch, port 8,9,10 was connected with IN1, IN2 and EN1 of L298N driver and the potentiometer was connected with A0, 5V and GND port.

Experimental Output Results:

Here are the results of a motor control system using Arduino and the necessary explanation of the results:

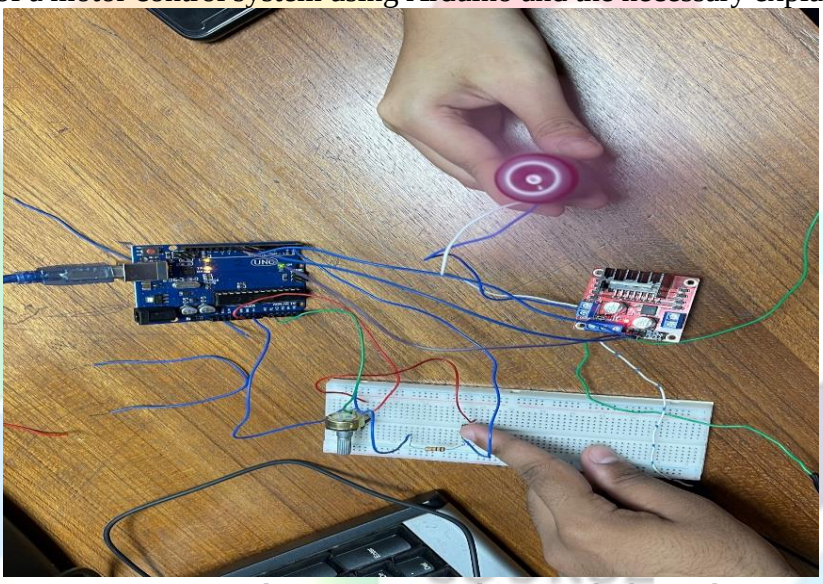

Fig.3 A motor control system using Arduino in Clockwise direction

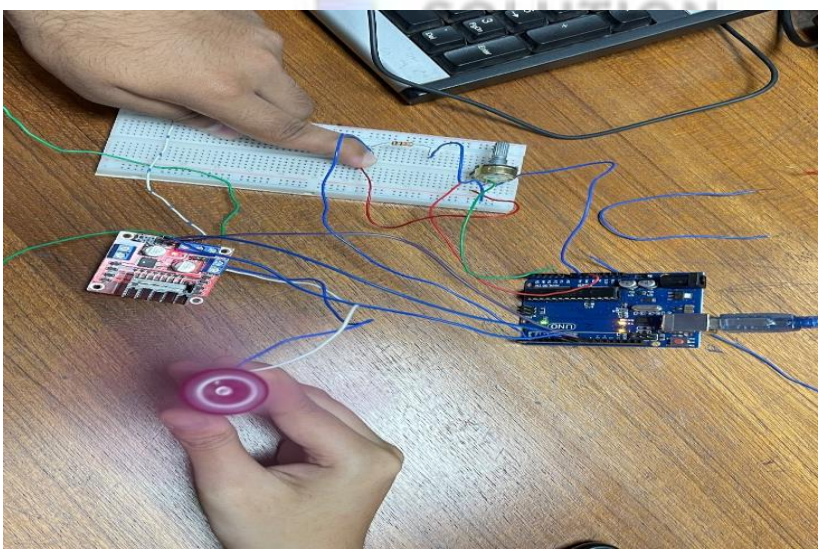

Fig.4 A motor control system using Arduino in Anti-Clockwise direction

Explanation:

In the experiment, the dc motor with fan blades were rotating by adjusting the potentiometer and initially it was rotating in clockwise direction. The fan blades rotate slowly if the potentiometer was set to low and its rotation speed increased by increasing the potentiometer. The direction of the fan blades was changed to anti-clockwise by pressing the push button.

