AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Science and Technology

Lab Report

Experiment title:	Design a circuit (using Proteus) where a DC Motor runs at three different speeds: 0%, 50%, 100% and show alphabetical order in a 7 segment display.				
Experiment No:	00		Date of Submission:	6 May 2022	
Course Title:	Microprocessor And Embedded Systems				
Group no:	05		Section:	F	
Semester:	Spring	2021-22	Course Teacher:	DR. NADIA ANAM	

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 acknowledgement 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 for review and comparison, including review by 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 formofcheatingandisaveryseriousacademicoffencethatmayleadtoexpulsionfromtheUniversity. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of them arterial used is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or to 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.

No	Name	ID	Program	Signature
01	AKTER, ALAM MORIUM	19-39957-1	CSE	
02	NABIL, JUBAER AHMAD	18-38837-3	CSE	
03	MOBARAK, BHUIYAN SAAD BIN	19-41059-2	EEE	
04	SAKIB, A.B.M. NAZMUS	19-41582-3	EEE	
05	BHUIYAN, FAHIM MAHMUD	20-42970-1	EEE	

Faculty use only		
FACULTYCOMMENTS	**Marks Obtained**	
	Total Marks	

Title:

Design a circuit (using Proteus) where a DC Motor runs at three different speeds: 0%, 50%, 100% and show alphabetical order in a 7 segment display.

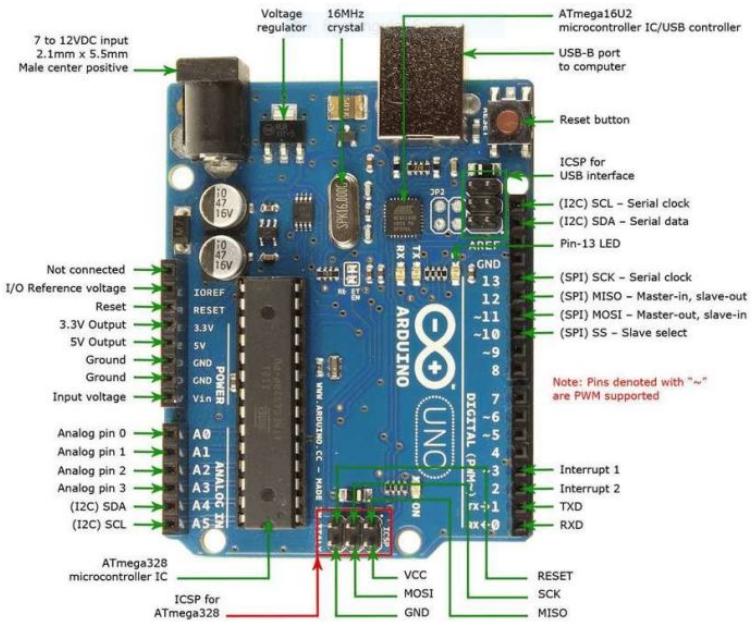
Introduction:

The objective of this experiment is to get familiarized with Microcontroller based motor speed control and to interface a 7 segment display with Arduino Uno. Also, for 0%, 50%, 100% speed of a motor the alphabet A, B, C will be shown in the 7 segment display respectively.

Theory and Methodology:

Arduino is an open-source platform that may be used to make interactive electronics projects. Arduino is made up of a programmable microcontroller and IDE (Integrated Development Environment) software that runs on your computer and is used to create and upload computer code to the microcontroller board. To load new code into the board, the Arduino Uno does not require a hardware circuit (programmer/burner). Using a USB cord and the Arduino IDE (which utilizes a simplified version of C++ to create code), we can quickly load a code into the board.

