



DC Motor Controller Using Arduino

Gaikwad Ankita K¹, Madke Devendra D.², Gavhane Aarti B.³, Vidhate Kalpana D.⁴

B. E Student, Department of Electrical Engineering, Vithalrao Vikhe Patil College of Engineering, A.nagar, India^{1,2,3}

Assistant Professor Department of Electrical Engineering, Vithalrao Vikhe Patil College of Engineering, A.nagar, India⁴

ABSTRACT

This abstract presents a DC motor speed controller and temperature sensor system developed using Arduino, offering a comprehensive solution for motor control and temperature monitoring. The system utilizes the flexibility and capabilities of the Arduino platform, enabling seamless integration and customization. The DC motor speed controller employs pulse width modulation (PWM) techniques to regulate the speed of the motor, allowing for precise control over its rotational speed. The controller offers a wide range of speed adjustments and incorporates bidirectional control, providing versatility for various applications. In addition, the system incorporates a temperature sensor that accurately measures the ambient temperature. The temperature data is collected and processed by the Arduino, enabling real-time monitoring and analysis. Users can set temperature thresholds and receive alerts or trigger actions based on specific temperature conditions. The Arduino-based system features a user-friendly interface, allowing users to easily configure motor speed settings, temperature thresholds, and notification preferences. The system is suitable for applications where motor speed control and temperature monitoring are essential, such as cooling systems, environmental control, and industrial automation. Overall, this Arduino-based DC motor speed controller and temperature sensor system offer a comprehensive and versatile solution, empowering users to effectively control motor speed and monitor temperature with ease and precision.

Keywords: Arduino, PWM, L293D

I. INTRODUCTION

In the field of electronics and control systems, it is often required to control the speed of a DC motor with precision. One of the popular ways to achieve this is by using an Arduino Uno board and PWM (Pulse Width Modulation) signal to control the motor speed. This report describes the development of a system that utilizes an Arduino Uno board, an L293D IC, and LM35 temperature sensor to control the speed of a DC motor and monitor the temperature of the system. The DC motor speed in general is directly proportional to the supply voltage, so if reduce the voltage from 9 volts to 4.5 volts then our speed become half of what it originally had. But in practice, for changing the speed of a dc motor we cannot go on changing the supply voltage all the time. Pulse width modulation speed control works by driving the motor with a series of "ON-OFF" pulses and varying the duty cycle, the fraction of time that the output voltage is "ON" compared to when it is "OFF", of the pulses while keeping the frequency constant. The speed controller PWM for a DC motor works by varying the average voltage supplied to the motor. PWM signal is essentially a high frequency square wave (typically greater than 1KHz). The Duty Cycle of this square wave is varied in order to vary the power supplied to the load. The input signals we given to PWM controller might be an analog or digital signal according to the design of the PWM controller. The PWM controller accepts the control signal and adjusts the duty cycle of the PWM signal according to the requirements.

II. LITERATURE REVIEW

The PID controller is generated by an Arduino program and used in the speed control of DC motor. The objective of this paper is to control the speed of the DC motor using Arduino. Arduino board plays the role of low-cost data acquisition board. LM393 is a sensor which measures the revolutions.[1]

Android apps provide a seamless interface for control, monitoring and visualization. Bluetooth terminal application and Arduino compiler was used to operate and control the switching action of the D.C motor of 5V. It was observed that we are able to control the D.C motor from a range of Bluetooth device within the range of ten meters.[2]

In this paper, the speed control of Permanent Magnet DC (PMDC) motor is done using Lab VIEW interfaced with Arduino. The main advantage of using Lab VIEW with Arduino is the cost and simple in structure. The other speed control methods like FPGA method, fuzzy control, using 555 timer and PID controllers are having the drawback of complicated design involved, unreliable control, difficult in the online monitoring with high-speed motors. These draw backs motivated us to work on Lab VIEW with Arduino.[3]

In this paper we describe a technical system for DC motor speed control. The speed of DC motor is controlled using Arduino

programming platform and MATLAB's Simulink coder. This paper contains introduction to using an Arduino board and Simulink PI controller in closed loop system. It will be described how to program Arduino with Simulink coder and in the end, we present the results of PI controller for DC motor speed will be given.[4]

Arduino represents the receiver side interface 8 channel relay module with 8-bit RF receiver. It is isolated from transmitter side. Figure 8 represents the DC SSR interface with DC Geared motor. Arduino Uno microcontroller board is interfaced with the RF transmitter side. Console is connected with personal computer through USB cable. DC SSR is placed between H-Bridge and DC source. It is operated by PWM signals from Arduino Uno board.[5]

