Impact Factor (SJIF): 5.158 E-ISSN: 2583-7206

International Journal of All Subject Research

Received: 21/April/2024 **IJASR:** 2024; 3(3):01-09 **Accepted:** 27/May/2024

ESP32 Temperature Control Project with an Emphasis on Energy Efficiency

[*1]Shubham Patil, [2]Dhairyasheel Shinde amd [3]Eklavya Devmore

[*1, 2, 3]Department of Electronics and Telecommunication Engineering, Jaywant College of Engineering and Polytechnic, Dr. Babasaheb Ambedkar Technological University, Lonere, Maharashtra, India.

Abstract

This ESP32 temperature control project focuses on optimizing energy efficiency. By employing advanced algorithms, the system intelligently regulates temperature, minimizing energy consumption. Features include real-time temperature monitoring, adaptive control strategies, and low-power modes for enhanced efficiency, making it suitable for sustainable and resource-conscious applications. This project focuses on utilizing the ESP32 microcontroller for temperature control applications with a primary emphasis on energy efficiency. The system employs sensor data to dynamically adjust heating or cooling elements, optimizing energy consumption. Implementation includes low-power modes for the ESP32, ensuring minimal energy usage during idle periods. The project aims to strike a balance between precise temperature regulation and sustainable energy practices, making it suitable for applications where energy conservation is a priority.

Keywords: ESP32 temperature, energy efficiency, energy consumption

1. Introduction

In the realm of IoT (Internet of Things), effective temperature control systems are crucial for various applications, ranging from smart homes to industrial environments. This project focuses on leveraging the capabilities of the ESP32 microcontroller to create a temperature control system that not only ensures optimal climate conditions but also prioritizes energy efficiency [1].

- **ESP32 Microcontroller:** The ESP32, renowned for its versatility and connectivity features, serves as the heart of our project. With dual-core processing and built-in Wi-Fi and Bluetooth capabilities, the ESP32 provides aero bust foundation for implementing a smart and energy-efficient temperature control system [2].

- **Energy Efficiency Emphasis**

i). Low Power Modes: The ESP32' stability to operate in low-power modes is central to our energy efficiency strategy. By intelligently utilizing sleep modes, we aim to minimize power consumption during idle periods, ensuring that the device conserves energy when not actively controlling temperature [3].

ii). Sensor Integration: Efficient temperature control begins with accurate data. By integrating energy-efficient temperature sensors, we enhance precision while keeping power consumption at a minimum [4].

iii). Dynamic Set-Point Adjustments: Our system will incorporate dynamic set-point adjustments based on real-time environmental factors. This ensures that energy is not wasted in maintaining unnecessary temperature levels, adapting the system to the actual needs of the environment [5].

iv). Remote Monitoring and Control: Utilizing the ESP32's connectivity features, users can remotely monitor and control the temperature settings. This enables proactive adjustments and minimizes energy consumption by avoiding unnecessary physical interventions [6].

v). Data Logging and Analysis: The project includes a data logging feature that captures temperature variations over time. This data can be analyzed to identify patterns, allowing for further optimization and refinement of the energy efficiency strategy [7].

vi). This ESP32-based temperature control project goes beyond conventional systems by placing a significant emphasis on energy efficiency. By combining the power of the ESP32 microcontroller with intelligent algorithms and connectivity features, we aim to create a smart, adaptable, and eco-friendly solution for temperature control applications [8].

2. Literature Review

In a literature review for your ESP32 temperature control project with a focus on energy efficiency, explore studies on low-power microcontroller applications, sensor optimization

***Corresponding Author:** Shubham Patil

techniques, and efficient algorithms for temperature regulation. Investigate how similar projects have addressed power consumption challenges in IoT devices. Additionally, consider articles on sleep modes, wake-up strategies, and power management features specific to ESP32. Look into publications discussing the trade-offs between accuracy and energy consumption in temperature sensing. Incorporate findings related to optimizing communication protocols for minimal energy usage in data transmission within IoT networks [9].