Simulation Output Results:

Here is the simulation implementation of a motor control system using Arduino and the necessary explanation of the implementation:

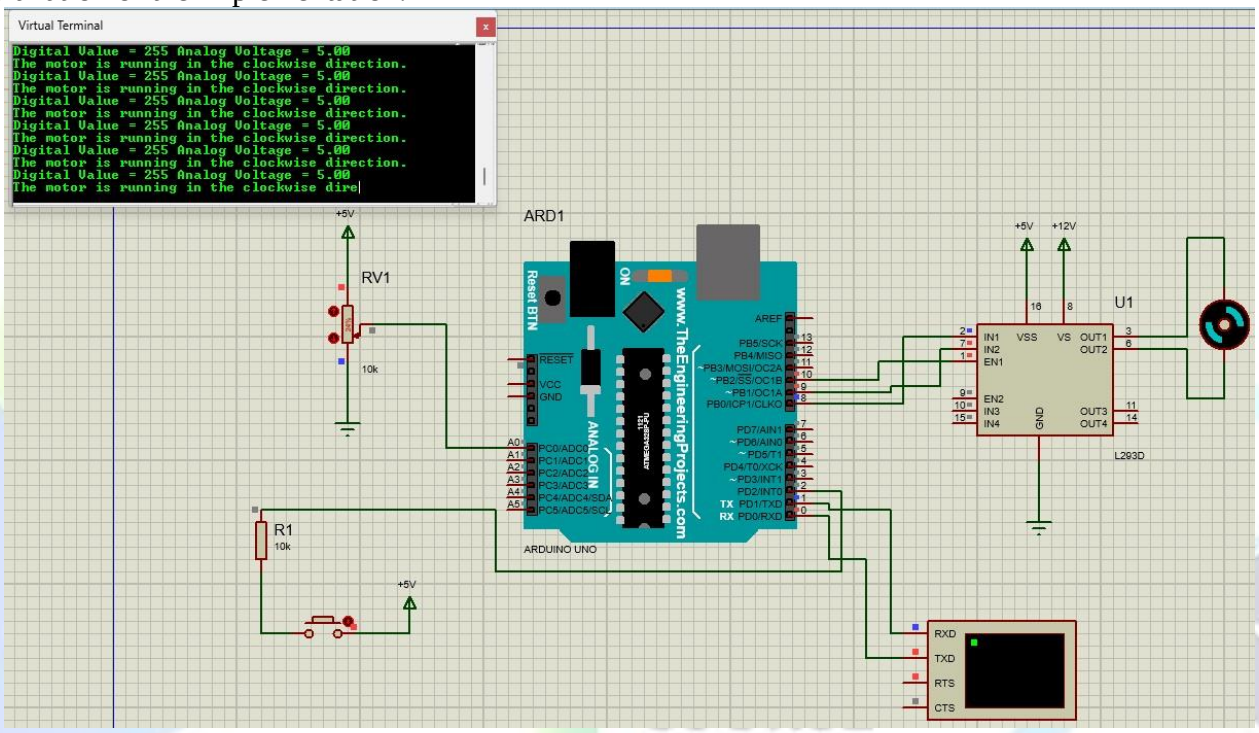

Fig.5 First Simulation where the fan is rotating in Clockwise direction

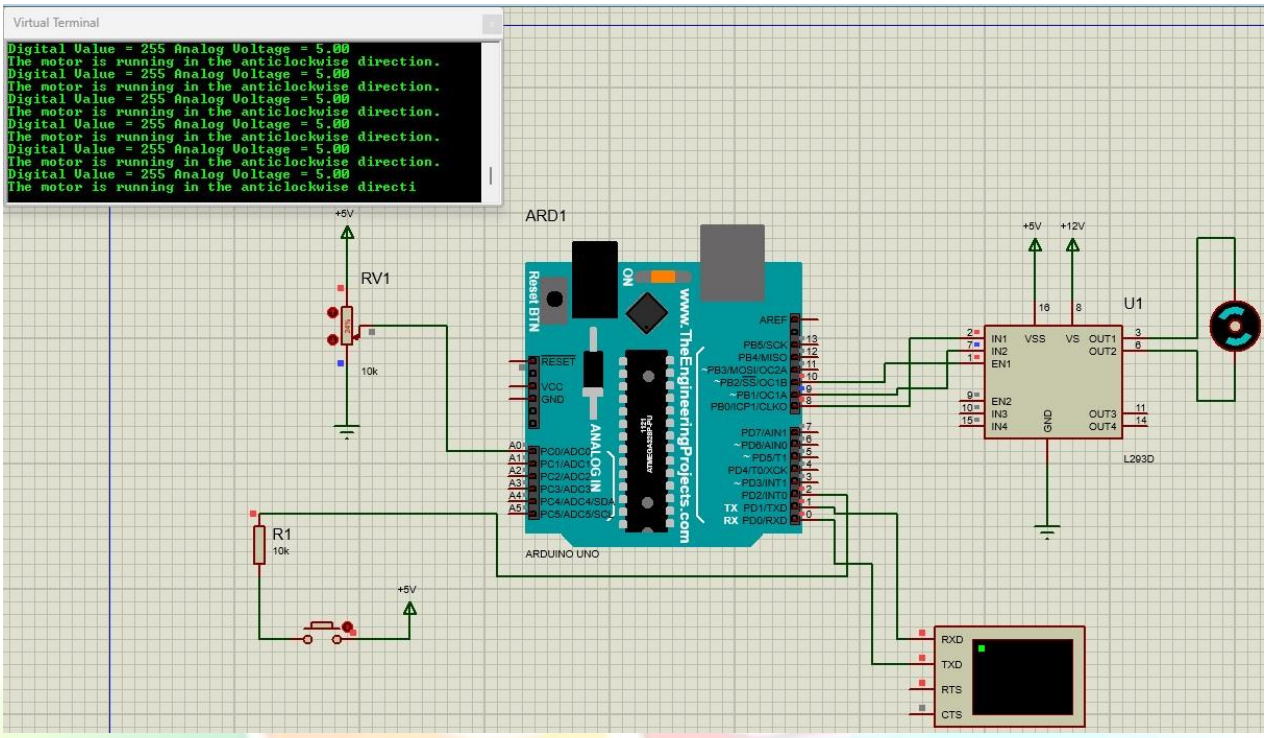

Fig.6 Second Simulation where the fan is rotating in Anti-Clockwise direction

Explanation:

In this experiment, the simulations were done using Proteus software. A new project in Proteus was created. Arduino UNO board was added to the schematic design window from the component's library. Then, the dc motor was placed in the design from the component library. Also, the L293D, push button, potentiometer and resistor were added in the design from the component library. Next, the potentiometer was connected with the A0 port of Arduino Uno board. The push button was connected with 10kΩ resistor and then it was connected with the pin 2 of Arduino Uno board. After that, the ports of Arduino Uno and the anode and cathode port of the dc motor was connected with the L293D driver. To monitor the output a virtual terminal was used and it was connected with port 0 and 1 of the Arduino Uno board. The program was written on the Arduino IDE and the HEX file was generated while compiling. The HEX file was imported in the Arduino Uno board in the Proteus simulation. Lastly, the simulation was observed and the results were obtained accordingly.

Answers to the Questions in the Lab Manual:

3 Configure the port numbers for outputs and inputs according to your ID. Consider four digits from your ID (if your ID is XY-PQABC-Z then consider In1, In2, switchrotation, and ConA from your ID's PQAB positions, if any digit is zero then use C digit. Include all the programs and results in your lab report.

