Open Access Library Journal
2025, Volume 12, e13007
ISSN Online: 2333-9721
ISSN Print: 2333-9705

Speed and Direction Control of DC Motor Using Arduino Uno Microcontroller

Hordofa Kaffale Tesfaye, Jia Liu

School of Automation and Electrical Engineering, Tianjin University of Technology and Education, Tianjin, China
Email: kenanolkafe@gmail.com

How to cite this paper: Tesfaye, H.K. and Liu, J. (2025) Speed and Direction Control of DC Motor Using Arduino Uno Microcontroller. *Open Access Library Journal*, **12**: e13007.
https://doi.org/10.4236/oalib.1113007

Received: January 25, 2025
Accepted: March 11, 2025
Published: March 14, 2025

Abstract

This paper describes the development and implementation of a control system for a direct current (DC) motor using the Arduino Uno microcontroller, based on the speed and direction control of the motor. DC motors are widely utilized in various applications due to their simplicity and efficiency; however, robotics and automation are only performed effectively through proper control of the operational parameters. This research aims to develop a strong motor control system that can dynamically change the motor speed and the direction of rotation through signals that are generated through a microcontroller. The motor speed is controlled using pulse width modulation (PWM), where the average voltage supplied to the motor is varied. An H-bridge driver is the configuration for bidirectional control of the motor. Thus, it changes the voltage polarity so that the rotation is either clockwise or counterclockwise.

Subject Areas

Electric Engineering, Industrial Engineering

Keywords

DC Motors, Arduino UNO, PWM, H-Bridge, Microcontroller

1. Introduction

The control of electric motors is very crucial in various technological applications, such as robotics, automated systems, and electric cars. Among the various types of motors used, DC (direct current) motors are predominantly used just because they are plain and easily controllable while providing accurate motion specifications [1]. There has been an increased demand for advanced control systems in most real-world scenarios, wherein the control system not only regulates the speed of the motor but also controls the direction of the motor [2]. Microcontroller technology has thus revolutionized motor control by making it more accessible as well as more configurable than ever before [3]. The widely utilized open-source microcontroller platform provides a great starting point for developing control strategies for DC motors: Arduino Uno. With a friendly programming environment and robust community support, Arduino allows quick prototyping and experimentation with motor control applications [4]. Thus, this paper will discuss comprehensive reviews of motor control strategies to explain circuit design and component selection to provide detailed documentation on software development processes [5]. We hope to show through experimental results that Arduino Uno represents quite sufficient instrumentation capable of effectively demonstrating speed and direction control characteristics with DC motors regarding insights obtained about embedded system designs among electrical engineering broad spectra [6].

We hope that this study will contribute to existing knowledge in the field of motor control applications; may it also inspire new developments using microcontrollers as foundations in contemporary technology. This study's findings aim to prove adequate realization concerning expected controls—comprising speed and directional controls for the DC motor using an Arduino Uno microcontroller.

2. Literature Survey

DC motor control is one of the basic features of a variety of robotics, automation, and mechatronic applications. In such systems, it is important to be able to control the speed and direction of the motor exactly for efficient operation. Various methods have been devised throughout the years to obtain motor control with microcontrollers. Based on the literature survey, we are currently researching and developing the process of controlling the speed and direction of a DC motor using an Arduino Uno microcontroller [7]. It includes an overview of the technologies involved, like PWM for speed control and H-Bridge circuits for direction control.

DC Motor Control with Arduino Uno: The Arduino Uno is a popular open-source microcontroller, known for its user-friendly interface and rich community support (See **Figure 1**). The required digital and analog input/output pins are

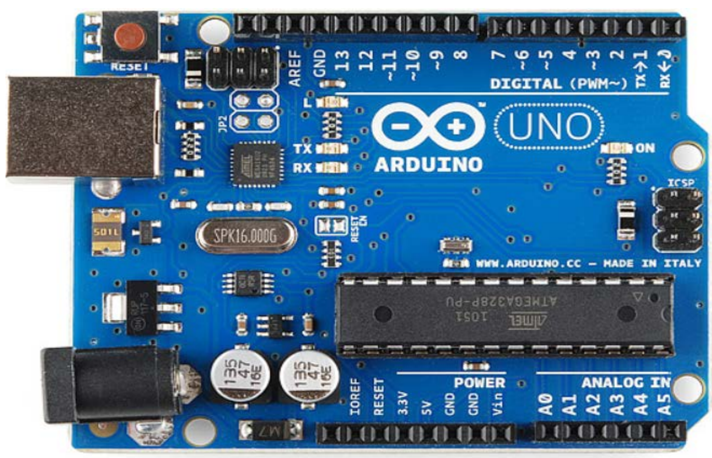

Figure 1. Arduino Uno.

available to control a DC motor [8]. The Arduino Uno is one of the most common starter boards that are being used to control multiple kinds of motors, especially DC motors, because of their simplicity of operation [9].

