**American International University- Bangladesh (AIUB) Faculty of Engineering (EEE)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name :** | Microprocessor Laboratory | **Course Code :** | COE3104 |
| **Semester :** | 9th | **Sec :** | Q |
| **Lab Instructor :** | Protik Parvez Sheikh |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment No :** | 09 | | |
| **Experiment Name :** | Implementation of a motor control system using Arduino: Digital input, outputs, and PWM |  | |
|  |  |  | |
| **Submitted by (NAME):** | HUMAIRA ZANNAT | **Student ID:** | 22-47789-2 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Group Members** |  | **ID** | **Name** |
|  | 1. | 22-47789-2 | HUMAIRA ZANNAT |
| 2. | 22-47801-2 | SANJUKTA DAS |
| 3. | 22-48021-2 | MD. TANJIL TASHRIK ZIM |
| 4. | 22-48055-2 | MD. ABRAR RAFID SHARIAR |
| 5. | 22-48023-2 | MD. AL-IMRAN SAYEM |
|  |  |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Performance Date :** | 13-05-2025 | **Due Date :** | **20-05-2025** |

**Marking Rubrics (to be filled by Lab Instructor)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Category | Proficient [6] | Good  [4] | Acceptable [2] | Unacceptable [1] | Secured Marks |
| **Theoretical**  **Background, Methods**  **& procedures sections** | All information, measures and variables are provided and explained. | All Information provided that is sufficient, but more explanation is needed. | Most information correct, but some information may be missing or inaccurate. | Much information missing and/or inaccurate. |  |
| **Results** | All of the criteria are met; results are described clearly and accurately; | Most criteria are met, but there may be some lack of clarity and/or incorrect information. | Experimental results don’t match exactly with the theoretical values and/or analysis is unclear. | Experimental results are missing or incorrect; |  |
| **Discussion** | Demonstrates thorough and sophisticated understanding.  Conclusions drawn are  appropriate for analyses; | Hypotheses are clearly stated, but some concluding statements not supported by data or data not well integrated. | Some hypotheses missing or misstated; conclusions not supported by data. | Conclusions don’t match hypotheses, not supported by data; no integration of data from different sources. |  |
| **General formatting** | Title page, placement of figures and figure captions, and other formatting issues all correct. | Minor errors in formatting. | Major errors and/or missing information. | Not proper style in text. |  |
| **Writing & organization** | Writing is strong and easy to understand; ideas are fully elaborated and connected; effective transitions between sentences; no typographic, spelling, or grammatical errors. | Writing is clear and easy to understand; ideas are connected; effective transitions between sentences; minor typographic, spelling, or grammatical errors. | Most of the required criteria are met, but some lack of clarity, typographic, spelling, or grammatical errors are present. | Very unclear, many errors. |  |
| Comments: |  |  |  | Total Marks  (Out of ): |  |

