International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 06 | June – 2025 DOI: 10.55041/ISJEM04243
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Automatic Temperature Controlled Fan using Arduino

Konanki Toneshwar Naidu[1*], Gali Kalpana[1], Afrin[1], Komati Rakesh[1], B. Santosh Kumar[1]

[1]*Siddhartha Institute of Technology & Sciences, Narapally, Korremula Road, Ghatkesar, Medchal- Malkajgiri (Dist)-500 088, India.*

* 23TQ1A0460@siddhartha.co.in, 23TQ1A0431@siddhartha.co.in , 23TQ1A0429@siddhartha.co.in, 23TQ1A0459@siddhartha.co.in

---

Abstract -In the current scenario, electricity availability has reached a critical stage, highlighting the need for efficient energy conservation. As the popular saying goes, "One unit saved is one unit generated." This project introduces a standalone automatic fan speed controller that adjusts the fan speed based on ambient temperature, activating only when the temperature exceeds a predefined threshold. The system employs embedded technology to create an efficient and reliable closed-loop feedback control mechanism. An Arduino microcontroller forms the core of the system, enabling fast and dynamic control. The circuit is compact and constructed using minimal components, making it suitable for integration into various applications such as air conditioners, water heaters, snow melters, ovens, heat exchangers, mixers, furnaces, incubators, thermal baths, and veterinary operating tables. The LM35 temperature sensor detects the surrounding temperature and converts it into an electrical signal, which is sent to the Arduino. Based on this input, the microcontroller controls a transistor that regulates the fan speed accordingly. The system operates using a regulated 9V, 1A power supply. This project is particularly useful in industrial settings for monitoring and controlling the temperature of boilers and other heat-sensitive equipment.

Key Words: Arduino, LM35, PWM, Temperature, Cooling

1. INTRODUCTION

With rapid advancements in technology, intelligent systems are increasingly integrated into everyday life. Microcontrollers, which are compact single-chip microprocessors designed for control and automation, play a key role in this transformation. They are widely used across various fields to perform automated tasks with high precision. Today, microcontrollers are found in devices such as air conditioners, power tools, toys, office equipment, and more.

A typical microcontroller integrates several essential components into a single chip, including a Central Processing Unit (CPU), timers, counters, interrupt controllers, memory, input/output ports, and Analog-to-Digital Converters (ADC). This integration results in reduced control board size and lower power consumption.

This paper presents the design and simulation of a fan speed control system based on room temperature, implemented using Pulse Width Modulation (PWM) with a microcontroller. A temperature sensor is used to monitor the ambient temperature. Based on the sensor input, the system adjusts the fan speed by varying the PWM duty cycle from 0% to 100%. The corresponding room temperature and fan speed are displayed on a Liquid Crystal Display (LCD).

Pulse Width Modulation (PWM) is a technique used to simulate analog output using digital signals. A digital controller generates a square wave—an on/off signal—that toggles between high (5V) and low (0V) states. By adjusting the ratio of the "on" time to the total period (known as the duty cycle), the system simulates different voltage levels.

For instance, when this technique is applied to an LED, varying the duty cycle changes the brightness, appearing as a continuous analog output. The same principle is used here to control the fan speed. The duration of the "on" time is called the pulse width, and modulating this width enables control of analog-like behavior using digital signals. The frequency of the PWM signal is the inverse of the period, and when the modulation is fast enough, it provides smooth control over devices such as fan

2. Need of Arduino

The beauty of microcontrollers is that, we get very precise control over the peripherals which are connected to it. In this project the user just need to input the threshold temperature in the program, the microcontroller will take care of rest of the function. There are tons of non- microcontroller based automatic temperature controller projects available around the internet, such as using comparator and transistors. They are very simple and they do work well but, the problem arises while calibrating the threshold level using preset resistor or potentiometer. We have a blind idea while calibrating it and the user may need to do trial and error method to find the sweet spot. These problems are overcome by microcontrollers, the user just need to enter the temperature in Celsius in this project, so no need for calibration.

This project can be used where internal temperature of circuit need to be stabilized or saving it from overheating. When the room temperature reaches the threshold temperature the fan turns on and turns off when the room cools down. So, it's basically an automated process.

System Configuration:

It is important to stress out the fact that heat is a normal by product of an electronic component or an entire electronic assembly that is under operation. This is the reason why electronic components are designed and built to withstand specific and certain levels of heat. However, excessive levels of heat results in overheating than in turn, result in damages to an electronic component. One common but specific effect is material degradation. . The changes in the physical and chemical properties affect the performance or in other words, the operation and function of an electronic component. There are technical and non-technical, as well as direct and indirect ways for preventing overheating. One of the simplest ways to reduce heat in the central electrode of a

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 06 | June – 2025 DOI: 10.55041/ISJEM04243
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

SET is by increasing its area and thickness. Doing so would increase the heat flowing from electron gas to phonons. It would also reduce thermal boundary resistance, thus reducing the resistance of the central electrode to thermal flow and allowing it to become resistant to too much heat to a certain extent. Another solution is the application of different enabling technologies for efficient cooling.