Solution: The following university ID was used:

2	0	-	4	2	6	5	7	-	1
X	Y	-	P	Q	A	B	C	-	Z

So, after taking PQAB position's value as In1, In2, switchrotation and ConA it becomes as below:

In1 = 4

In2 = 2

Switchrotation = 6

ConA = 5

The following is the code for A motor control system using Arduino with the necessary code explanation:

The Modified Code:

```
int switchrotation = 6; // input pin to switch the direction of rotation
int in1 = 4; //Declaring where our module is wired
int in2 = 2;
int ConA = 5;// Don't forget this is a PWM DI/DO
int speed1;
void setup() {
Serial.begin(9600);
pinMode(switchrotation, INPUT);
pinMode(in1, OUTPUT);
pinMode(in2, OUTPUT);
pinMode(ConA, OUTPUT);
}
void TurnMotorA1() { //A function to control the direction and speed in one
direction
digitalWrite(in1, LOW); //Switch between these HIGH and LOW states to change
direction
digitalWrite(in2, HIGH);
float analogvalue = analogRead(A0); // declaring and reading an analog
voltage value from the pin
int PWMvalue = map(analogvalue, 0, 1023, 0, 255); // mapping the analog
readings to change
// range from 0-1023 to 0-255 to divide the value by 4 to get a PWM value
analogWrite(ConA, PWMvalue);// To activate the DC motor
Serial.println("The motor is running in the clockwise direction."); // May
need to change
Serial.print("Digital Value = ");
Serial.print(PWMvalue); //print digital value on serial monitor
//convert digital value to analog voltage
float analogVoltage = (PWMvalue *5.00)/255.00;
Serial.print(" Analog Voltage = ");
Serial.println(analogVoltage);
}
void TurnMotorA2() { //A function to control the direction and speed in
another direction
digitalWrite(in1, HIGH); //Switch between these HIGH and LOW states to change
direction
digitalWrite(in2, LOW);
float analogvalue = analogRead(A0); // declaring and reading an analog
voltage value from the pin
int PWMvalue = map(analogvalue, 0, 1023, 0, 255); // mapping the analog
readings to change
```

```
// range from 0-1023 to 0-255 to divide the value by 4 to get a PWM value
analogWrite(ConA, PWMvalue);// To activate the DC motor
Serial.println("The motor is running in the anticlockwise direction."); //
May need to change
Serial.print("Digital Value = ");
Serial.print(PWMvalue); //print digital value on serial monitor
//convert digital value to analog voltage
float analogVoltage = (PWMvalue *5.00)/255.00;
Serial.print(" Analog Voltage = ");
Serial.println(analogVoltage);
}
void loop() {
if (digitalRead(switchrotation) == LOW) {
TurnMotorA1(); // function that keeps looping to run the motor continuously.
// you can add another one to stop through the delay() function to run for
a certain duration.
}
else if (digitalRead(switchrotation) == HIGH) {
TurnMotorA2();
}
}
```

Here is the simulation of a motor control system using Arduino after modification:

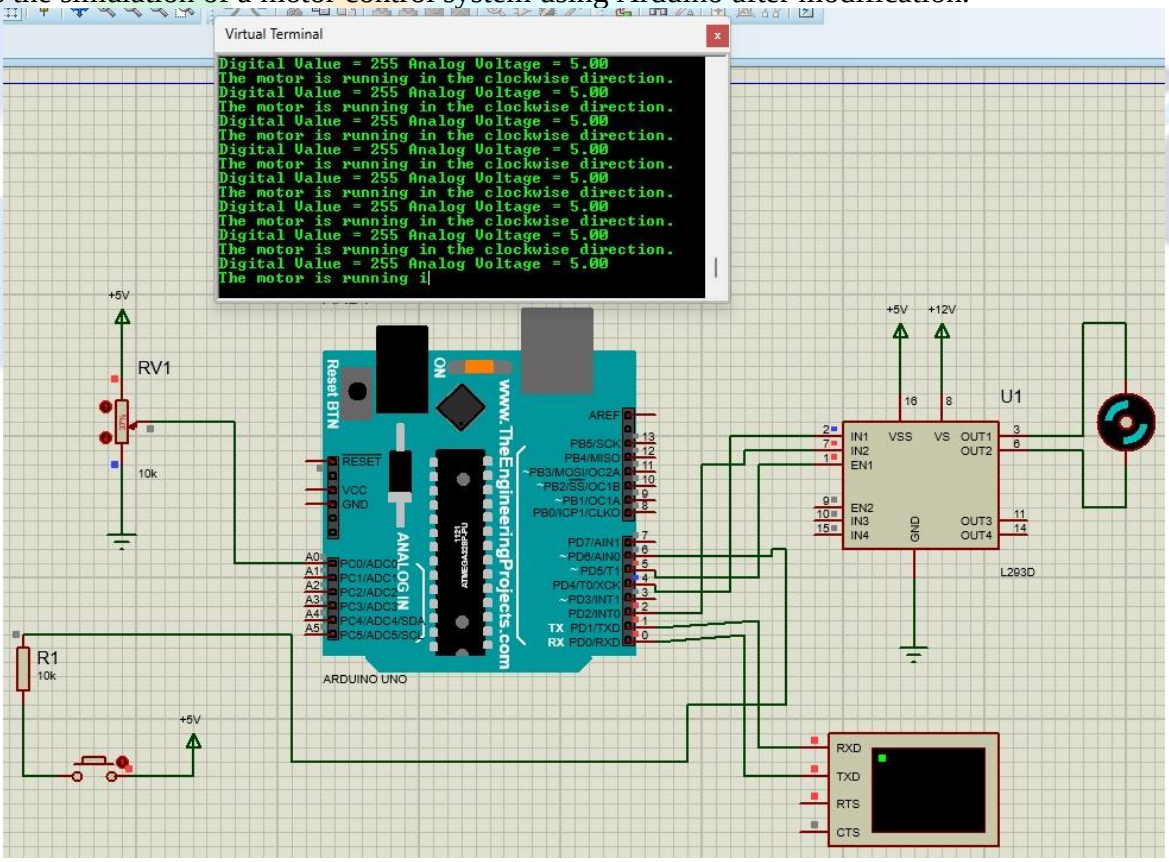

Fig.7 First Simulation where the fan is rotating in Clockwise direction (After Modification)