**Title of the Experiment:**

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

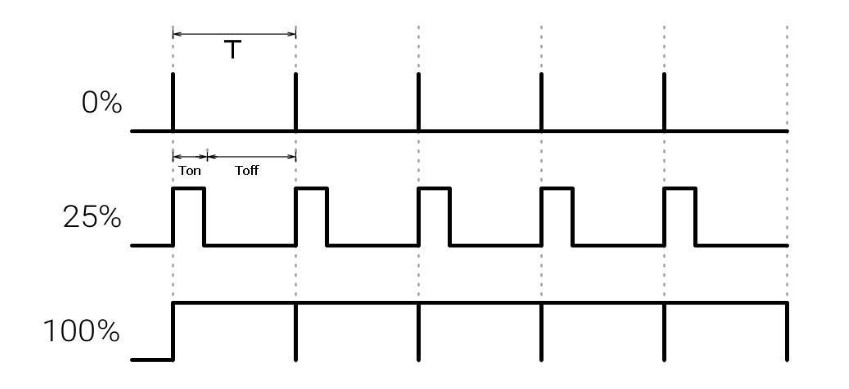
**Introduction:**

The objectives of this experiment are to-

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

**Theory and Methodology:**

Microcontrollers and Arduino are digital devices; they cannot give analog output. The microcontroller gives Zero and ONE as output, where ZERO is logical LOW and ONE is logical HIGH. In our case, we are using a 5-volt version of the Arduino. So, its logical ZERO is zero voltage, and logical HIGH is 5 voltages. The digital output is good for digital devices but sometimes we need analog output. In such a case the PWM is very useful. In the PWM, the output signal switches between zero and one, on a high and fixed frequency, as shown in the figure below.

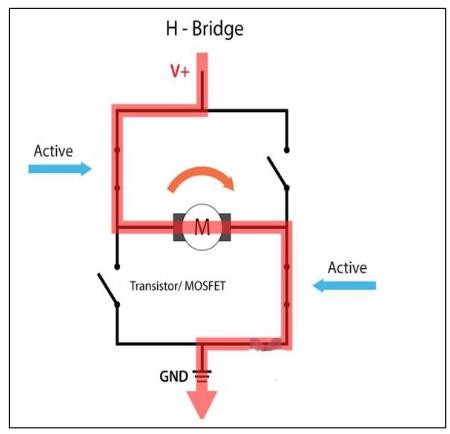


# Figure-1: Output Signal Of PWM

As shown in the above figure the ON time is Ton and the OFF time is Toff. T is the sum of the Ton and Toff, which is called the Period. In the concept of PWM, T is not varying, and the Ton and the Toff can vary, in this way when Ton increase Toff will decrease, and Toff increase when Ton decrease proportionally. The duty cycle is a fraction of one Time period. The duty cycle is commonly expressed as a percentage or a ratio. A period is the time it takes for a signal to complete an on-and-off cycle. As a formula, a duty cycle may be expressed as:

DUTY CYCLE = (Ton / T) x 100 %

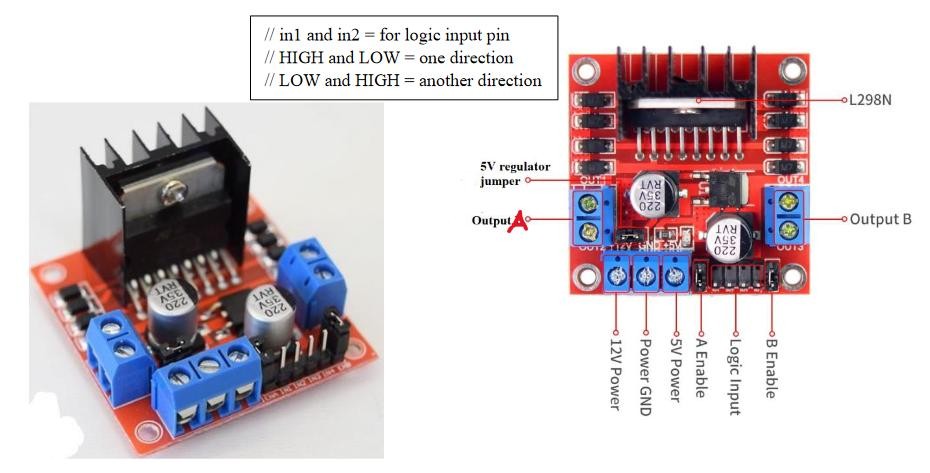
**H-Bridge for DC Motor Control-** The H-bridge circuit, composed of four switching elements, is a versatile tool for controlling DC motors. By strategically activating pairs of switches, it inverts the current flow through the motor, thereby reversing its rotation direction. This simple yet effective mechanism provides a convenient solution for controlling both the direction and speed of DC motors.



# Figure-2: H-Bridge for DC Motor Control

In conjunction with pulse-width modulation (PWM), the H-bridge enables precise control over the motor's rotational speed. By varying the duty cycle of the PWM signal, the average voltage applied to the motor can be adjusted, resulting in a proportional change in its rotational speed.