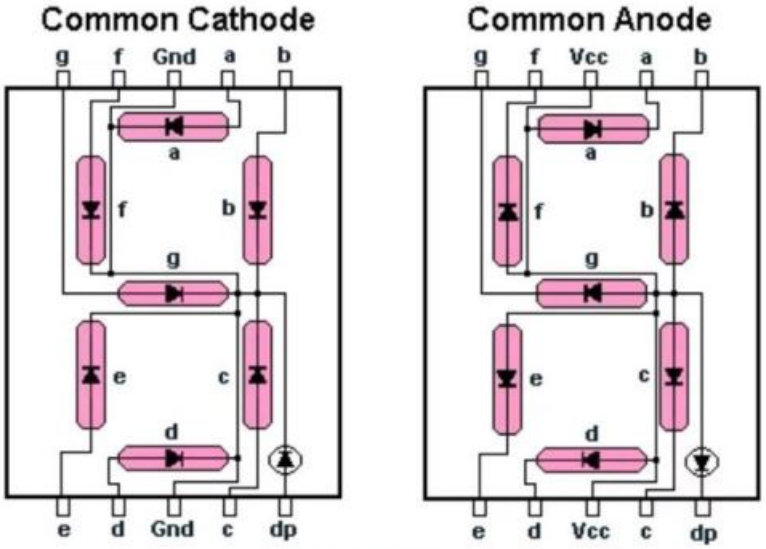
Pin configuration of the board (Arduino Uno R3)



Basis of a 7 segment display:

A seven-segment display is made up of seven LEDs/segments that are all organized in the shape of the number "8." The majority of the segment display is made up of eight parts (with a dot at the right side of the digit representing the decimal point). The seven segments are called in

alphabetical order, from "A" to "G," with "DP" representing the decimal point. And, much like a standard LED, each section may be controlled independently. Seven segment displays are electronic devices which are used as a simpler method to display decimal numerals and a substitute to the more complex dot matrix displays. Initially they grew popular as a way of displaying numbers. Nowadays they are commonly used as displays in home appliances, cars, and various other digital devices. The commonly used as designs and includes seven LED bars aligned in a figure eight pattern as seen in Fig.1. It is capable of displaying the numbers 0-9 and the letters A-F by lighting the appropriate segments.



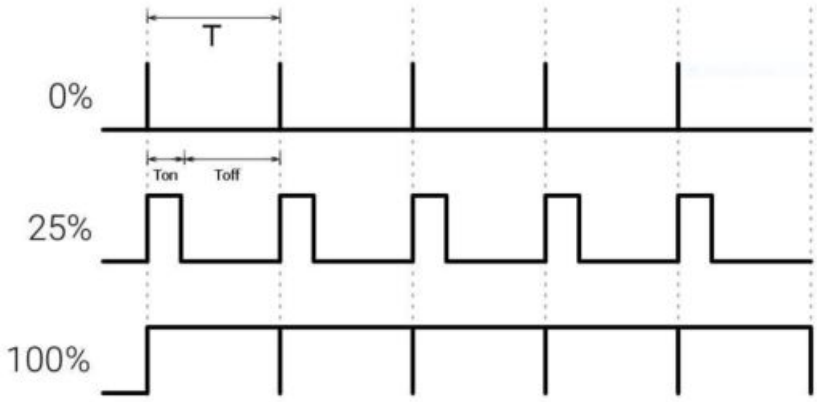
SSD Configuration

Microcontroller based motor

Microcontroller and Arduino are digital devices and they are unable to provide analog output. The microcontroller outputs ZERO and ONE, where ZERO denotes logical LOW and ONE denotes logical HIGH. In our case, we are utilizing the Arduino 5-volt version. As a result, logical ZERO equals zero voltage and logical HIGH equals five voltage.

While digital output is ideal for digital equipment, analog output is sometimes required. PWM is extremely handy in this situation. As illustrated in the diagram below, the output signal flips between zero and one on a high and fixed frequency in the PWM.