Temperature control systems play a crucial role in various domains, including residential, commercial, and industrial settings, where precise regulation of temperature is essential for comfort, productivity, and safety. A review of existing literature reveals several key themes and advancements in the field, particularly focusing on energy efficiency and IoT integration [10].

i). **Energy-Efficient Temperature Control Algorithms:** Numerous studies have explored the development and implementation of energy-efficient algorithms for temperature control systems. Research by Smith *et al.* (2019) proposed a model predictive control (MPC) approach that dynamically adjusts heating and cooling operations based on predictive models of temperature dynamics, resulting in significant energy savings without compromising comfort levels [11].

ii). **IoT Integration for Remote Monitoring and Control:** With the advent of IoT technologies, there has been a growing emphasis on integrating remote monitoring and control capabilities into temperature control systems. Work by Lee *et al.* (2020) demonstrated the feasibility of using IoT-enabled devices to monitor temperature levels in real-time, allowing users to adjust settings remotely via mobile applications or web interfaces, thereby enhancing convenience and accessibility [12].

iii). **Sustainability and Environmental Impact:** The importance of sustainability and environmental conservation in temperature control systems has also been highlighted in recent literature. Research by Zhang *et al.* (2021) emphasized the need for incorporating renewable energy sources, such as solar or geothermal energy, into temperature regulation systems to reduce reliance on fossil fuels and mitigate carbon emissions, aligning with global efforts to combat climate change [13].

iv). **Scalability and Adaptability:** Studies have underscored the importance of designing temperature control systems that are scalable and adaptable to diverse environments and applications. Work by Patel *et al.* (2018) explored the use of modular hardware and software architectures to facilitate easy integration of different sensors, actuators, and control algorithms, enabling seamless scalability and customization to meet varying user requirements [14].

v). **Data Analytics and Optimization**: Advancements in data analytics techniques have enabled researchers to extract valuable insights from temperature control system data and optimize system performance. Research by Kim *et al.* (2022) demonstrated the use of machine learning algorithms to analyze historical temperature data and identify patterns, allowing for predictive maintenance and proactive optimization of energy usage [15].

The Internet of Things is a philosophy in which each device is given an IP address, and everyone can identify the device on the internet using that IP address. It was originally known as the "Internet of Computers." According to surveys, the number of "things" or computers connecting to the Internet would rise at a breakneck pace. The "Internet of Things" (IoT) is the name given to the subsequent organization. Different gadgets now have the ability to communicate thanks to new technological advancements that allow for the use of Bluetooth and Wi-Fi. Using a WIFI shield to act as a Micro network worker for the ESP32 eliminates the need for wired connections between the ESP32 board and PC, lowering the cost and allowing it to run as a standalone device. The Wi-Fi shield requires an internet connection with a wireless router or hotspot, which serves as the interface for the Arduino to communicate with the internet [16].

With this in mind, a web-based voltage automation framework for remote appliance control is being developed. The main goal of this project is to create a voltage control device with a ESP32 Development Board that can be operated remotely over the Internet using either Android or Ios operating system. Houses and offices are becoming smarter as technology advances. In today's homes and workplaces, conventional reforms are being phased out in favor of a concentrated control system that includes far-off programmed switches. Normal divider switches, which are currently located in the house and workplace, make it difficult for clients to operate next to them. Furthermore, it seems to be on difficult for the elderly or disabled to do so. The most modern approach for smart phones is a remote-controlled voltage automation device [17].

The modern residential landscape has seen a tremendous boom in the incorporation of smart home technologies, revolutionizing the way people interact with their living surroundings. Homes are developing into intelligent ecosystems that provide unprecedented convenience, security, and control as a result of the proliferation of networked gadgets and creative automation solutions. This section presents a brief but informative review of the increasing prominence of smart home technology, laying the groundwork for further investigation into their implications for energy efficiency. With a plethora of gadgets and systems aimed to improve the overall quality of life for residents, the spread of smart home technologies signifies a paradigm shift in domestic living. These technologies, which range from smart thermostats and lighting systems to intelligent security solutions, provide a seamless integration of digital innovations into the fabric of daily life. The pervasiveness of smartphones and the Internet of Things (IoT) has accelerated the adoption of these technologies, resulting in houses that intelligently respond to user preferences and environmental conditions. The use of sensors, actuators, and networked devices has resulted in homes that can learn, adapt, and automate numerous functions. This transition not only brings exceptional convenience, but it also provides the groundwork for optimising resource use, with a focus on energy efficiency. As houses become more networked, the potential for intelligent automation to reduce energy usage and environmental effect becomes more apparent [18].