3. PROPOSED METHOD

The proposed system aims to design and implement an automatic fan speed controller that adjusts the speed of a DC fan based on real-time ambient temperature readings. This system utilizes an Arduino Uno microcontroller as the central processing unit, a DHT11 sensor for temperature measurement, an L298N motor driver to control the fan, and an I2C 16x2 LCD display to show live temperature and fan speed status.

In this system, the DHT11 sensor continuously monitors the surrounding temperature and sends the data to the Arduino. Based on predefined temperature thresholds, the Arduino processes this data and generates a corresponding PWM (Pulse Width Modulation) signal. This PWM signal is sent to the motor driver, which controls the speed of the DC fan proportionally—ensuring efficient cooling, reduced energy consumption, and quieter operation.

The I2C LCD display provides user-friendly visual output, showing both the current temperature and the fan's operating speed as a percentage. This allows users to monitor system performance in real time.

The key features of the proposed system include:

- Automatic, real-time control of fan speed based on ambient temperature.
- Smooth PWM-based speed regulation instead of simple ON/OFF control.
- User-friendly display showing temperature and fan speed.
- Low-cost, energy-efficient solution using easily available electronic components.

This system can be applied in a variety of environments such as homes, offices, electronics cooling units, and greenhouses, where maintaining a stable temperature is important. The modularity and simplicity of this system also make it ideal for learning and prototyping in the field of embedded systems and IoT.

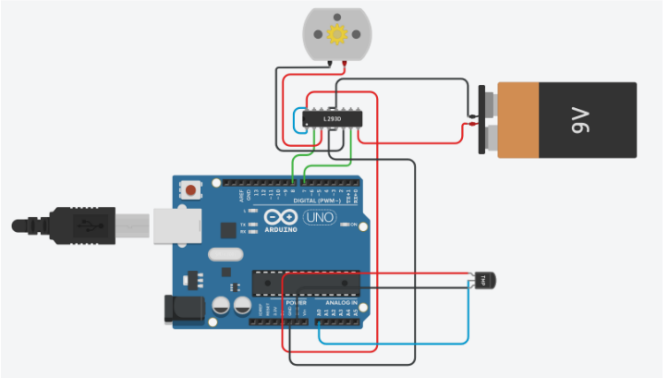

Fig.1. Connection for temperature controlled fan using Arduino

System Components:

Arduino Uno

Function: Acts as the main controller of the system.

Role: Processes temperature data from the sensor, generates PWM signals to control fan speed, and manages the LCD display.

Features: Open-source, based on the ATmega328P microcontroller, with digital I/O pins and PWM support.

DHT11 Temperature and Humidity Sensor

Function: Measures the surrounding air temperature.

Role: Sends real-time temperature data to the Arduino.

Features: Digital output, temperature range 0–50°C, basic accuracy suitable for general applications.

L298N Motor Driver Module

Function: Drives the DC fan by receiving PWM signals from the Arduino.

Role: Controls the speed and direction of the DC motor (only one direction is used in this project).

Features: Dual H-Bridge driver capable of handling high current and voltage for motors.

DC Fan

Function: Provides airflow for cooling.

Role: Operates at variable speeds based on temperature data to maintain a desired environment.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 06 | June – 2025 DOI: 10.55041/ISJEM04243
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Features: Operates on 5V or 12V depending on the model; speed is controlled using PWM via the motor driver.

I2C 16x2 LCD Display

Function: Displays temperature readings and fan speed percentage.

Role: Provides a user interface for real-time monitoring.

Features: Uses I2C communication to minimize the number of pins used on the Arduino.

Working of Proposed System:

Power ON the System: The Arduino Uno and all connected components (DHT11, motor driver, LCD) are powered up.

Initialize Components: The Arduino initializes communication with the DHT11 sensor and I2C LCD.PWM and control pins are set as output.

Read Temperature Data: The Arduino reads temperature (and optionally humidity) data from the DHT11 sensor.

Display Temperature: The current temperature is shown on the I2C LCD display for user monitoring.

Compare Temperature to Thresholds: The Arduino compares the temperature value to predefined thresholds (e.g., 25°C, 30°C, 35°C).

Determine Fan Speed: Based on the temperature range:

Below 25°C: Fan is turned OFF

25°C – 29°C: Fan speed is set to LOW (PWM ~ 100)

30°C – 34°C: Fan speed is set to MEDIUM (PWM ~ 180)

35°C and above: Fan speed is set to HIGH (PWM = 255)

Send PWM Signal: Arduino sends the corresponding PWM signal to the motor driver's ENA pin.

Control DC Fan: The motor driver (L298N) regulates the DC fan speed based on the PWM input.

Display Fan Speed Status: The fan speed level (e.g., LOW, MEDIUM, HIGH or percentage) is displayed on the LCD.

Repeat the Cycle: The process repeats every few seconds (as defined in the loop), constantly adjusting fan speed based on temperature changes.