**L298N Dual H-Bridge Motor Driver-** The L298N is a versatile dual H-bridge motor driver that allows you to control two DC or stepper motors simultaneously. It can operate with a wide range of voltages from 5 to 35 volts and can handle peak currents up to 2 amperes. You can easily control the direction and speed of the motors using logic inputs. Additionally, the L298N supports pulse-width modulation (PWM) for precise speed control. For optimal performance, it is recommended to use an external power supply.

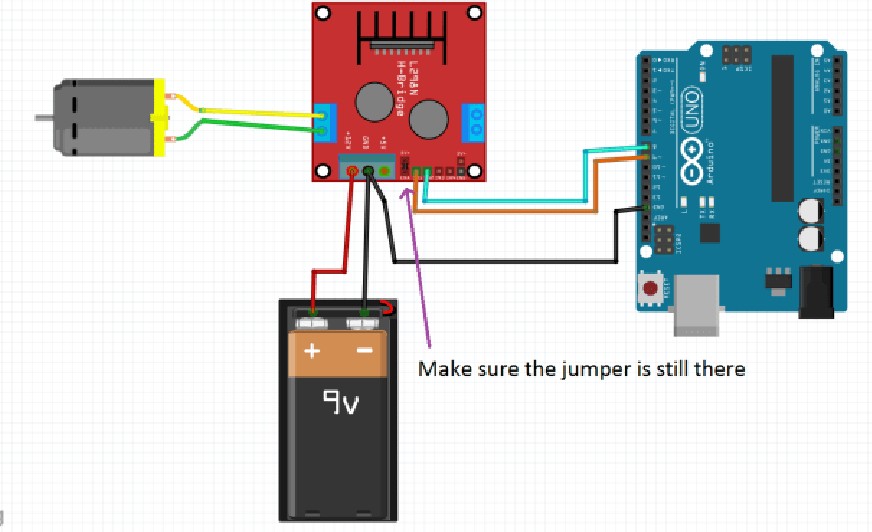


# Figure-3: L298N Dual H-Bridge Motor Driver

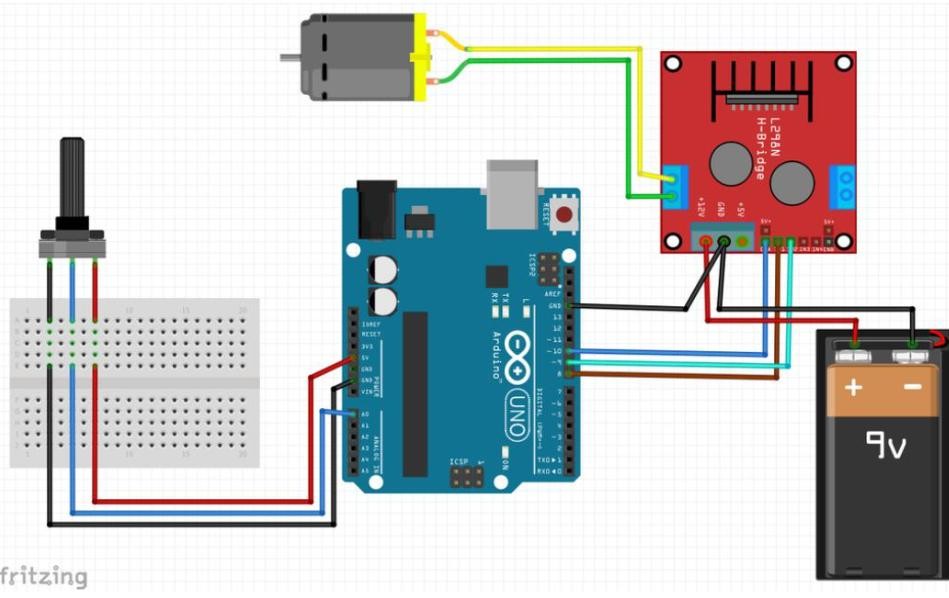
Pinout and Functionality:

* Motor terminals: Screw terminal blocks for connecting motors A and B.
* Power supply: VCC for motor power and 5V input/output.
* Logic inputs: Control motor direction (Forward, Backward, Stop) and speed.
* Enable pins: Enable/disable motors and control speed with PWM.