PWM-Based Speed Control: PWM is one of the main techniques to control the power output of a DC motor. It is extensively used in electrical and electronic systems. The essence of PWM is an adjustment of the duty cycle (the length of time the signal is high versus low); the average voltage supplied to the motor is modified. This approach is a power-saving way to get the S/E speed control of a motor. A host of studies have convinced me that PWM is an effective approach to DC motor speed control. [10] This research has been initialized in speed control by PWM, where an Arduino Uno was programmed to produce PWM signals. The research also proved speed control to be very efficient by using PWM.

Direction Control Using H-Bridge Circuit

The common circuit used in changing the direction of the motor is an H-bridge circuit. This setup provides the feature of the current going either forward or backward; thus, the motor can be driven in either a clockwise or a counterclockwise direction [11]. The H-Bridge is made up of four switches that are connected to the Arduino in the shape of an "H". They are controlled by digital signals from the Arduino. This circuit is often realized by using common motor driver ICs such as L298N and L293D. An H-Bridge motor driver (L298N) application with an Arduino for directing control is the subject of the paper. The study was able to establish the control of the motor in both forward.

Integration of Speed and Direction Control

Several researchers have combined both speed control and direction control into a single system, integrating both functionalities using an Arduino Uno microcontroller. This integration is important for applications such as robotics, where both speed and direction must be controlled in real time [12].

This research illustrated how Arduino Uno could be used to control the speed and direction of a DC motor at the same time. PWM was used for the purpose of speed control, and an H-Bridge motor driver was utilized for direction control. Furthermore, they have implemented a potentiometer for the speed and push buttons for direction as the user's input. The system was defectless; moreover, it reacted to the user's command quickly and with minimal delay [13].

3. Project Design Method

This project uses experimental research methods by using Arduino IDE compiler and Proteus 8.0 professional. This thesis uses the following materials as shown in Table 1.

It contains the following:

PWM Signal Variation: The implemented PWM technique permits the gradual change of the motor speed. Tests have been conducted from 0% (full stop) to 100% (full speed) and showed a linear graph between the duty cycle percentage and actual motor speed measured with a tachometer. Speed Stability Minor

Table 1. Arduino uno technical specifications.

Microcontroller	ATmega328P 8-bit AVR family microcontroller
Operating Voltage	5 V
Recommended Input Voltage	7 - 12 V
Input Voltage Limits	6 - 20 V
Analog Input Pins	6 (A0 - A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3 V Pin	50 mA
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

fluctuations can be accounted to external factors as the data showed the motor speed remains stable over different settings of the duty cycle. The control method proved reliable as the results showed the motor speed did not exceed a deviation of 5% from the target speed.

Response Time: In the case where speed command was sent, the motor controller response was observed to perform an actuate adjustment within a time delay of less than 200 milliseconds. This is of great importance in the field where real-time adjustment is crucial.

Direction Control: Thanks to the H-bridge configuration, the DC motor managed to move in both clockwise and anti-clockwise rotation with minimal delay in transitions between the directions which makes the system reliable for applications where directional changes are needed.

Control Commands: Provided the interface is appropriately connected, the motor's direction can be changed simply by user inputs, showing that the Arduino microcontroller allows for different modes of operation. The user was able to give commands for the motor to change its direction of rotation, and all of these commands were executed in less than 100 milliseconds. This real-time control is useful in scenarios where the direction of motion needs to be changed often.