The ON time is Ton, and the OFF time is Toff, as indicated in the diagram above. The Time Period is defined as the sum of the Ton and Toff. T is not variable in the PWM concept, but Ton and Toff are. As a result, when Ton increases, Toff decreases, and Toff increases when Ton decreases correspondingly.



Output Signal Of PWM

The duty cycle is a fraction of a period. The duty cycle is usually stated as a percentage or as a ratio. The time it takes for a signal to complete an on-and-off cycle is called a period. A duty cycle can be calculated using the following formula:

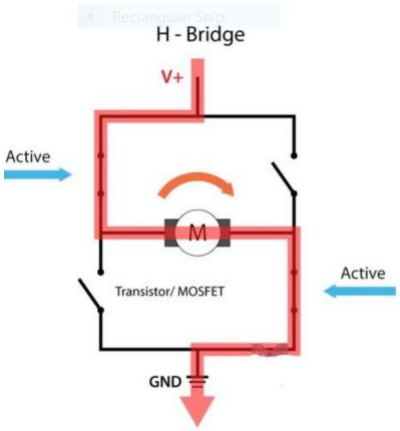
DUTY CYCLE = (Ton / T) x100 %

The motor speed now fluctuates depending on the duty cycle. If the duty is zero, the motor will not operate, and if the duty is 100 percent, the motor will run at maximum RPM. However, this concept is not necessarily correct because the motor only starts operating until a certain fixed voltage, known as the threshold voltage, is applied.

Microcontrollers and Arduinos can process signals and use about 20 to 40 milliamps of current, but motors require much more current and voltage, so we use a transistor to drive the motor. The transistor is in series with the motor, and the transistor's base is connected to the Arduino's PWM pin via a resistor. The transistor acts as a switch, short-circuiting the Emitter (E) and Collector (C) when the PWM signal is in the HIGH state and usually opening when the PWM signal is in the LOW state. This procedure continues indefinitely, with the motors running at the specified speed.

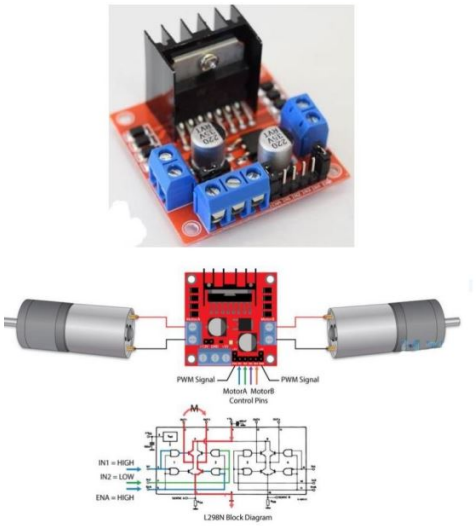
H-Bridge DC Motor Control

We only need to reverse the direction of the current flow through the motor to regulate the rotation direction, and the most popular method for doing so is to use an H-Bridge. An H-Bridge circuit is made up of four switching devices, such as transistors or MOSFETs, with the motor in the middle, producing a H shape. We can change the direction of the current flow and consequently the rotation direction of the motor by activating two specific switches at the same time. Combining these two methods, the PWM and the H-Bridge, we can have complete control over the DC motor. L298N DC motor has been used here.



L298N Driver:

The L298N is a dual H-Bridge motor driver that allows for simultaneous speed and direction control of two DC motors. The module can power DC motors with voltages ranging from 5 to 35V and peak currents of up to 2A. The module has two screw terminal blocks for the motor A and B, and another screw terminal block for the Ground pin, the VCC for motor and a 5V pin which can either be an input or output.