**Experimental setup:**



# Figure-4: Setup for no speed control and change of direction



# Figure-5: Setup for speed control using potentiometer

**Apparatus:**

1. Arduino board [1]
2. L298N Driver
3. 12 V High Torque DC Motor with Fan Blades connected to it.
4. Potentiometer, push switch, and a resistor of 10 k𝛀
5. A DC Power Supply
6. Breadboard
7. Jumper Wires

**Experimental Procedure: A. Initial Setup**

1. A USB cable Type-A to Type-B was used to connect the Arduino Uno board [1] to the PC.

1. MotorA control pins in1 and in2 was connected to pin-8 and 9 of the Arduino board.

1. When speed control is not required it was made sure that jumper was shorted for the EnA and EnB pins of the L298 driver as shown in figure-4.

1. For speed control of the motor the PWM Signal was connected to pin-10 of the Arduino board, and it was made sure that the jumper was open for the EnA and EnB pins of the L298 driver.

1. A potentiometer was used for speed control where the Vcc and GND of POT was connected to the 5V and GND pin whereas the output pin of the POT was connected to the PWM pin A0 of the Arduino board as shown in figure-5.

1. **Using Arduino IDE to write code for the motor control system**

**PART-1: (No speed control)**

|  |
| --- |
| int in1 = 9; int in2 = 8; void setup(){ pinMode(in1, OUTPUT); pinMode(in2, OUTPUT);  } void TurnMotorA(){ digitalWrite(in1, HIGH); digitalWrite(in2, LOW);  } void TurnOFFA(){ digitalWrite(in1, LOW); digitalWrite(in2, LOW);  }  void loop(){ TurnMotorA(); delay(3000); TurnOFFA(); delay(2000);  } |

**PART-2: Turn on and change the direction (No speed control)**

|  |
| --- |
| int in1 = 9; int in2 = 8; void setup(){ pinMode(in1, OUTPUT); pinMode(in2, OUTPUT);  } void TurnMotorA(){ digitalWrite(in1, HIGH); digitalWrite(in2, LOW);  }  void TurnOFFA(){ digitalWrite(in1, LOW); digitalWrite(in2, LOW);  }  void TurnMotorA2(){ digitalWrite(in1, LOW); digitalWrite(in2, HIGH);  }  void loop(){ TurnMotorA(); delay(3000); TurnOFFA(); delay(2000); TurnMotorA2(); delay(3000); TurnOFFA(); delay(2000); } |

**PART-3: Speed control**

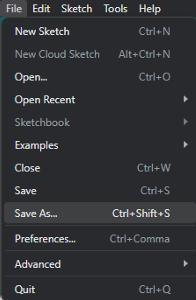
|  |
| --- |
| int in1 = 9; int in2 = 8; int ConA = 10;  void setup(){ pinMode(in1, OUTPUT); pinMode(in2, OUTPUT); pinMode(ConA, OUTPUT); } void TurnMotorA(){ digitalWrite(in1, LOW); digitalWrite(in2, HIGH); analogWrite(ConA, 100); } void TurnOFFA(){ digitalWrite(in1, LOW); digitalWrite(in2, LOW); analogWrite(ConA, 0); } void TurnMotorA2(){ digitalWrite(in1, LOW); digitalWrite(in2, HIGH); analogWrite(ConA, 250); } void loop(){ TurnMotorA(); delay(2000); TurnOFFA(); delay(2000); TurnMotorA2(); delay(4000); TurnOFFA(); delay(2000); } |