4. RESULT

Arduino based temperature-controlled fan is implemented. Thus, here fan speed has been controlled by using Pulse Width Modulation and Arduino board according to the temperature sensed by the help of Temperature. PWM technique is found to be the best technique for controlling the fan speed using the sensed temperature. The system is working properly. The speed of fan depends on the temperature and there is no need for regulating the fan speed manually again and again.

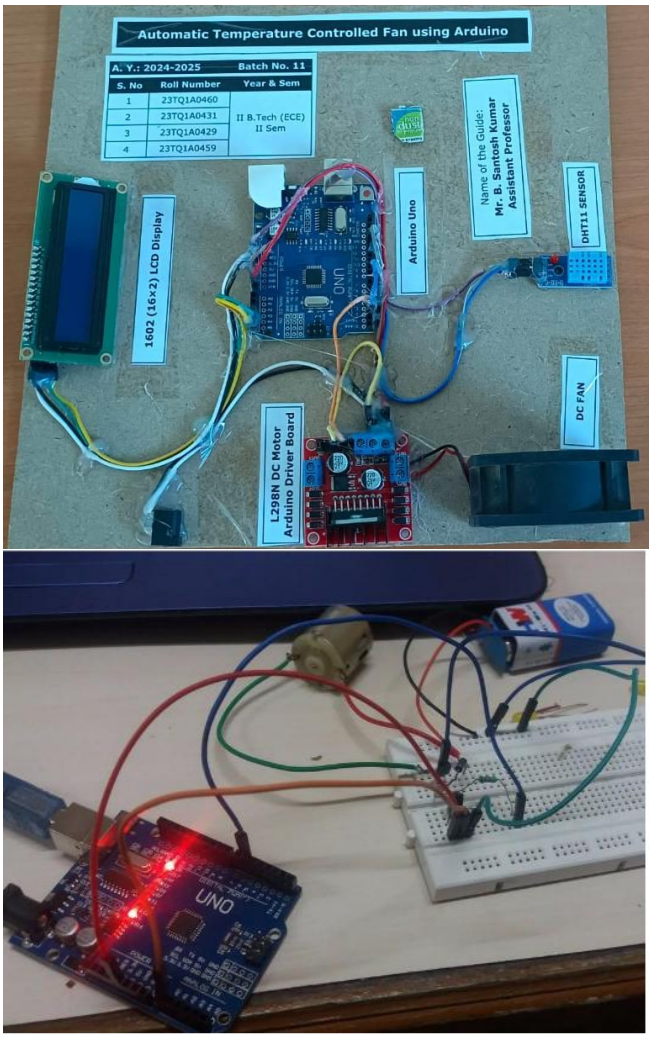

Fig.2.Temperature Controlled Fan Using Arduino.

5. CONCLUSIONS

The beauty of microcontrollers is that, we get very precise control over the peripherals which are connected to it. In this paper the user just need to input the threshold temperature in the program, the microcontroller will take care of rest of the function.There are tons of non-microcontroller based automatic temperature controller projects available around the internet, such as using comparator and transistors. They are very simple and they do work well but, the problem arises

while calibrating the threshold level using preset resistor or potentiometer. We have a blind idea while calibrating it and the user may need to do trial and error method to find the sweet spot.

These problems are overcome by microcontrollers, the user just need to enter the temperature in Celsius in this project, so no need for calibration. This paper can be used where internal temperature of circuit need to be stabilized or saving it from overheating. We are connecting a CPU fan as output. This setup can be used to control the internal ambient temperature of an enclosed circuit. When the threshold temperature is reached the fan turns on. When the temperature goes below threshold temperature fan turns off. So it's basically an automated process. We connected a relay for controlling devices which runs on mains voltage such as table fan. When the room temperature reaches the threshold temperature the fan turns on and turns off when the room cools down. This may be the best way for saving power and this can be heaven for lazy people who wish others to switch the fan ON when they feel warm.

REFERENCES

1. Official Arduino BT Website – http://www.arduino.cc/cn/guide/ArduinoBT
2. Mazidi, M. A. – The Microcontroller and Embedded Systems.
3. Prasad, D. V. – Electronic Components.
4. Saad, M., Abdoalgader, H., & Mohamed, M. – "Automatic Fan Speed Control System Using Microcontroller."
5. Singh, K., Dhar, M., & Roy, P. – "Automatic Fan Speed Control System Using Arduino," International Journal of Novel Research and Development.
6. Surabhi – "Design & Fabrication of Temperature-Based DC Fan Speed Control System Using Microcontroller & PWM Technique," International Journal of Research in Science, Engineering and Technology (IJRSET), Vol. 4, Issue 7, July 2015.
7. Bhatia, V. & Bhatia, G. – "Room Temperature-Based Fan Speed Control System Using PWM Techniques," International Journal of Computer Applications, Vol. 81, No. 5, November 2013. (ISSN: 0975-8887)
8. Soren, G. S. & Gupta, R. A. – "Temperature Controlled DC Fan Using Microcontroller," National Institute of Technology, Rourkela.