III. METHODOLOGY

A DC motor is a machine that converts electrical energy into mechanical energy. The operation is based on simple electromagnetism, i.e. "when a current carrying conductor is placed in a external magnetic field it experiences a force which is proportional to the current and the external magnetic field." A torque is generated by the magnetic reaction and the armature revolves and this induces a voltage in the armature windings which is opposite in direction to the outside voltage applied to the armature, when current is passed through the armature of the DC motor, and hence is called back voltage or counter e.m.f. The back voltage rises till it becomes equal to the applied voltage as the motor rotates faster, the speed at which the DC motor rotates depends on two factors - the armature current as well as the strength of the magnetic field acting on the armature. The code for the system is written in Arduino IDE. The code reads the temperature from the LM35 sensor and displays it on the I2C display. It also reads the state of the two push buttons and adjusts the PWM signal to control the motor speed accordingly. If the temperature goes above 45 degrees Celsius, the code activates the relay to turn on the fan. The DC motor speed control using PWM (Pulse Width Modulation) through Arduino Uno and L293D driver IC involves the following steps:

- Connect the Arduino Uno to the computer using a USB cable and open the Arduino IDE.
- Design the circuit by connecting the necessary components, including Arduino Uno, L293D driver IC, DC motor, LM35 temperature sensor, and a 5V SPDT relay.
- Upload the required program to the Arduino Uno to control the speed of the DC motor based on the PWM signal.
- Set the desired speed using the program and observe the response of the DC motor.

IV. BLOCK DIAGRAM

A block diagram of DC motor controller typically consists of several key components interconnected to achieve motor control functionality.

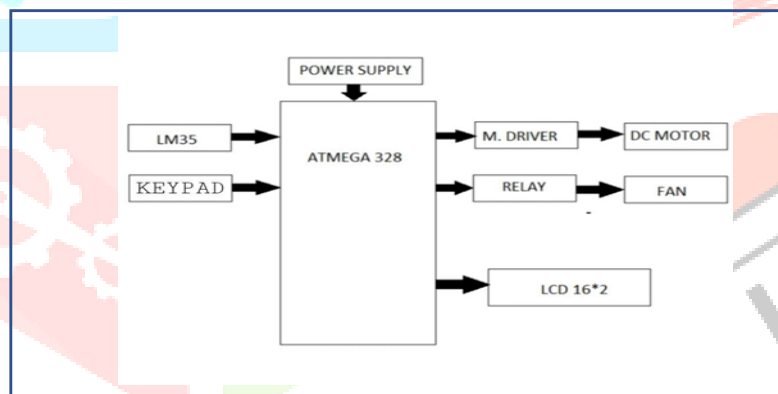


Fig: Block Diagram of System

Overall, the block diagram illustrates the integration of the DC motor, Arduino board, PWM signal generation, and potentially sensors to create a system that enables precise control of the motor's speed and direction using the Arduino platform.

V. HARDWARE DESCRIPTION

A. Arduino Uno:

Arduino Uno is a widely used open-source microcontroller board that is based on the ATmega328P chip. It is one of the most popular choices for beginners and experienced enthusiasts alike. The Arduino Uno board offers a range of features that make it ideal for a variety of projects. The board features 14 digital input/output pins, of which 6 can be used as pulse width modulation (PWM) outputs, allowing for precise control of connected devices such as motors or LEDs. It also provides 6 analog input pins, enabling the measurement of sensors or other analog signals.

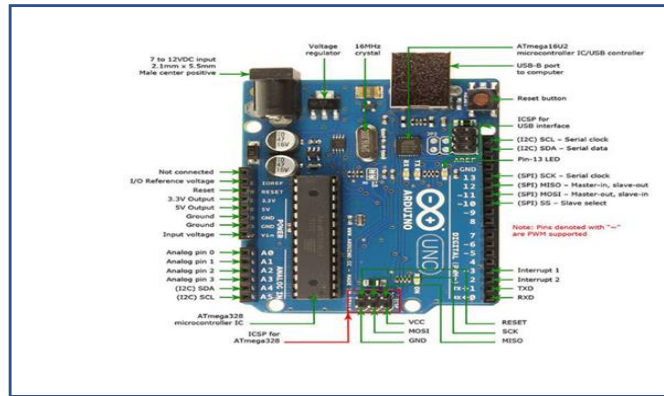


Fig: Arduino Uno

Arduino Uno is equipped with a USB interface, allowing for easy connection to a computer for programming and power supply. It supports the Arduino Integrated Development Environment (IDE), a user-friendly software for writing and uploading code to the board. With a clock speed of 16 MHz and 32KB of flash memory, Arduino Uno offers sufficient processing power for most applications. It can be powered either through the USB connection or an external power supply. The Arduino Uno board has a large community of users, which means there are extensive resources, tutorials, and libraries available online. Overall, Arduino Uno is a versatile and accessible microcontroller board, making it an excellent choice for prototyping, educational projects, and DIY electronics. Its simplicity, wide range of features, and extensive community support have made it a go-to platform for makers, students, and hobbyists worldwide.