Apparatus

1. L298N Driver
2. 12V High Torque DC Motor
3. Arduino Uno R3
4. 7 segment display
5. Proteus Software 8.13 v1.0
6. Arduino IDE

Using Arduino IDE to write code:

```
int enA=9;
int in1=11;
int in2=12;
int sp_cntrl_1 =A0;
int MotorSpeed1 =0;
int value;
int segA =2;
int segB =3;
int segC =4;
int segD =5;
int segE =6;
int segF =7;
int segG =8;
void setup()
{
Serial.begin(9600);
pinMode (enA, OUTPUT);
pinMode (in1, OUTPUT);
pinMode (in2, OUTPUT);

pinMode (segA, OUTPUT);
pinMode (segB, OUTPUT);
pinMode (segC, OUTPUT);
pinMode (segD, OUTPUT);
pinMode (segE, OUTPUT);
pinMode (segF, OUTPUT);
pinMode (segG, OUTPUT);
```

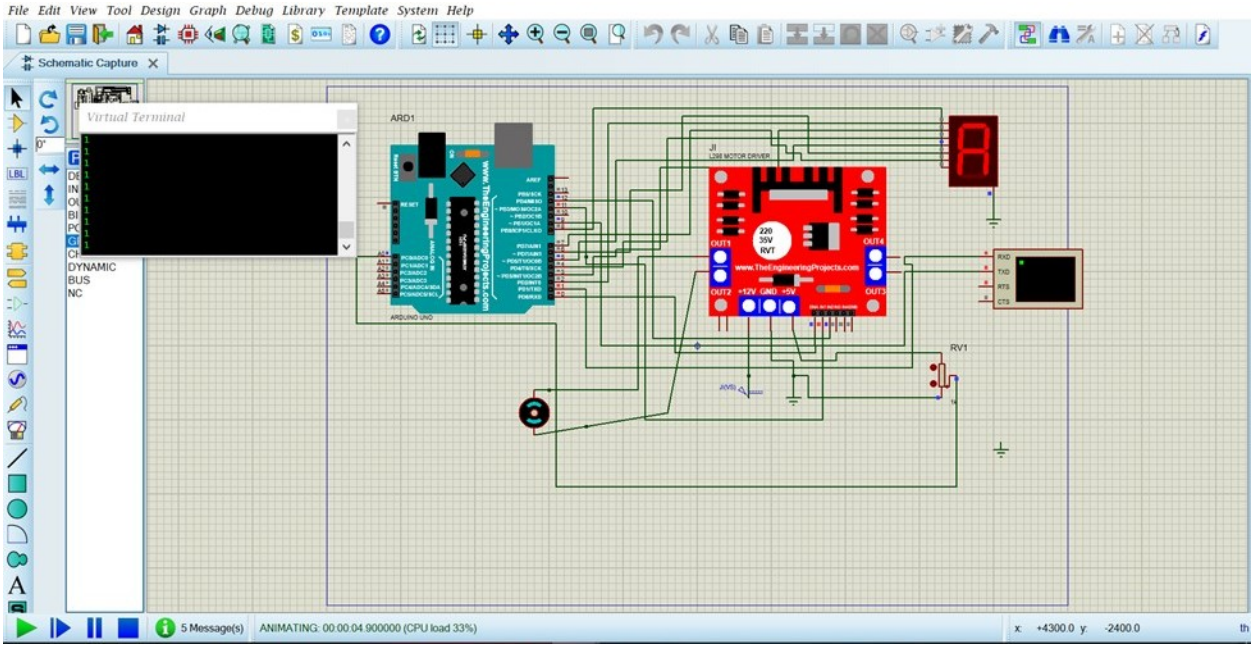
```
}

void loop(){
digitalWrite (in1, HIGH);
digitalWrite (in2, LOW);
MotorSpeed1=analogRead(sp_cntrl_1);
MotorSpeed1=map(MotorSpeed1, 0, 1023, 0, 255);
if(MotorSpeed1<10)
MotorSpeed1=0;
analogWrite(enA, MotorSpeed1);
int value = MotorSpeed1;
value = analogRead(A0);
Serial.println(value);

if (value==1){
digitalWrite (segA, HIGH);
digitalWrite (segB, LOW);
digitalWrite (segC, LOW);
digitalWrite (segD, HIGH);
digitalWrite (segE, HIGH);
digitalWrite (segF, HIGH);
digitalWrite (segG, LOW);
}
else{
digitalWrite (segA, LOW);
digitalWrite (segB, LOW);
digitalWrite (segC, LOW);
digitalWrite (segD, LOW);
```

```
digitalWrite (segE, LOW);
digitalWrite (segF, LOW);
digitalWrite (segG, LOW);
}

if (value==512){
digitalWrite (segA, HIGH);
digitalWrite (segB, LOW);
digitalWrite (segC, LOW);
digitalWrite (segD, HIGH);
digitalWrite (segE, HIGH);
digitalWrite (segF, HIGH);
digitalWrite (segG, LOW);
}

else{
digitalWrite (segA, LOW);
digitalWrite (segB, LOW);
digitalWrite (segC, LOW);
digitalWrite (segD, LOW);
digitalWrite (segE, LOW);
digitalWrite (segF, LOW);
digitalWrite (segG, LOW);
}

if (value==1023){
digitalWrite (segA, HIGH);
digitalWrite (segB, LOW);
```