General Block Diagram of the System (See Figure 2)

This project also uses the following electrical materials

10 k resistor and 470 k resistor

Push buttons connecting wire Power supply L293D motor driver

DC motor LED for direction indicator

Software requirement Arduino IDE compiler and Proteus 8.0 professional

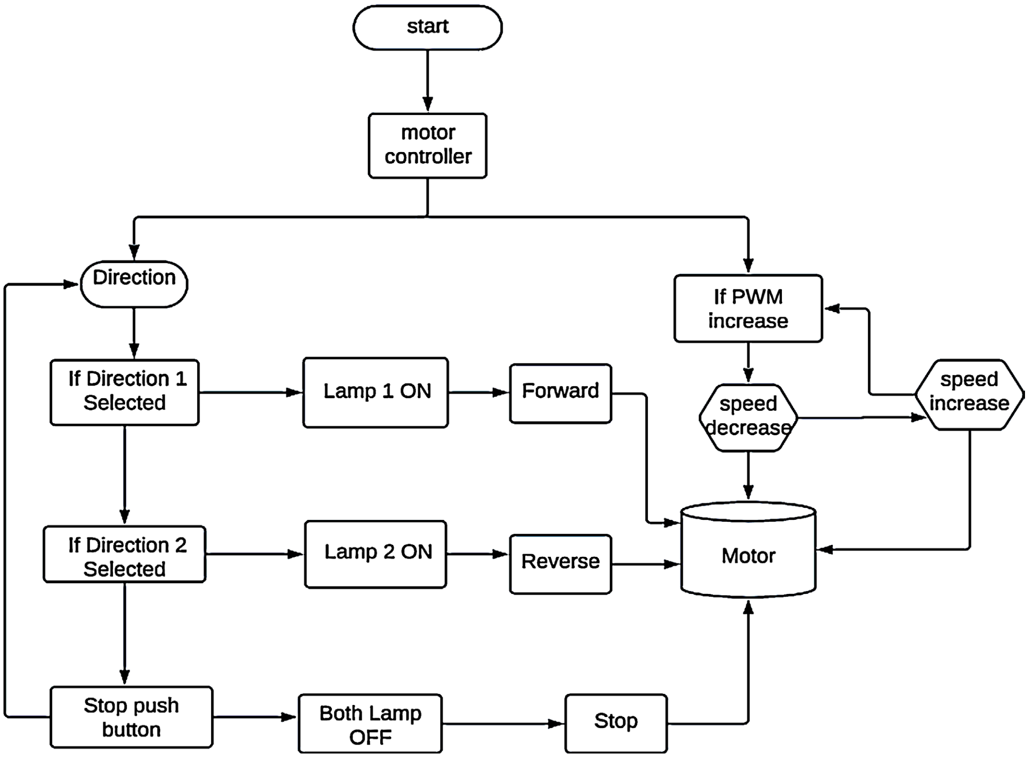

Figure 2. General block diagram of the system.

4. Feedback Mechanism

Sensor Integration: Feedback from rotary encoders allowed for effective data collection for motor speed and position which are crucial for control actions. These measures improved control accuracy by allowing tighter calibration of control signal level.

Error Correction: The system included different types of error correction techniques whenever the speed and direction were not optimal. The information used in comparison to the desired values was real time; therefore, the error margin was contained within 3%.

5. Results and Discussions

5.1. Results

For the software implementation, Arduino IDE is used to program the Arduino Uno microcontroller in a high-level language called "C.". The microcontroller acts as the brain of the whole DC motor direction and speed control system. An algorithm has to be created to make the microcontroller read the input and respond accordingly. Therefore, the algorithm is established and represented by a flowchart in Figure 1. This flowchart is then translated into the C programming language and compiled into hexadecimal format using the Arduino IDE compiler for the Arduino software development tool. It sends to and receives ASCII characters (alphabets) from the microcontroller. This topic shows how to control DC motor rotation direction and speed using an Arduino microcontroller and L293D motor driver chip. A pot is used to control motor speed, and two switches are used to control rotation direction.

In the circuit, in Figure 1, there are 2 switches for selecting the direction. The 2 LEDs are used to indicate the motor rotation direction; if LED one is ON, that means direction 1 has been chosen, and the same thing for LED two. If both LEDs are OFF, that means the motor has been stopped. The nominal voltage of the motor is 12 volts, as well as the L293D VS input voltage. The L293D VS voltage is always the same as the DC motor voltage, and the L293D Visual Snow Syndrome (VSS) voltage is +5 volts. A pot or pin AN0 is used to change the motor speed. The microcontroller Arduino reads analog data from channel A zero and uses the digital value to set the PWM duty cycle. If the direction 1 switch is pressed, the microcontroller starts PWM1 (pin 11) and stops PWM2 (pin 10), and if the direction 2 switch is pressed, the microcontroller stops PWM1 (pin 11) and starts PWM2 (pin 10). When the 2 switches are OFF, the microcontroller stops PWM1 and PWM2 signals, and the motor will stop.

