DEVELOPMENT AND DEPLOYMENT OF A MICROCONTROLLER-BASED ADJUSTABLE TEMPERATURE CONTROLLER

Supported by tetfund

[1]Iliya Tizhe Thuku,[2]Visa M. Ibrahim and [3]Joseph Stephen Soja, [4]Abubakar Abdullahi
[1,2,3,4] Department of Electrical and Electronics Engineering, Modibbo Adama University, Yola Adamawa state

Received: March 1, 2025, Accepted: April 17, 2025

Abstract: This study focuses on the design and implementation of a microcontroller-based variable temperature controller utilizing the ATMEGA328 microcontroller, capable of regulating temperatures between -45°C and +140°C. A temperature sensor integrated circuit measures the ambient temperature, providing an output signal to the microcontroller's analog input, which is then converted into a digital value. The software for the system was developed and compiled using the Arduino Integrated Development Environment (IDE) in C++, and its functionality was tested through simulation in Proteus Computer-Aided Design (CAD) software. Laboratory experiments were conducted to evaluate performance, with measured data indicating an average sensitivity of 9.38mV/°C, closely aligning with the sensor's standard value of 10mV/°C, thereby confirming the controller's accuracy.

Keywords: Microcontroller, Adjustable Temperature Controller, LCD Display, Complex Instruction Set Computing (CISC).

Introduction

Automation emerged as a solution to improve quality, reduce labor and energy consumption, and enhance system accuracy and precision Patel, S.S. et al. (2014). It involves utilizing control systems to operate equipment with minimal human intervention. Temperature control is essential for maintaining a stable environment by measuring and regulating temperature variations (Das, T.K. & Das, Y. 2013). In engineering applications, temperature controllers are designed using heat-sensing devices such as thermistors and integrated circuits to detect temperature changes. (Atayero, A.A. et al 2015). These controllers incorporate switching circuits that activate heating or cooling mechanisms in response to temperature fluctuations. Temperature control systems play a crucial role in industrial production and research laboratories (Kumar, K. & Kurian, C.P. 2014). Traditional temperature control systems rely on mechanical devices such as bimetallic strips and fluid columns. Bimetallic strips function through expansion and contraction in response to temperature changes. However, modern applications, such as microwave drying, require more precise control than conventional methods provide. (Li, Z. 2004). Mechanical devices have limitations: bimetallic strips lose efficiency over time, reducing switching response speed, while fluid columns lack long-term uniformity. Additionally, both methods are difficult to calibrate, lack precision, and incur high costs. To address these challenges, microcontrollers and semiconductor-based devices have been introduced, offering lower costs, greater efficiency, and higher accuracy while overcoming the limitations of mechanical control systems. This research is motivated by the need for a more effective and reliable temperature control system. (Li, Z. 2004)..

Several studies have explored microcontroller-based temperature control systems. For instance, researchers have implemented fuzzy logic algorithms in microcontrollers for temperature regulation in microwave hyperthermia devices. (Santoso, I. et al. 2014). Their study demonstrated stable temperature maintenance during microwave heating. However, without fuzzy logic control, temperature fluctuations could not be managed effectively.

Another study developed a microcontroller-based temperature and lighting control system for smart homes, aiming to enhance security and energy efficiency. (Lwin, J.T. & Ya, A.Z. 2014). The research focused on using a PIC microcontroller to regulate room temperature and lighting in a single-story building with six rooms. Experimental results confirmed the system's ability to optimize energy consumption.

A separate study explored a fuzzy logic-based microcontroller system designed to stabilize temperature amid sudden fluctuations. (Jebelli, A. & Yagoub, M.C. 2015). The research employed an LM35 temperature sensor, which provides an output voltage proportional to the centigrade temperature. The intelligent system incorporated both automatic and manual control mechanisms for heating and cooling functions.