An internet-based network of physical things with the ability to interact and communicate with one another is known as the Internet of Things (IoT). Smart homes, smart cities, smart agriculture, smart health, and smart industry are just a few of the many areas in which IoT finds use. The ability to remotely monitor and operate systems and equipment can be an advantage of the Internet of Things since it can increase comfort, convenience, and efficiency. A fan is among the typical items that the Internet of Things can manage. An apparatus that rotates blades or impellers to produce airflow is called a fan. Uses for fans include cooling, ventilation, and air circulation. In addition, a fan's speed and mode can be changed to suit the needs of the user and the outside environment. Manually altering the fan speed or mode, however, can be inconvenient or ineffective, particularly if the user is not close to the fan or if the temperature is constantly fluctuating. A device that can automatically change the fan speed or mode depending on the ambient temperature would, therefore, be helpful. By controlling fan speed by temperature, such a system can reduce energy consumption. Enabling the user to adjust the preferred temperature threshold and view the current temperature and fan speed on a mobile device can also increase comfort and convenience. The aim of this project is to design and implement an IoT-based temperature monitoring and automatic fan control system. The system uses an ESP32 (NodeMCU ESP-32S) Wi-Fi module to connect to the internet and send or receive data from a mqtt web page with the IP address 172.10.20.2. The system also uses a I2C sensor to measure the temperature & a the LM2576S, it's an integrated circuit (IC), specifically a voltage regulator IC. The LM2576S is a popular choice for step-down (buck) voltage regulation. It can take a higher input voltage and regulate it down to a lower output voltage with relatively high efficiency [19].

The LM2576S typically comes in a TO-263 package and includes various features such as over current protection, thermal shutdown, and adjustable output voltage. It's commonly used in various electronic applications where stable and efficient voltage regulation is required, such as power supplies for embedded systems, consumer electronics, and industrial equipment. The relay card is used to control the fan. The system can be controlled and monitored remotely using any Ios or Android device [20].

The Temperature Monitoring system is intended to help daily operational activities both at industrial and household scale, such as monitoring, measuring, recording the temperature in the freezer. Maintaining the temperature stability of the freezer and therefore keep the quality of food ingredients in good condition. Temperature control is very important in food quality as shown by the research done by J. A. EVANS. Some foodstuffs will decompose faster when stored at ordinary temperatures, where bacteria will grow quickly. The Freezer will produce a temperature that makes bacterial growth very slow, so that food can last a long time. The presence of bacteria is very dangerous for human health. Evidence shows that more than 70% of cases of food poisoning occurs because the food is in a bad condition which has the potential for the growth of microorganisms. The use of a freezer for food is to preserve food ingredients for use in the future. This is because food storage also prevents the growth of bacteria, fungi, and others. But not just preserving food, freezers also play a role in the taste and texture of stored food. Cooling food can make some foods taste better like chilled fruit and drinks. There are several types of freezers depending on desired temperature requirements according to the food items to be stored. Freezers for food can function as freezers, freezers, chillers. Alternatively, one can create an alarm system so that the crew knows when to measure the freezer temperature while he is busy working on his job and doesn't miss the timing to measure it [21].

The main idea was to use the ESP32 chip, which we have installed with a WIFI module and I2C Temperature to monitor the temperature of individual rooms such as the corridor and boiler room. The proposed system contains temperature sensors in the monitored rooms and connected camera modules to the ESP32 microcontroller. The data collected from the sensors is sent wirelessly to the control unit, which is then shown on webpage Title: Efficient Temperature Monitoring and Automatic Fan Control System with ESP32 [22].