Proteus ISIS Simulation

Simulation of the proposed circuit using ISIS Proteus for testing. This simulation uses a microcontroller generating PWM signals to control the speed of a motor. The PWM waves generated by the microcontroller can be seen below when 25% of the total resistance is inserted into the circuit, the PWM waves generated by the microcontroller are observed as follows:

Simulation result for 25% of total resistance

Figure 3 shows the output waveform that the PWM waveform encompasses a low-duty cycle. The duty cycle is the fraction of time the PWM signal is high in a cycle, expressed in percentage. 25% resistance means the duty cycle is low, thus the voltage on the motor is for a short duration as shown in Figure 3.

The low-duty cycle thus means less power will be passed to the motor and, therefore, the motor will be run at a much lower speed. As resistance increases, the motor speed continues to decrease, because the average voltage that is applied

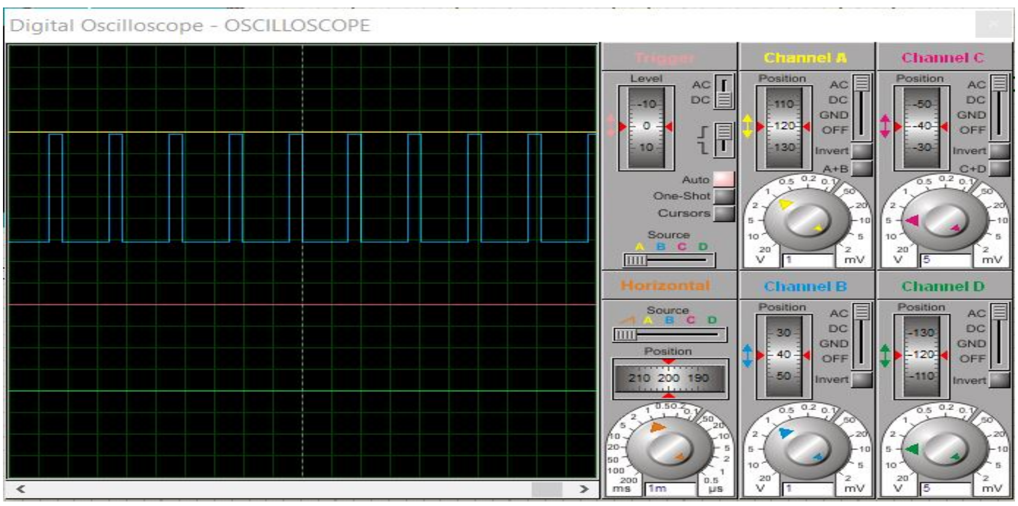

Figure 3. PWM waves generated at 25%.

is lower. This is a consequence of how the speed of the motor is related to the duty cycle, which can be seen in the simulation, where the motor reaches a lower speed with higher resistance as shown in Figure 4.