Furthermore, a real-time weather monitoring system was developed to track temperature, atmospheric pressure, relative humidity, and dew point using a GSM network. (Gouda, K., Preetham, V., & Swamy, M.S. (n.d.) 2025). The transmitter, programmed in C, processed and transmitted data to the receiver.

Another research effort implemented temperature control using a PIC16F877A microcontroller interfaced with an LM35DZ temperature sensor, a seven-segment LED display, and a buzzer. (Auwala, et al. 2017). The system, developed through hardware and software integration, included power supply, microcontroller, temperature sensor, display, and alarm units. The software was written in assembly language, and the system's accuracy was validated against a standard mercury thermometer, yielding a Mean Absolute Percentage Deviation (MAPD) of 4.69%.

Similarly, another study employed an LM35 temperature sensor for precise temperature monitoring and control. The microcontroller compared the desired temperature with real-time readings to activate heating or cooling devices (Amin, M.R., Ghosh, A., & Hadi, A. 2018). The system utilized fixed-voltage regulators (7412 and 7405) at 5V, with C++ used for microcontroller programming. Validation was conducted using LM35 and an industrial thermometer, and results demonstrated the ability to maintain temperature between 39°C and 41°C.

The goal of this research is to design and implement a microcontroller-based temperature controller using the ATMEGA328 microcontroller with an integrated Liquid Crystal Display (LCD) to show real-time temperature readings. The system allows users to adjust the temperature range using push buttons, with values stored in the Electrically Erasable Programmable Read-Only Memory (EEPROM) of the ATMEGA328, ensuring data retention after power loss. The selection of components is based on their affordability, availability, reliability, and low power consumption.

Component Description

This section covers the design aspects of the microcontroller.

ATMEGA328P Microcontroller Architecture

The ATMEGA328P is a fully integrated computer on a chip, incorporating a processor, registers, and both program and data memory as shown in figure 1. It is an 8-bit, low-power CMOS microcontroller based on the AVR enhanced RISC architecture. Unlike the traditional Von Neumann architecture, which follows the Complex Instruction Set Computer (CISC) model and uses a shared bus for data and addresses, the ATMEGA328P employs the Harvard architecture, where the data bus and address bus are separate. The AVR core features an extensive instruction set and 32 general-purpose working registers. These registers are directly linked to the Arithmetic Logic Unit (ALU), enabling simultaneous access to two independent registers within a single clock cycle. This design improves code efficiency and delivers processing speeds up to ten times faster than conventional CISC microcontrollers. (Corporation, A. 2009)

Key features of the ATMEGA328P include:

a. ***Memory:*** 32K bytes of In-System Programmable Flash with Read-While-Write capability, 1K bytes EEPROM, and 2K bytes SRAM.
b. ***I/O and Registers***: 23 general-purpose input/output (I/O) lines and 32 working registers.
c. ***Timers and Counters***: Three flexible Timer/Counters with compare modes.
d. ***Communication Interfaces***: A serial programmable USART, a byte-oriented 2-wire Serial Interface, and an SPI serial port.
e. ***Analog Features***: A 6-channel 10-bit ADC (or 8 channels in TQFP and QFN/MLF packages).
f. ***Power Management***: A programmable Watchdog Timer with an internal oscillator and five software-selectable power-saving modes.

In Idle mode, the CPU halts while key components such as SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and the interrupt system remain operational, ensuring efficient power consumption. (Corporation, A. 2009)

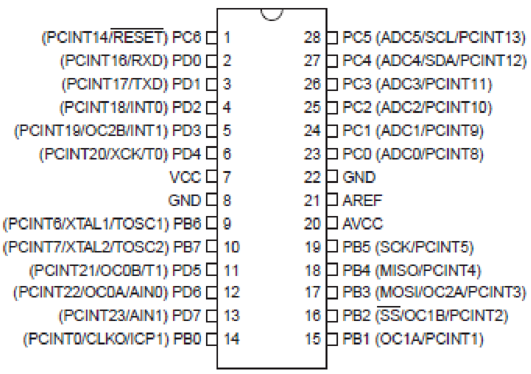

Figure 1: Pin description of ATMEGA. (Corporation, A. 2009)

LM041L Liquid Crystal Display (LCD)

The LM041L is a 16-character by 4-line LCD equipped with an integrated HD44780 controller as shown in figure 2. It operates on a 5V power supply and displays data through bitwise operations. An LCD is constructed using two layers of polarized glass, positioned at a 90° angle to each other, effectively blocking light passage.