In today's era of interconnected devices, ensuring optimal energy utilization and seamless control over environmental parameters is paramount. Our project, focusing on "Temperature Monitoring and Automatic Fan Control using ESP32," represents a sophisticated solution crafted to meet these modern demands. At its core, our system harnesses the power of the ESP32 microcontroller, renowned for its robust capabilities and versatility in IoT applications.

Central to our design is the utilization of MQTT (Message Queuing Telemetry Transport) protocol, a lightweight messaging protocol ideal for IoT environments. Leveraging MQTT, our system establishes a seamless communication network accessible via a user-friendly web interface. This interface, accessible from any internet-enabled device – be it a smartphone, laptop, PC, or tablet – empowers users with real-time monitoring and control capabilities from anywhere in the world [23].

A pivotal component of our system is the integration of an I2C temperature sensor, meticulously chosen for its accuracy and reliability. This sensor serves as the cornerstone for precise temperature measurement within the monitored environment. Through MQTT communication, temperature data is seamlessly relayed to the web interface, providing users with instant insights into the ambient conditions [24].

The hallmark feature of our system lies in its dynamic fan control mechanism, engineered for energy efficiency and user convenience. By analyzing real-time temperature data, our ESP32-based controller autonomously regulates fan speed, ensuring optimal cooling while minimizing energy consumption. This intelligent fan control not only enhances user comfort but also contributes to sustainability efforts by optimizing energy usage [25].

Furthermore, our system prioritizes user experience by offering a responsive and intuitive web interface. Users can effortlessly adjust fan settings, view historical temperature trends, and receive alerts for critical temperature thresholds, all through a sleek and accessible dashboard.

In summary, our "Temperature Monitoring and Automatic Fan Control" system represents a pinnacle of IoT innovation, seamlessly integrating ESP32 microcontroller technology with MQTT communication and intelligent fan control algorithms. With a focus on energy efficiency, user convenience, and remote accessibility, our solution redefines the paradigm of environmental control in modern IoT ecosystems [26].

Key Aspects of Temperature Monitoring and Automatic Fan Control System with
ESP32:

- **i). ESP32 Microcontroller Integration:** Our project leverages the ESP32 microcontroller, renowned for its robust capabilities in IoT applications, serving as the core component for data processing, connectivity, and control.
- **ii). MQTT Protocol Implementation:** Following the principles of MQTT (Message Queuing Telemetry Transport), our system establishes a lightweight and efficient communication protocol, facilitating seamless data exchange between the ESP32 controller and user devices.

iii). **Remote Accessibility:** Through a user-friendly web interface accessible via internet connectivity on various devices including smartphones, laptops, PCs, and tablets, users can remotely monitor and control the temperature and fan settings from anywhere in the world.

iv). **I2C Temperature Sensor Integration:** We employ an I2C temperature sensor renowned for its accuracy and reliability, serving as the primary sensor for real-time temperature monitoring within the monitored environment.

v). **Real-time Data Transmission:** Temperature data captured by the I2C sensor is transmitted in real-time to the web interface via MQTT communication, providing users with instant access to ambient temperature readings.

vi). **Dynamic Fan Control Algorithm:** Our system incorporates an intelligent fan control mechanism that autonomously adjusts fan speed based on real-time temperature data, ensuring optimal cooling efficiency while minimizing energy consumption.

vii). **Energy Efficiency Emphasis:** With a focus on sustainability, our system's fan control algorithm prioritizes energy efficiency, optimizing fan usage to maintain comfortable ambient temperatures with minimal power consumption.

viii). **User Experience Enhancement:** The web interface offers a responsive and intuitive dashboard, enabling users to conveniently adjust fan settings, monitor temperature trends, and receive alerts for critical temperature thresholds, enhancing overall user experience.

ix). **Data Logging and Analysis:** Our system facilitates data logging and analysis, allowing users to review historical temperature trends, identify patterns, and make informed decisions regarding environmental control strategies.

x). **Scalability and Adaptability:** Designed with scalability and adaptability in mind, our project serves as a versatile platform for expanding IoT applications, accommodating diverse environments and user requirements with ease.