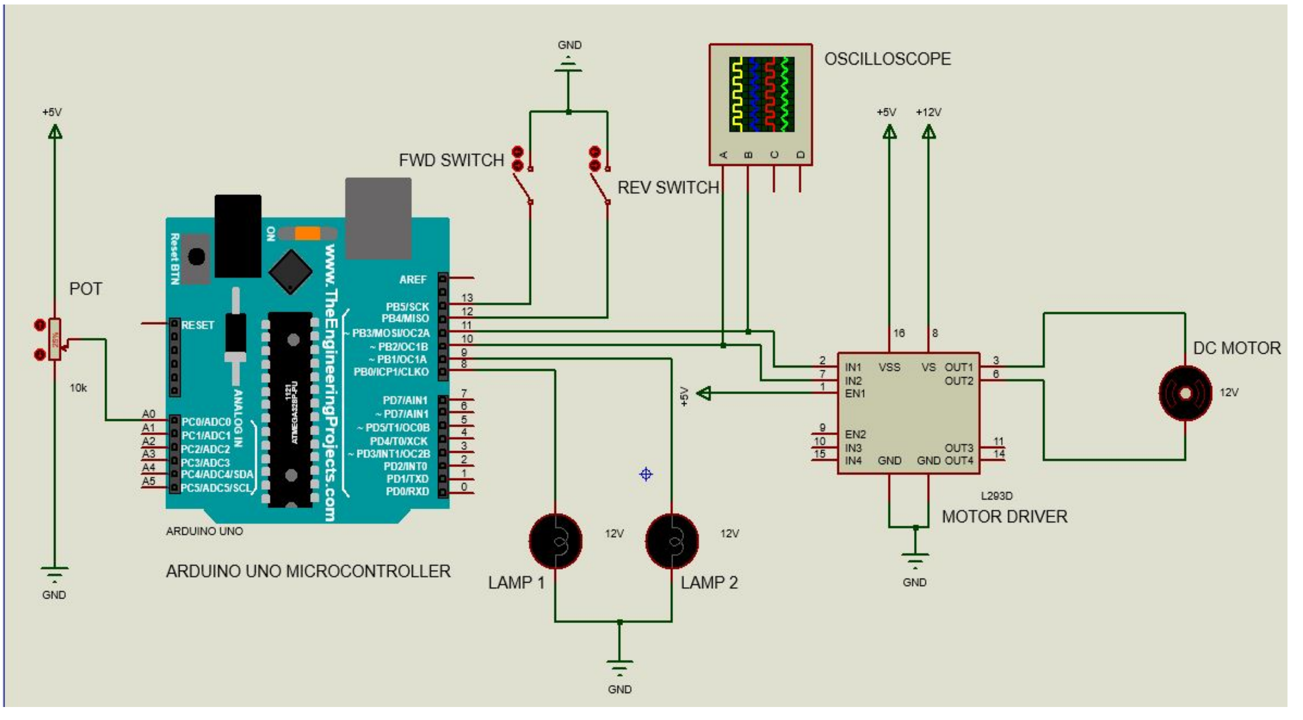

Figure 4. Simulation result for 25% of total resistance.

Simulation result for 50% of total resistance

The proposed circuit is simulated in ISIS Proteus for testing purposes. When 50% of the total resistance is inserted into the circuit, the PWM waves generated by the microcontroller are observed as follows:

The PWM waveform generated at this point shows a duty cycle that is higher compared to the 25% resistance case. However, it is still relatively low compared to higher resistance settings. The duty cycle at 50% resistance causes the motor to receive a moderate amount of average voltage over each PWM cycle. As a result, the motor attains a moderate speed as shown in Figure 5.

It can be seen that the duty cycle is medium as a result the motor attains medium speed as shown from Figure 6, since speed of the motor is directly proportional to the applied voltage. .as shown in Figure 6.

Simulation result for 100% of total resistance

When 100% of the total resistance is inserted in the circuit, the simulation enables us to visualize the effect of varying the resistance on the PWM signal. With the oscilloscope option in the Proteus software, we can control with full accuracy the frequency and duty cycle of PWM signals. There is also the possibility to measure the motor's reaction to these changes, making the connection between the PWM signal and motor speed control easier to comprehend. .as shown in Figure 7.

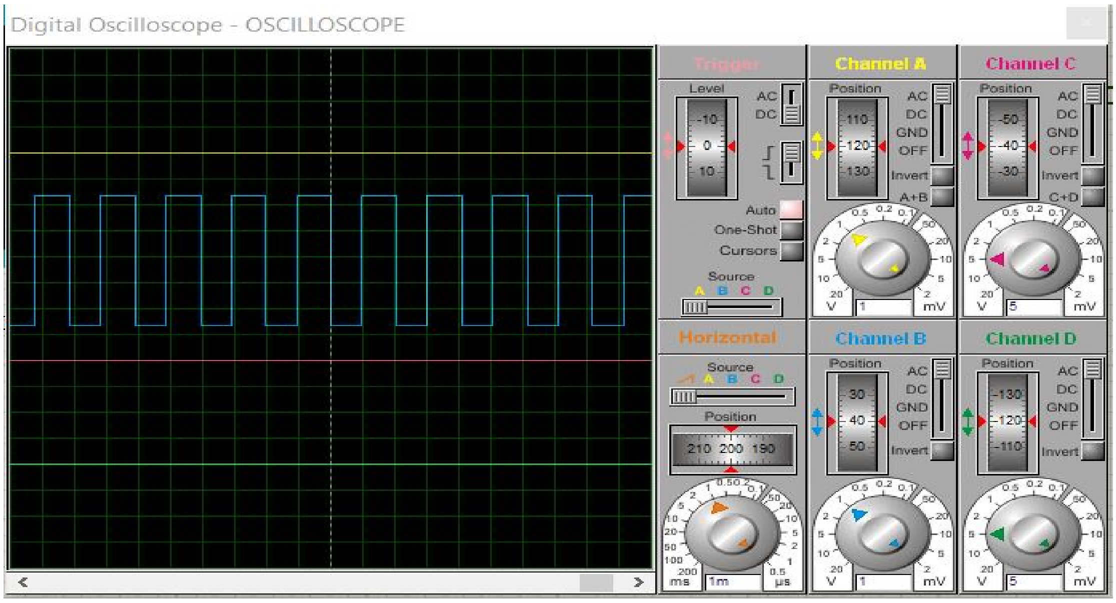

Figure 5. PWM waves generated at 50%.