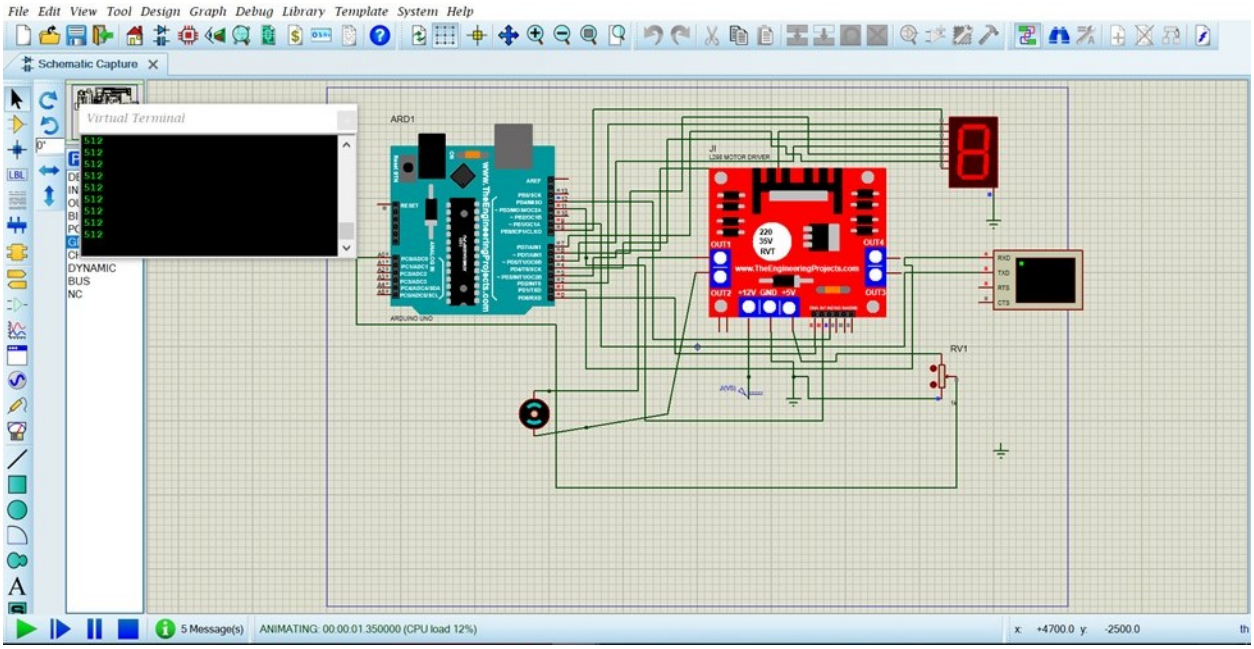
```
digitalWrite (segC, LOW);
digitalWrite (segD, HIGH);
digitalWrite (segE, HIGH);
digitalWrite (segF, HIGH);
digitalWrite (segG, LOW);
}

else{
digitalWrite (segA, LOW);
digitalWrite (segB, LOW);
digitalWrite (segC, LOW);
digitalWrite (segD, LOW);
digitalWrite (segE, LOW);
digitalWrite (segF, LOW);
digitalWrite (segG, LOW);
}
}
```