B. LM35 Sensor:

The LM35 is a precision analog temperature sensor widely used in various applications. It provides an accurate output voltage proportional to the temperature in Celsius. The sensor has a linear temperature range of -55°C to 150°C , making it suitable for a wide range of temperature measurement needs. It requires a single power supply and provides a low self-heating feature, minimizing any temperature interference caused by the sensor itself.



Fig: LM35 Sensor

The LM35 has a simple analog output that can be easily interfaced with microcontrollers, such as Arduino, for temperature monitoring and control. Its small size, high accuracy, and straightforward operation make it a popular choice for temperature sensing in electronic projects, industrial systems, and environmental monitoring.

C. L293D Driver IC:

The L293D is a popular motor driver IC that allows bidirectional control of small to medium-sized DC motors. It is designed to handle moderate currents and is commonly used in robotics and automation projects.

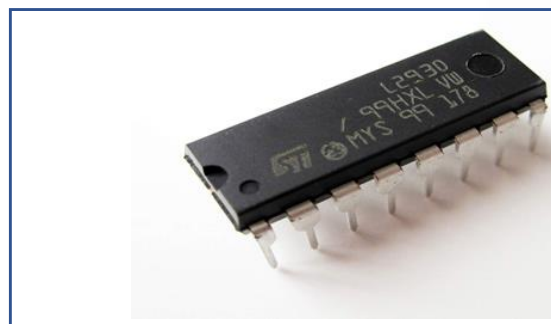


Fig: L293D Driver IC

Motor Channels: The L293D has two independent motor channels, labeled as Motor 1 and Motor 2. Each channel can control the direction and speed of a DC motor.

H-Bridge Configuration: Each motor channel of the L293D consists of an H-bridge configuration. The H-bridge allows the motor to be driven in both forward and reverse directions.

Maximum Voltage: The L293D can handle a wide range of supply voltages, typically from 4.5V to 36V. It is important to ensure that the motor's voltage matches the IC's voltage rating.

VI. SOFTWARE DESCRIPTION:

The Arduino IDE (Integrated Development Environment) is a software tool used for programming and uploading code to Arduino boards. It provides a user-friendly interface that simplifies the process of writing, compiling, and uploading code for Arduino projects. The IDE supports the Arduino programming language, which is based on C/C++. It offers a wide range of built-in libraries and functions that make it easier to work with various hardware components and sensors. Additionally, the Arduino IDE allows users to manage libraries, monitor serial data, and access a vast community of resources and examples. Its intuitive interface and extensive features make it an essential tool for both beginners and experienced users in the Arduino ecosystem.

VII. EXPERIMENTAL SETUP

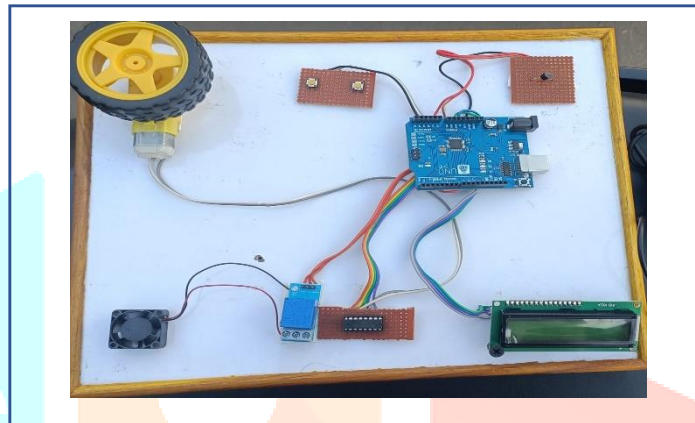


Fig: Proposed Experimental Setup

VIII. CALCULATIONS

Determine the required PWM frequency:

Let's assume a desired PWM frequency of 500 Hz. Calculate the required period (T) for the PWM signal:

$$T = 1/\text{frequency}$$

$$T = 1/500 \text{ Hz}$$

$$T = 0.002 \text{ seconds or } 2 \text{ milliseconds (ms)}$$

Determine the resolution (bit depth) of the PWM signal:

Arduino Uno has a 8-bit PWM resolution, which means it can generate 256 different duty cycles (0 to 255).