**PART-4: Speed control using potentiometer**

|  |
| --- |
| int in1 = 8; int in2 = 9; int ConA = 10; int speed1; void setup(){ pinMode(8, OUTPUT); pinMode(9, OUTPUT); pinMode(10, OUTPUT);  }  void TurnMotorA(){ digitalWrite(in1, LOW); digitalWrite(in2, HIGH); speed1 = analogRead(A0); speed1 = speed1 \* 0.2492668622; analogWrite(ConA, speed1);  }  void loop(){  TurnMotorA();  } |

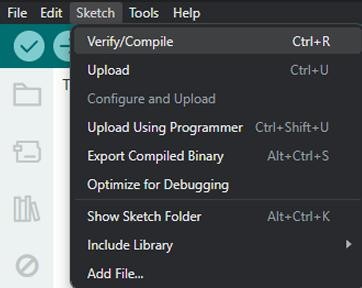
1. **Compile and upload user code from Arduino IDE to Arduino Uno board**

1. After the program was written, the sketch had to be saved. This was done by going to File->Save As, giving a file name and selecting Save.



# Figure-6: Save user code in Arduino IDE

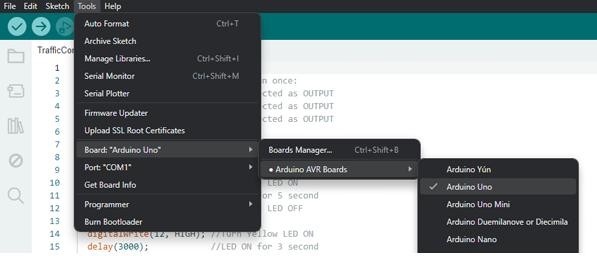
2. Next, the code needed to be verified/compiled to identify and correct any errors. This was achieved by going to Sketch->Verify/Compile.



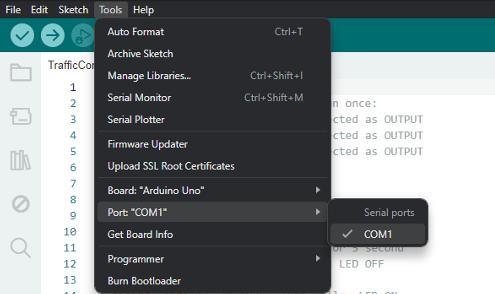
# Figure-7: Verify/Compile user code in Arduino IDE

1. After the compilation was completed, the code needed to be uploaded to the Arduino Uno board. The Arduino Uno R3 board was connected to the PC using a USB cable.

1. Before uploading the code, the board type and port had to be selected in the Arduino IDE. This was done by navigating to:
   * Tools-> Board: -> Arduino AVR Boards -> Arduino Uno.
   * Tools->Port-> COMx.



# Figure-8: Select board type in Arduino IDE



# Figure-9: Select port in Arduino IDE to upload user code

5. After selecting the board and port, the upload option in the Arduino IDE was selected to upload the code to the Arduino board.

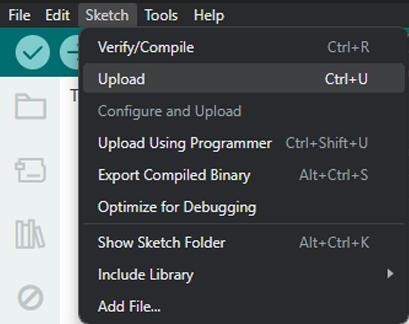
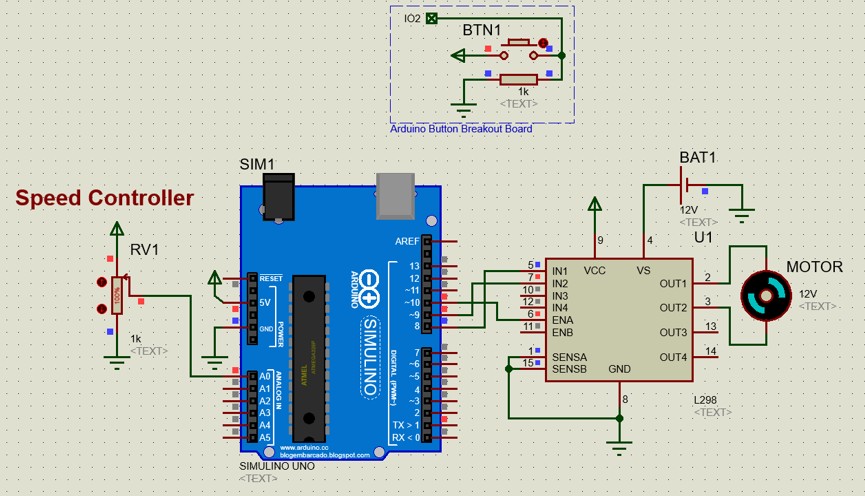