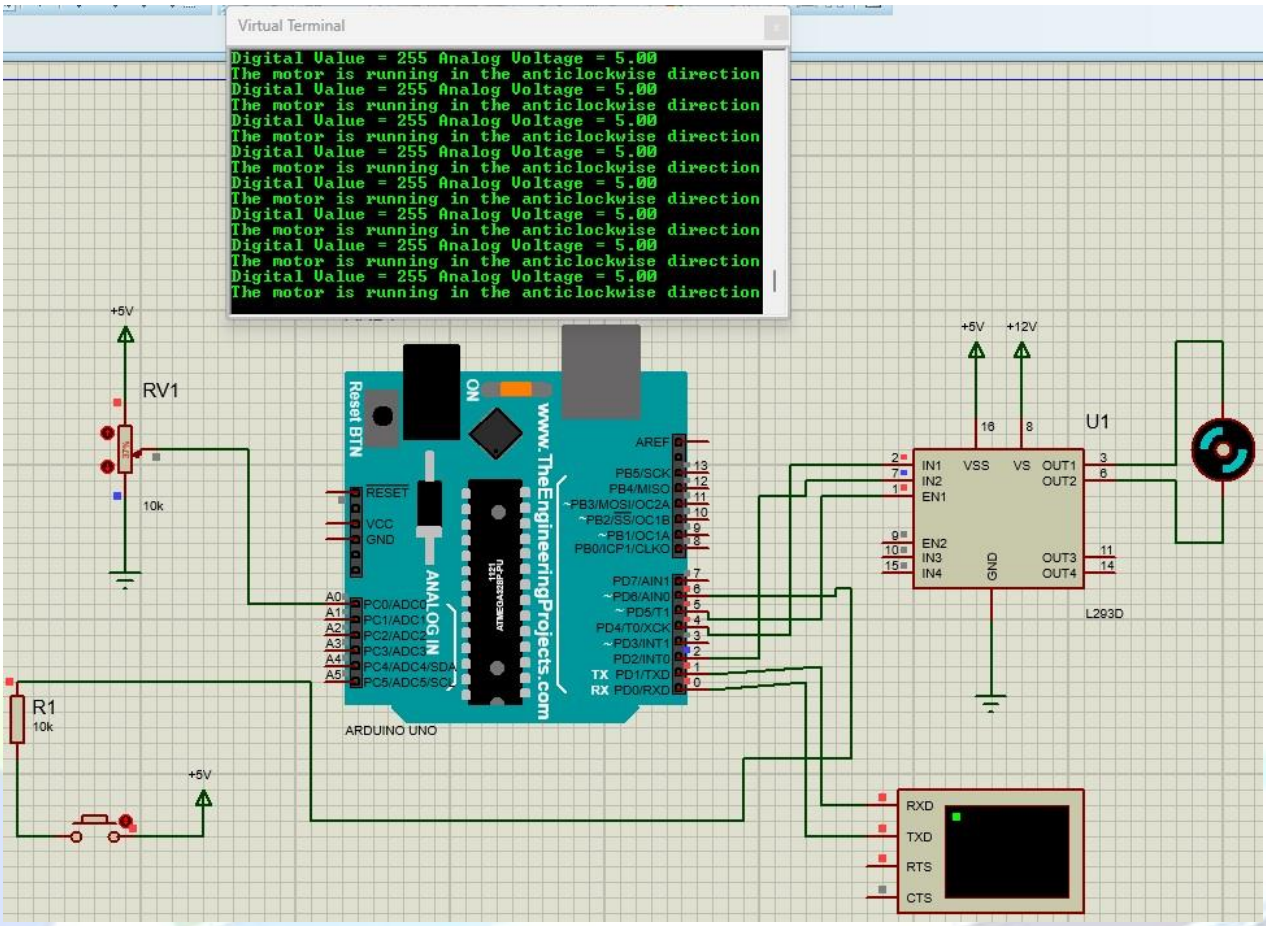

Fig.8 Second Simulation where the fan is rotating in Anti-Clockwise direction (After Modification)

Discussion:

In this microprocessor lab experiment was to implement a motor control system using an Arduino board, focusing on digital input, outputs, and Pulse Width Modulation (PWM) techniques. The Arduino platform offers an accessible and versatile platform for control systems due to its ease of use, affordability, and large community support. During the experiment, we successfully constructed a motor control system using the Arduino board. The system involved interfacing the Arduino with a DC motor, enabling us to control its speed and direction based on various inputs. By employing digital inputs, we were able to receive external signals to the Arduino, allowing us to adjust the motor's behavior dynamically. The digital output pins of the Arduino played a crucial role in controlling the direction of the DC motor. We utilized these output pins to interface with an H-bridge motor driver, which permitted us to change the polarity of the voltage across the motor, thus determining its rotation direction. This aspect of the experiment provided valuable insights into digital control of hardware devices and how to harness the full potential of the Arduino platform. Moreover, the experiment introduced us to the concept of Pulse Width Modulation (PWM), which was utilized to regulate the speed of the DC motor. PWM allowed us to simulate an analog voltage by generating a series of digital pulses with varying duty cycles. By changing the duty cycle of these pulses, we could effectively control the average voltage supplied to the motor, thus adjusting its rotational speed. This technique is widely used in motor control systems as it offers smooth speed regulation while reducing power consumption. One of the major challenges we encountered during the experiment was managing the PWM frequency and duty cycle to ensure efficient motor control. The choice of PWM frequency is crucial as it affects the motor's performance and noise generated during

operation. A lower frequency can result in audible noise, while a higher frequency may lead to reduced torque and efficiency. Careful consideration and testing were required to strike a balance between these factors.

References:

1) Arduino IDE, https://www.arduino.cc/en/Main/Software accessed on 2 nd July 2023
2) Arduino CC Website, [Online] [Cited: July 29, 2023] Available: https://docs.arduino.cc/hardware/uno-rev3
3) *https://www.tinkercad.com/things/b6oU31mFyQa-brilliant-snaget/editel?tenant=circuits, accessed on 2nd July 2023.*
4) AIUB Microprocessor and Embedded Systems Lab Manual 9