Determine the desired duty cycle (DC) for the motor speed:

Let's assume a desired duty cycle of 75%.

Calculate the ON time for the PWM signal:

$$\text{ON time} = (\text{DC}/100) * T$$

$$\text{ON time} = (75/100) * 2 \text{ ms}$$

$$\text{ON time} = 1.5 \text{ ms}$$

Determine the OFF time for the PWM signal:

$$\text{OFF time} = T - \text{ON time}$$

$$\text{OFF time} = 2 \text{ ms} - 1.5 \text{ ms}$$

$$\text{OFF time} = 0.5 \text{ ms}$$

Connect the L293D motor driver to Arduino Uno:

Connect the control pins (IN1, IN2, IN3, IN4) of L293D to digital output pins of Arduino Uno.

Connect the motor to the outputs (OUT1, OUT2) of L293D.

Write the Arduino code for the PWM control using analog Write():

Set the digital pin for PWM output: `pinMode(pin, OUTPUT)`

Set the duty cycle using analog Write(`pin, dutyCycle`)

For example, if the motor control pins are connected to Arduino digital pins 9 and 10:

`analogWrite(9, 191); // 75% duty cycle (75% of 255)`

`analogWrite(10, 191); // same duty cycle for the other motor terminal.`

Upload and run the Arduino code to control the motor speed using PWM.

IX. ADVANTAGES

- i. **Precise Motor Control:** The DC motor speed controller enables precise control over the rotational speed of the motor, allowing for accurate adjustments to meet specific application requirements.
- ii. **Versatility:** The bidirectional control feature of the DC motor speed controller provides versatility, enabling seamless control over motor rotation in both directions.
- iii. **Real-time Temperature Monitoring:** The temperature sensor integrated with Arduino facilitates real-time temperature monitoring, ensuring timely intervention in critical scenarios.
- iv. **Customization and Expandability:** The Arduino platform allows for easy customization and integration of additional sensors, modules, or displays, expanding the functionality of the system.
- v. **User-Friendly Interface:** The system provides a user-friendly interface for easy configuration of motor speed settings, temperature thresholds, and notification preferences.
- vi. **Cost-Effective Solution:** The Arduino-based system offers a cost-effective solution compared to proprietary motor controllers and temperature monitoring systems.

X. CONCLUSION

The development of DC motor controller using Arduino has proven to be a valuable and versatile solution for a precise motor control in a wide range of applications. Through an extensive literature review, it is evident that significant progress has been made in the field of DC motor control using Arduino, with advancements in control algorithms, hardware integration, and system optimization. The systematic approach ensures that the system requirements are carefully analyzed, appropriate motor and driver selections are made, and the circuit design is optimized for reliable operation. Arduino programming facilitates the implementation of control algorithms, making it easier to achieve precise speed and direction control.

The future scope of work presented in this study highlights exciting directions for further advancements in DC motor control using Arduino. Exploring advanced control algorithms, wireless communication integration, and enhanced sensing techniques opens up new possibilities for more control systems. Additionally, focusing on efficiency optimization, system integration and real time monitoring and analytics can lead to more energy-efficient, compact, and data-driven motor control solutions.

XI. REFERENCES:

- [1] "SPEED CONTROL OF DC MOTOR USING ARDUINO" Prof.P.P.Mahajan, Rounak Shetye, Pranav Vedpathak, Shubham Naik. Vol-3 Issue-1 2017 IJARIE-ISSN(O)-2395-4396
- [2] "SOLAR -POWERED ANDROID-BASED SPEED CONTROL OF DC MOTOR VIA SECURE BLUETOOTH", Abhishek Khanna Priya Ranjan. 2015 Fifth International Conference on Communication Systems and Network Technologies
- [3] Speed Control of Permanent Magnet (PM)DC Motor Using Arduino and LabVIEW Angalaeswari. S Divyanshu Kumar, Amit Kumar, Shubham Bhadoriya. School of Electrical Engineering VIT University, Chennai Campus Tamilnadu, India.
- [4] "PI controller for DC motor speed realized with Arduino and SimulinkMario Gavran*, Mato Fruk** and Goran Vujisić", MIPRO 2017, May 22- 26, 2017, Opatija, Croatia
- [5] "ARDUINO BASED DC MOTOR SPEED CONTROL". Hirdesh Kr. Saini ,Shahid Firoz Ashutosh Pande. IJRDO-Journal of Electrical and Electronics Engineering ISSN: 2456-6055