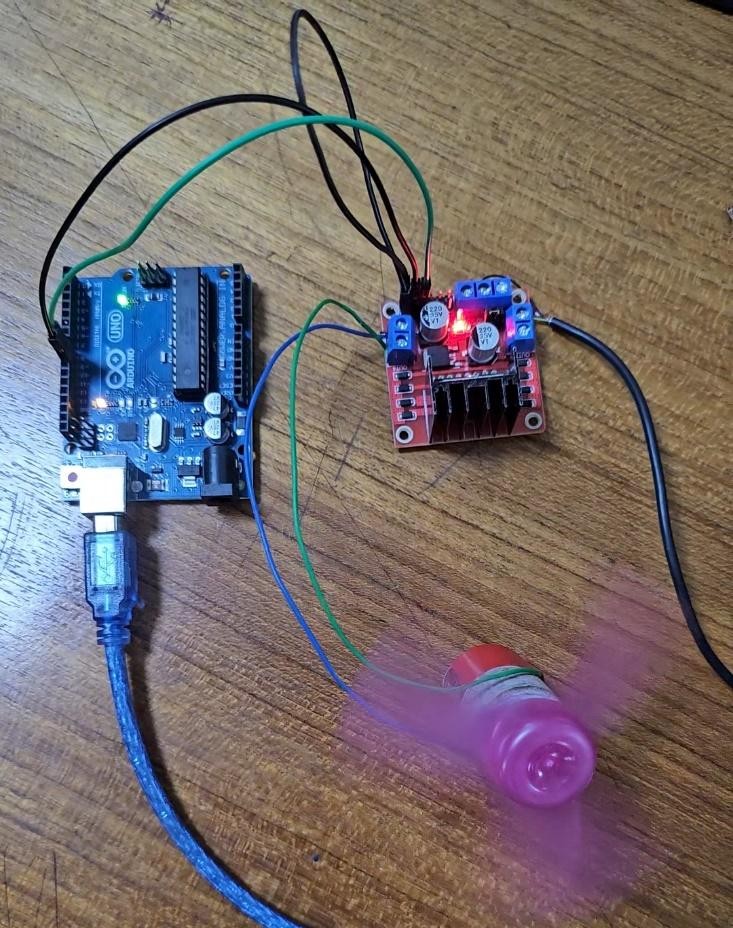
Figure-10: Upload user code from Arduino IDE