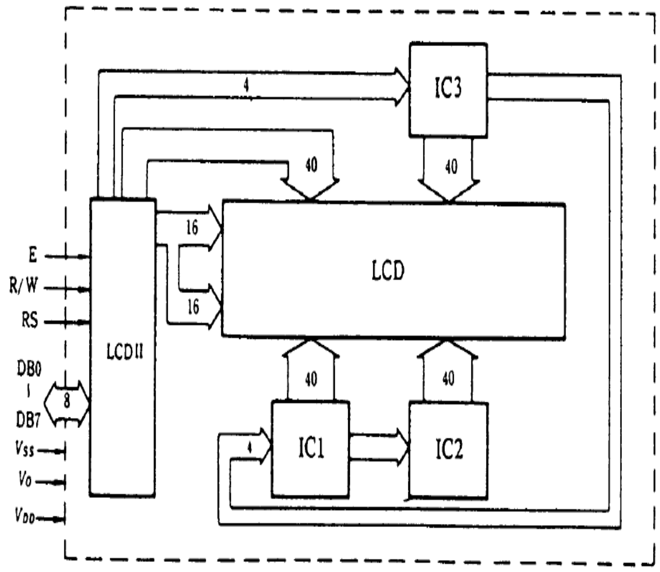

Figure 2: Block diagram of LM041L LCD. (Corporation, A. 2009)

The LCD is controlled by the HD44780 controller, a widely used interface for LCD displays. It operates in two modes: a 4-bit mode, which is more complex but reduces the number of connections needed, and an 8-bit mode, which simplifies data transmission but requires more connections.

LM35 Temperature Sensor IC

The LM35 is an integrated circuit temperature sensor that provides an output voltage directly proportional to the measured temperature. Internally, it consists of three transistors, four resistors, two operational amplifiers, and constant current sources, as illustrated in Figure 3.

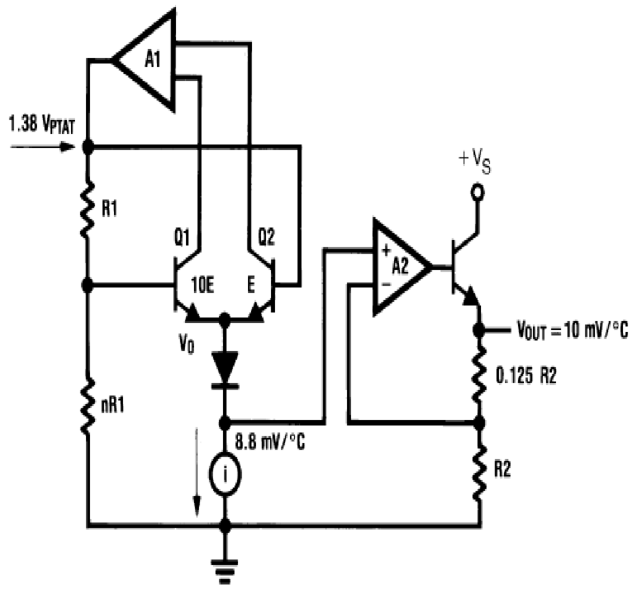

Figure 3: Circuit of LM35 Temperature sensor Instrument, (T. 2016)

The two transistors, Q1 and Q2, positioned at the center of the circuit in Figure 3, are configured so that one operates at one-tenth the current density of the other, despite carrying the same current. This setup generates a voltage across resistor R1 that is directly proportional to the absolute temperature, ensuring a linear relationship between temperature and voltage. (T. 2016)