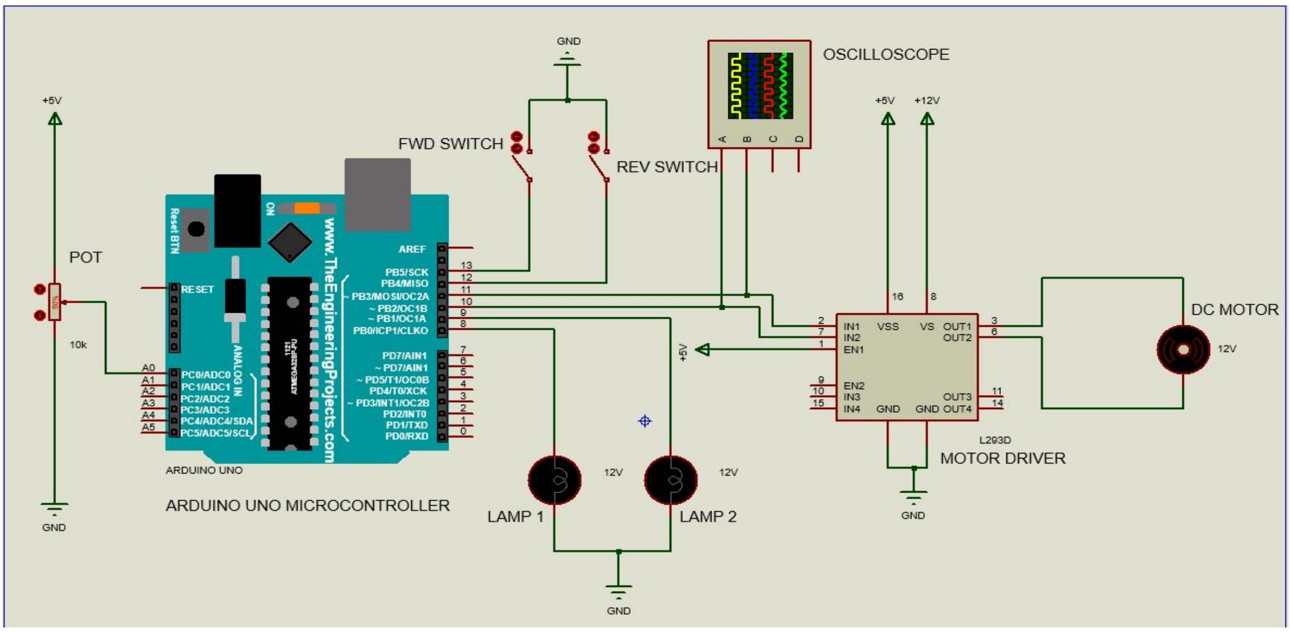

Figure 6. Simulation result for 50% of total resistance.

The simulation of the related circuit using ISIS-Proteus has been done for debugging purposes.

This simulation uses a microcontroller that provides PWM signals to enable controlling the speed of a motor. The PWM waves are a series of the ups and downs of the electrical signal created by the microcontroller. When the total resistance is 100%, the PWM waves produced by the microcontroller will then be in a duty cycle of over 100%, hence a continuous high signal that gives the maximum power to the motor will be the result. Using this type of setting, the operation can be efficient, ensuring that the motor runs at its full speed as shown in Figure 8.