Simulation:

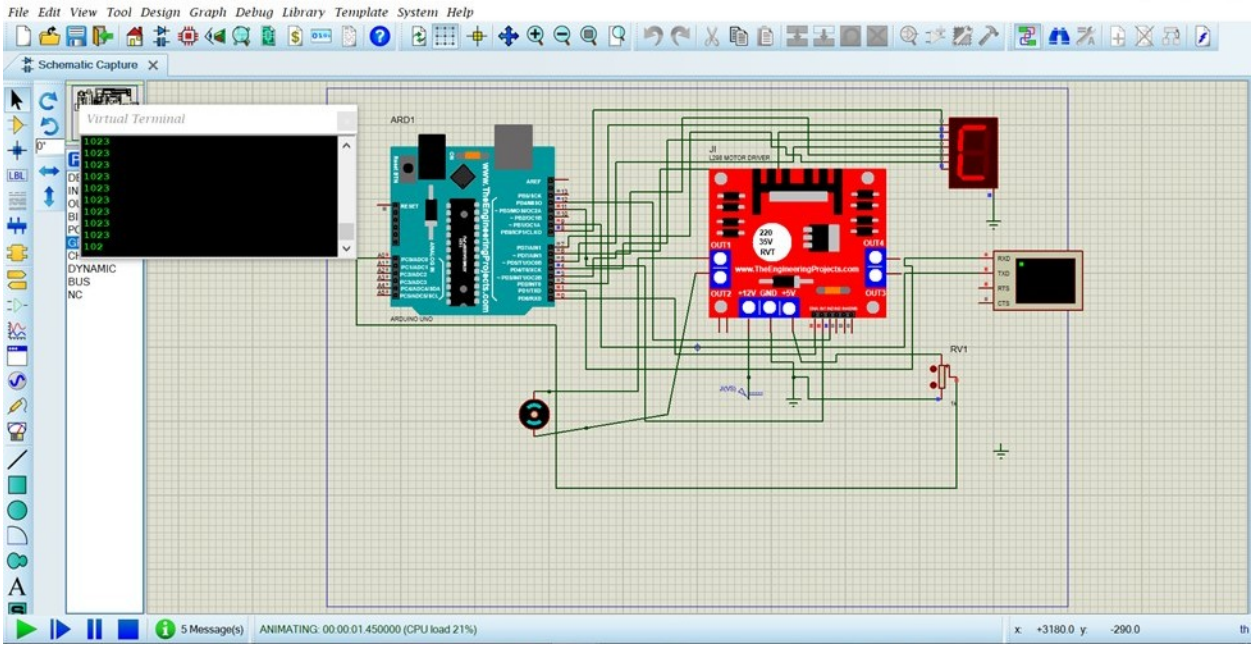
For 0% -- A



For 50% -- B



For 100% -- C



Discussion and Conclusions:

In this experiment, we have used two software Proteus 8.13 Professional and the Arduino IDE software to design the circuit. In this work, the Arduino UNO was used to acquaint the microcontroller and a 7-segment display was used to develop some alphabets in LED segment. All connections were given carefully in the system. Our course teacher outlined the entire procedure, so we applied it with a minimum of effort. We found some bugs or errors but we managed to correct those mistakes successfully run the simulation.

References:

1. https://www.arduino.cc/.
2. HDSP5503 Datasheet.
3. https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-controltutoriall298npwm-h-bridge/