Amplifier A1 compares the outputs of the two transistors to verify that the voltage at the base of transistor Q1 remains proportional to absolute temperature. Meanwhile, amplifier A2 converts the absolute temperature into Celsius, producing a voltage output. Resistors R1 and R2 are factory-calibrated to achieve high accuracy. (T. 2016). The sensitivity of the temperature controller can be determined using Equation 1.

$$Ave.Sensitivity = \frac{Change\ in\ Volt.}{change\ in\ Temp.} \quad (1)$$

Average sensitivity is measured in Volts per temperature.

Proteus CAD Software

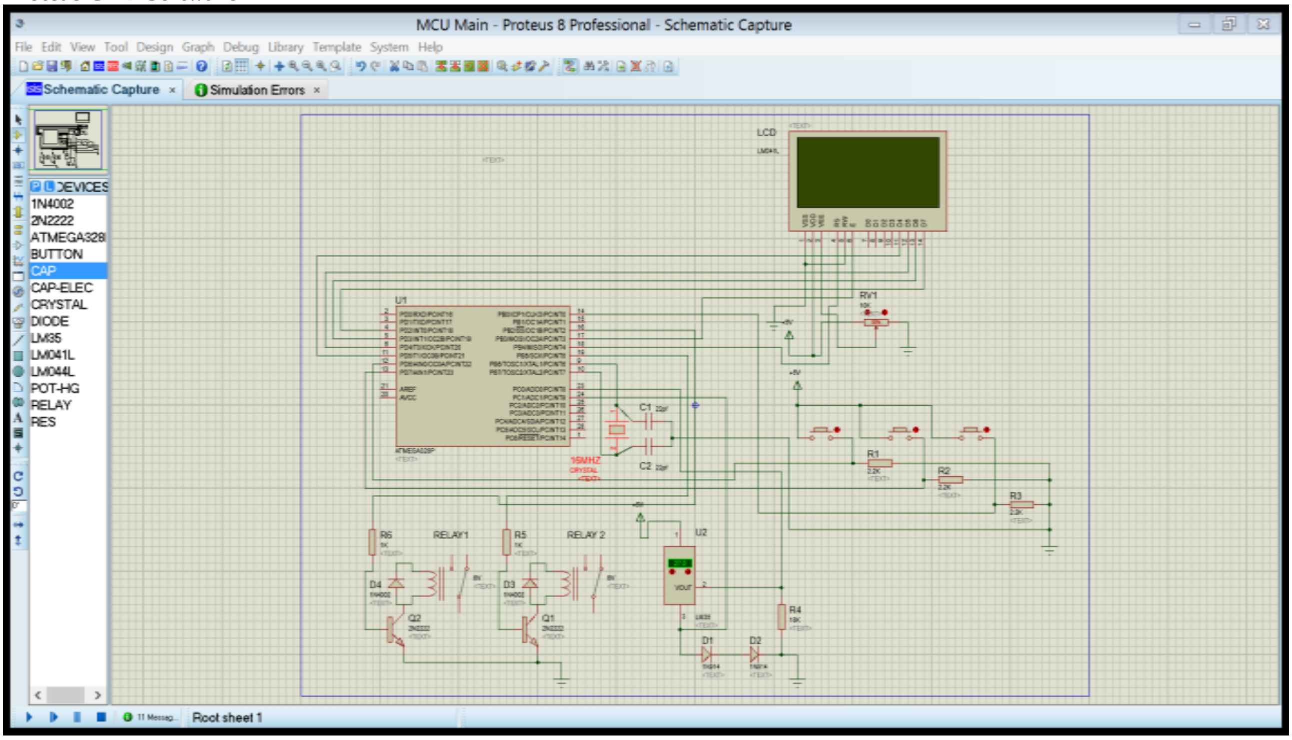

Figure 4: Proteus CAD software window

Proteus is a software tool used for microprocessor and circuit simulation, schematic design, and printed circuit board (PCB) development, as shown in Figure 4. It offers various features, including the flexibility to create custom component footprints and store them in libraries. Additionally, it supports the PADS ASCII file format, a widely accepted PCB footprint description standard, and is compatible with the Boundary Scan Description Language (BSDL), which facilitates electronic testing through the Joint Test Action Group (JTAG) protocol.