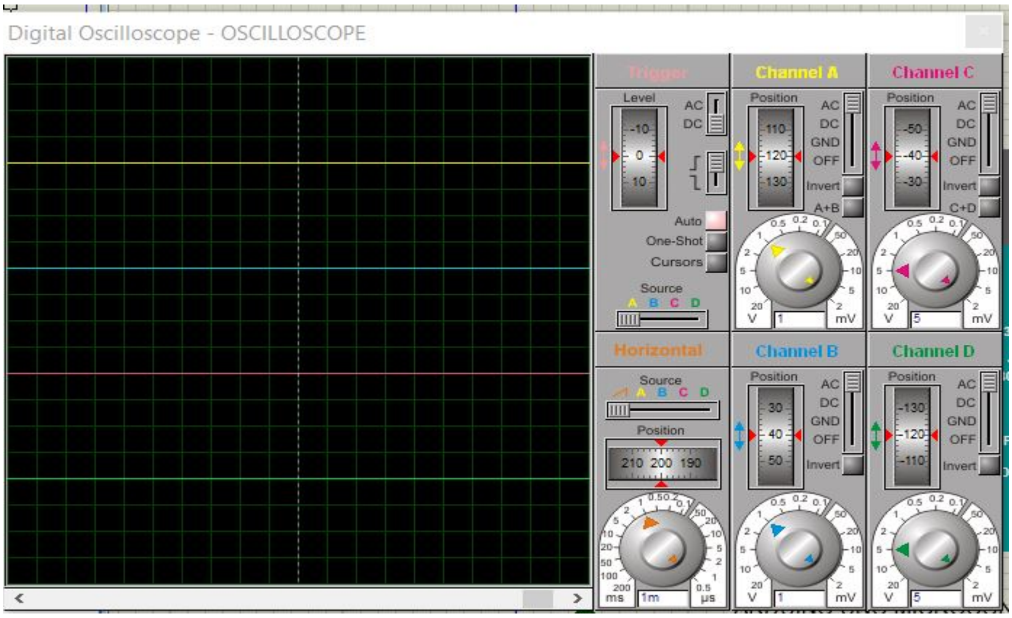

Figure 7. PWM waves generated at 100%.

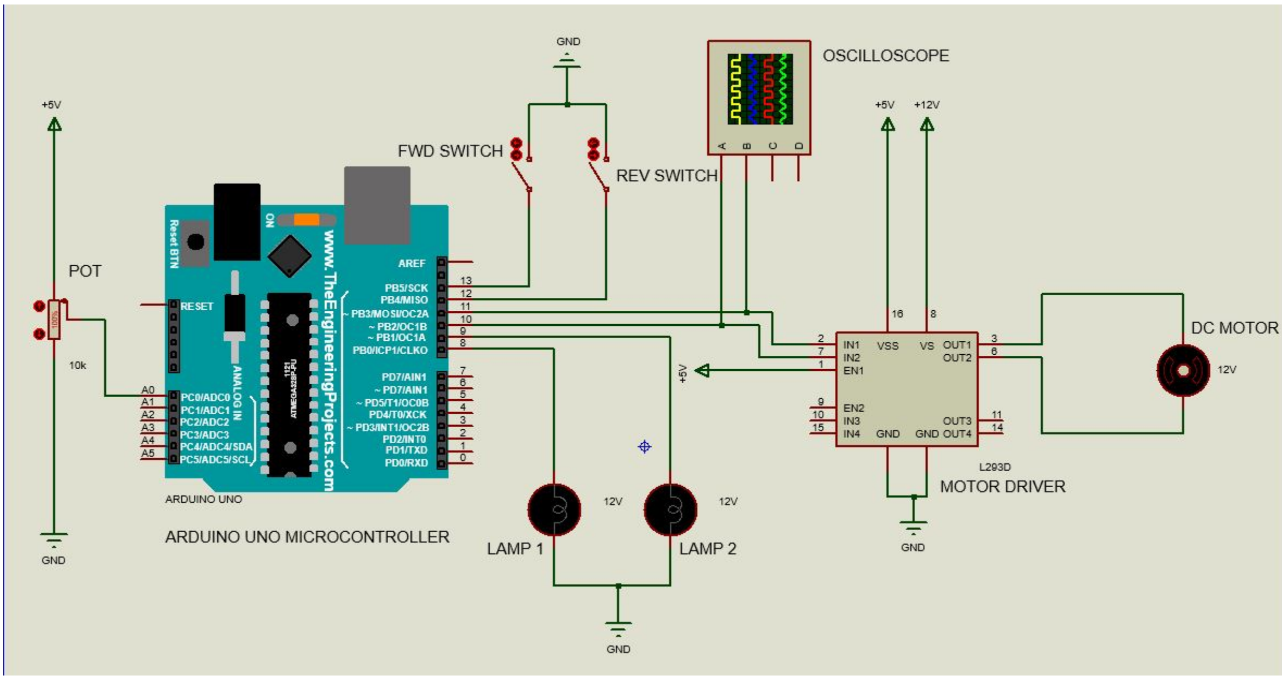

Figure 8. Simulation result for 100% of total resistance.