**Simulation:**

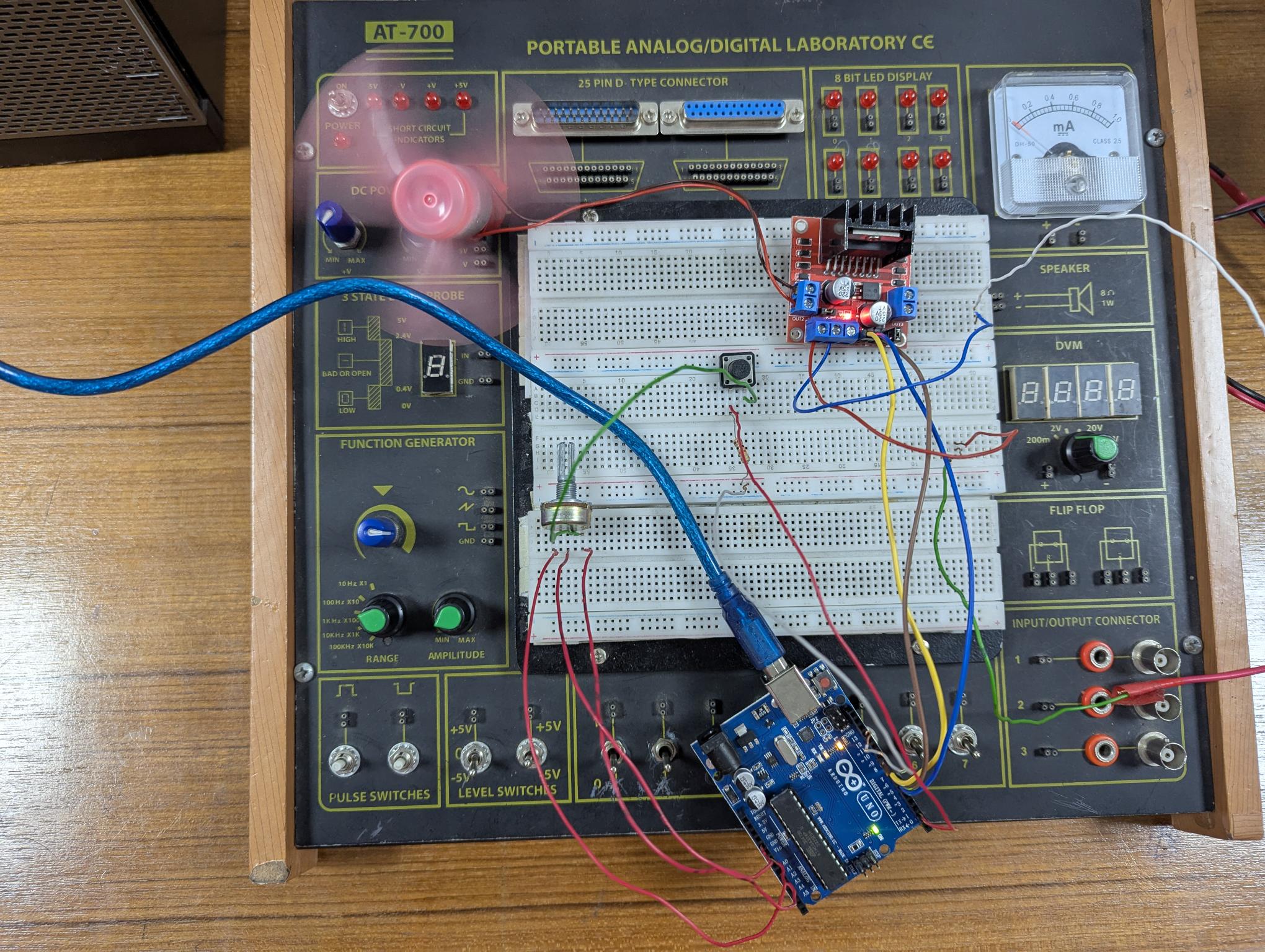


# Figure-11: Simulation of Motor Control System

**Experimental Findings:**



# Figure-12: Lab setup of Motor Control System without Speed Control

 Figure-13: Lab setup of Motor Control System with Speed Control

**Discussion:**

In this experiment,

* DC motor control was explored, including rotational direction and speed modulation.
* An analog sensor was used for real-time speed adjustments.
* Pulse Width Modulation (PWM) on the EnA pin was used for precise speed control.
* The system was made responsive to analog input variations.
* Clear functions were provided for clockwise and anticlockwise rotations.

The experiment was a success as it effectively demonstrated the principles of DC motor control, including direction and speed modulation.