System Design and Implementation

The temperature controller design integrates both hardware and software to ensure the efficient operation of the system. The microcontroller-based approach enhances flexibility while reducing the number of components, as most control functions are executed through software rather than hardware.

To develop the temperature controller, a flowchart was created, and an algorithm was designed to enhance accuracy. The program was written, debugged, compiled, and then programmed into the microcontroller according to the predefined flowchart. The controller is designed to measure temperature using an integrated temperature sensor, process the data using the microcontroller's built-in analog-to-digital converter (ADC), compare the readings to the desired range, execute necessary control actions, and display the measured temperature on an LCD screen.

The hardware design includes a microcontroller interface board that connects the ATMEGA microcontroller to the display unit and relay driver circuit. The LCD display is directly interfaced with the ATMEGA using its built-in LCD connection capability as illustrated in Figure 5.

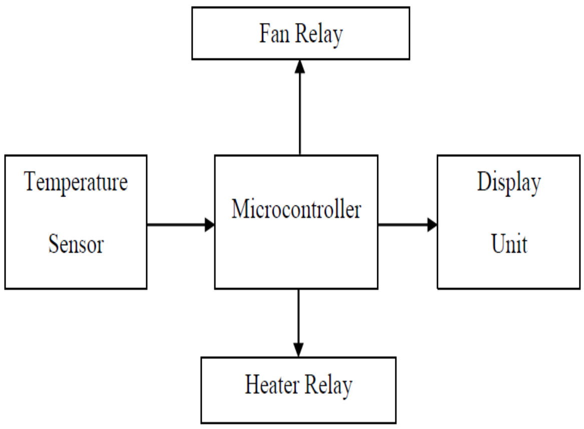

Figure 5: Micro-controller based Variable Temperature Controller

The program is structured into three main parts:

1. A routine for reading the digital value from the analog-to-digital converter (ADC).
2. A routine for computing and displaying the temperature.

3. A routine for comparing the measured temperature with preset values and executing the necessary control actions.

The program follows the following syntax structure:

1. Defining variables required for program execution.
2. Initializing port pins as either input or output.
3. Configuring the microcontroller's ADC (Analog-to-Digital Converter).
4. Reading ADC input values and converting them into temperature readings.
5. Storing the temperature value for further processing.

The push buttons were programmed to function as a menu button, along with "+" and "–" buttons. Pressing and holding the menu button for three seconds opens the maximum temperature setup screen. Pressing it again switches to the minimum temperature setup screen, and another press saves the settings. The "+" button increases the preset temperature value, while the "–" button decreases it. These functions were successfully tested during the simulation process. Figure 4 illustrates the simulated splash screen display in Proteus.

The program operates as follows:

1. Display the temperature on the LCD.
2. Compare the measured temperature with the lower set limit; if the result is positive, proceed to the end; otherwise, turn on the heater.
3. Display the heater status on the LCD.
4. Compare the measured temperature with the upper set limit; if the result is negative, turn on the fan; otherwise, proceed to the end.
5. Display the fan status on the LCD.

Simulation

The program was simulated using Proteus Design Suite 8.1. The simulator required the hex file of the program along with the complete circuit connected to the microcontroller. Lamps were used as loads to visually indicate the switching on and off of the fan and heater. During the simulation, the temperature was adjusted using the control button of the LM35 sensor, triggering the appropriate response either turning the heater or fan on/off as needed. After successfully completing the simulation, an experiment was conducted to observe the communication format between the microcontroller and the LCD. An oscilloscope was used for this purpose, with its horizontal axis set at 500 milliseconds and the vertical axis at 2.2V.