5.2. Discussions

In general, the feedback loop included in the simulation gives us the opportunity to change the PWM signal generation on the motor performance in real-time. Doing so results in us attaining the most suitable speed control continuously throughout the process. Overall, the simulation provides a robust platform for testing and refining the motor control strategy utilizing PWM signals generated by a microcontroller.

1) Motor speed at 25% Resistance:

- Figure 3 shows the PWM wave has a low-duty cycle, resulting in less voltage being applied to the motor.
- The motor operates at a low speed as shown on Figure 3.

2) Motor speed at 50% Resistance:

- The PWM wave exhibits an equal on-time and off-time, corresponding to a medium duty cycle.
- The motor achieves medium speed since speed is proportional to the applied voltage as shown on Figure 5.

3) Motor speed at 100% Resistance:

- The PWM wave displays a maximum duty cycle with continuous on-time and no off-time.
- Outcome: The motor reaches its maximum speed due to the highest voltage application as shown on Figure 7.

6. Conclusion

In summary, the findings confirm the ability of the Arduino Uno as the controller of the DC motors. From the experiments, using PWM for speed control and an H-bridge configuration to manage the direction of the motors worked as intended to provide controlled and accurate operation of the motor. This suggests that microcontroller technology can be beneficial in future automation and robotics and further indicates the potential for possible uses in the industry. The project contains a simple method of changing the components and programming in order to adapt the system to work in an appropriate way, which offers a significant area of development in motor control systems.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Ganesh, C.V., Suryaprakash Reddy, G., Rajashree, and Kumar, G.A. (2021) DC Motor Speed and Direction Control Using Android. *International Journal of Creative Research Thoughts* (*IJCRT*), **9**, 48-50.

[2] Gaikwad Ankita, M.D.D. (2023) DC Motor Controller Using Arduino. *International Journal of Creative Research Thoughts* (*IJCRT*), **11**, 1-5.

[3] Mahalunge, M. (2021) Speed & Direction Control of DC Motor Using Android Smartphone Application. *International Research Journal of Engineering and Technology*, **8**, 221-223.

[4] Larika, P.P.W.H. (2021) DC Motor Speed Control Through Arduino and L298N Motor Driver Using PID Controller. *International Journal of Electrical Engineering & Emerging Technology*, **4**, 21-24.

[5] Abin Shaju, A. (2017) Voice Integrated Speed and Direction Control for DC Motor. *International Journal of Advanced Research* (*IJAR*), **6**, 1925-1930.

[6] Mane Devanand, B., Koli Rohan, B., Bhosale Sushilraje, S. and Jogdand Vishakha, V.

(2017) Arduino Based Speed Control of DC Motor by Using LabVIEW. *International Journal of Innovative Research in Science*, **6**, 8998-9001.

[7] Vutturi, M. (2022) Automatic Speed Detection and Reporting System Using Arduino. *International Journal of Scientific Engineering and Research* (*IJSER*), **8**, 1-4. https://doi.org/10.70729/14012002

[8] Divya, P.J. (2024) DC Motor Speed and Direction Control. *International Journal of Innovative Research in Electrical*, *Electronics*, *Instrumentation and Control Engineering*, **12**, 133-138.

[9] Khade, P. (2021) Speed and Direction Control of DC Moter with the Help of Arduino. *International Journal for Science and Advance Research in Technology* (*IJSART*), **7**, 426-432.

[10] Mahajan, P.P. (2017) Speed Control of DC Motor Using. *International Journal of Advance Research and Innovative Ideas in Education* (*IJARIIE*), **3**, 1572-1575.

[11] Gajbhiye, T., Chimurkar, P. and Likhar, Y. (2019) A Speed Controlling, Monitoring and Protection of Dc Motor Using Internet of Things (IOT). *International Conference on Management Practices*, *Innovations & Research*, 25-29.

[12] Chauhan, P.E.A. (2020) Speed and Direction Control of DC Motor for Robotics Applications. *International Journal of Robotics & Automation*.

[13] Zhou, Y.E.A. (2019) Integrated Speed and Direction Control of DC Motor Using Arduino. *Journal of Electrical Engineering and Electronics*.