The oscilloscope had four channels:

a. Channel A (yellow) – Connected to D4 of the LCD.
b. Channel B (blue) – Connected to D5 of the LCD.
c. Channel C (pink) – Connected to D6 of the LCD.
d. Channel D (green) – Connected to D7 of the LCD.

This setup is illustrated in Figure 6.

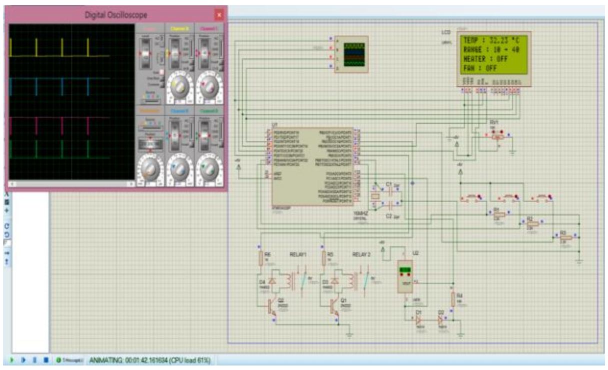

Figure 6: Simulated experimental setup.

The output readings were recorded for different LCD states, including the main menu, maximum temperature settings, minimum temperature settings, main menu with the fan activated, and the maximum temperature saved screen. The results are presented in Figure 7.

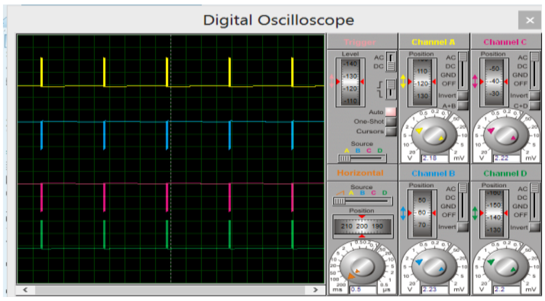

Figure 7: Oscilloscope reading for Main "Menu" Display with Fan and Heater off

Results and Discussion

This section presents the experimental findings of the microcontroller-based variable temperature controller. Table 1 displays the test values obtained from the sensitivity test.

Table 1: Test Values for the sensitivity test

Temperature (°c)	Test 1 (mV)	Test 2(mV)	Test 3(mV)	Average test (mV)	Average Sensitivity (mV/°C)
23.93	224	223	224	223.67	9.34
24.41	229	227	228	228.00	9.34
25.39	240	239	240	239.67	9.44
26.37	249	249	247	248.33	9.41

The graphical representation of the values from Table 1 is illustrated in Figure 8.

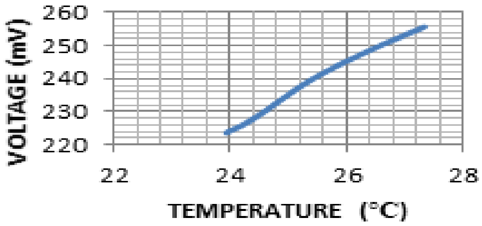

Figure 8. Graph of voltage against temperature

Based on Figure 8, the average sensitivity of the temperature controller is calculated using Equation 1.

Conclusions

The design was successfully executed, as demonstrated in Figures 9, 10, and 11: A red light bulb turns on, with the LCD displaying the heater's status. The temperature sensor placed near an ice block triggers the blue bulb, with the LCD displaying the fan's status.

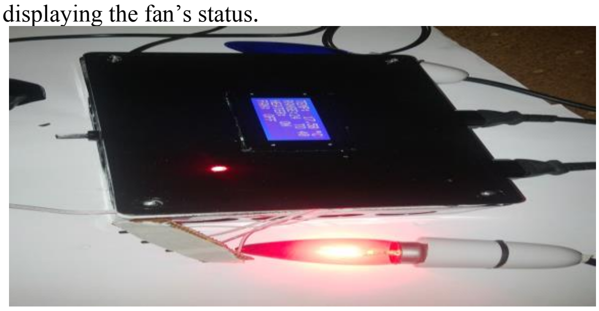

Figure 9: Red Light Bulb going on with the LCD Displaying the Heaters Status

Figure 10: Temperature Sensor Close to an Ice Blocks

Figure 11: Blue Blub going on with the LCD Displaying the Fan's Status

Factors such as environmental conditions, the absence of a high-precision multimeter, and delays in the program code of the temperature controller influenced the calculated average sensitivity value. An automatic temperature controller is essential for industrial processes and can be applied in both industrial and residential temperature regulation. The design, construction, and testing of a microcontroller-based variable temperature controller using the ATMEGA328P microcontroller was successfully implemented. The average sensitivity was determined to be 9.38mV/°C, which closely aligns with the 10mV/°C specified in the LM35 temperature sensor datasheet. This research contributes to the existing body of knowledge by demonstrating the use of a microcontroller-based variable temperature controller in applications such as thermal flow processing, temperature-controlled incubators, nurseries, digital thermometers, and room temperature regulation.

References

Auwala, A.M., Morufata, A.T., Yusufb, M.A., Muhammadc, A., Sanid, M.A., & Abubakare, S.B.U. (2017). Design and Implementation of a Microcontroller-Based Digital Thermometer. *Technology (ICONSEET)*, 2(17), pp. 132-140.

Amin, M. R., Ghosh, A., & Hadi, A. (2018). Design and implementation of microcontroller-based programmable smart industrial temperature control system: An undergraduate level approach. *International Journal of Control*, *11*(2), pp. 45–52.

Atayero, A. A., Alan, E., & Adeyemi, A. (2015). Design and construction of a microcontroller-based automatic irrigation system. In *Proceedings of the World Congress on Engineering and Computer Science* (Vol. 1), pp. 72–76). IAENG.

Corporation, A. (2009). *Gravite ATMEGA328 Data Sheet*.

Das, T.K. & Das, Y. (2013). Design of a Room Temperature and Humidity Controller Using Fuzzy Logic. *American Journal of Engineering Research*. 1, pp. 86-97.

Gouda, K., Preetham, V., & Swamy, M.S. (n.d.) (2025). Microcontroller-Based Real Time Weather Monitoring Device with GSM.

Instrument, T. (2016). *LM35 Precision Centigrade Temperature Sensors*.

Jebelli, A., & Yagoub, M. C. (2015). Development of sensors and microcontrollers for small temperature controller systems. *Journal of Automation and Control Engineering*, *3*(2), pp 134–139.

Kumar, K., & Kurian, C. P. (2014). Model-based control using C2000 microcontroller. In *2014 International Conference on Advances in Energy Conversion Technologies (ICAECT)* pp. 13–20. IEEE.

Li, Z. (2004). *Design of a Microcontroller-Based, Power Control System for Microwave Drying*. Master of Science. Department of Bioresource Engineering, McGill University, Montreal.

Lwin, J. T., & Ya, A. Z. (2014). Development of microcontroller-based temperature and lighting control system in smart home. *International Journal of Scientific and Research Publications*, *4*(6), pp 1–6.

Patel, S.S., Patil, D.R., & Brahmbhatt, B.H. (2014). Design of a Low Cost Temperature Controller for High Temperature Furnaces Used in Crystal Growth. *International Journal of Research in Engineering and Technology*, 3, pp. 187-192.

Santoso, I., Widodo, T.S., Susanto, A., & Tjokronagoro, M. (2014). Application of Fuzzy Logic for Temperature Control in Microcontroller-Based 2.45 GHz Microwave Hyperthermia Device. *International Journal of Applied Engineering Research*, 9, pp. 665-673.