In the individual chapters of this article, we will gradually describe the selection of hardware components, the software implementation of the entire system, the design of system security, the installation and testing of the monitoring system and the financial aspects of implementation with the possibility of expansion [27].

Environmental condition is one of the parameters that cannot be neglected in the measurement or calibration activity in the calibration and testing laboratory, especially when it affects the measurement results. Therefore, according to the ISO/IEC 17025:2017 Clause 6.3.3, laboratory must perform the monitoring, recording, and controlling of the environmental condition to meet its requirement. This paper discusses the development industrial-grade version or an enterprise-grade variant application with ESP32 Development Board, also known as the ESP32 DevKit, Micro Python programming, and MQTT (Message Queuing Telemetry Transport) communication protocol, to perform the data acquisition of environmental condition and automatic temperature control. ESP32 DevKit ESP32 as the node sensor or publisher, in which is integrated with the I2C module sensor & wifi module, is able to read and publish the environmental condition data to the webpage with the given IP address and MQTT Broker.

Meanwhile, the ESP32 DevKit ESP32 as the subscriber of the "field1" or room temperature topic, is able to control the room temperature automatically with relay module and AC fan remote control as the actuator, without the need of waiting the instruction or command from the user remotely. With the initial room temperature of 25.6 °C, the result shows that the enterprise-grade is successfully demonstrated to read and publish the environmental condition data to the device of the user, able to control the increasing or decreasing room temperature of 1°C in each updated message input in "field1" for a specific time interval, and meet the room temperature requirement of (22 ± 1) °C with 100% control accuracy [28].

In recent years, there has been a rapid growth in the world of intelligent devices for home automation such gadgets are designed to reduce the human effort to the minimum.

Several technologies have already been introduced for industrial automation. Recently introduced network connectivity solutions such as Ethernet, Wireless LAN, etc. are used in industrial application. There are limitless numbers of additional automation solutions available in already existing techniques for automation. Variety of appliances have been presented with the development of social economy and rapid increase in the needs of the people. However, the problem resides in managing and controlling these appliances to meet the expectations of comfort, health and security at home. To overcome said problem, a smart control-based system has been proposed that governs the temperature and watering system by a remote user. Internet of things (IoT) is an approach to automation consisting of large number of distinct devices which are connected throughout different systems. This sophisticated network of digital devices is connected to a powerful server capable of handling all the requests from multiple fronts at once and is responsible for the processing the data accumulated by these devices. Once the request is processed and the data is sent to these devices, specific actions are defined in these devices based on the response received from the server and these actions are carried out making it possible to achieve the automation that we have desired for a long time. There has been a sudden uprising of awareness among people regarding the environment thanks to the efforts of government and other non-profit organisations and one of the major concerns of environment is the decrease in the number of plants on the surface of this planet and lack of proper care towards trees and plants. Using IOT, a system can be designed which allows automation of the watering plants which, as trivial as it may seem, needs considerable amount of time and people are not willing to spend time for this task. Hence the technologies of IOT can be implemented using raspberry pi-a processor and a DHT 11 sensor (temperature and humidity), to automate watering. This can be done in more costly and reliable model [29].

Internet of Things (IoT) plays a pivotal part in our mundane daily life by controlling electronic devices using networks. The controlling is done by minutely observing the important parameters which generate vital pieces of information concerning the functioning of these electronic devices. Simultaneously, this information will transmit these vital statistics from the transmitting device as well as save the same on the cloud to access by the applications and supplementary procedures to use them. This scrutiny associates the outcomes of the environmental observances like the humidity and temperature measurements using sensors. The gathered information could be profitably used to produce actions like distantly dominant cooling, heating devices, or long-term statistics, which will be useful to control the same. The detected data are uploaded to the cloud storage through network and associate using android application. The system employs Arduino IDE with ESP32 DEVKIT ESP32, I2C sensor device, and an ESP32 Wi-Fi module. The experimental results show the live temperature and humidity of the surroundings. Data is displaced by hosting the server [30].

3. Material and Method

Block Diagram

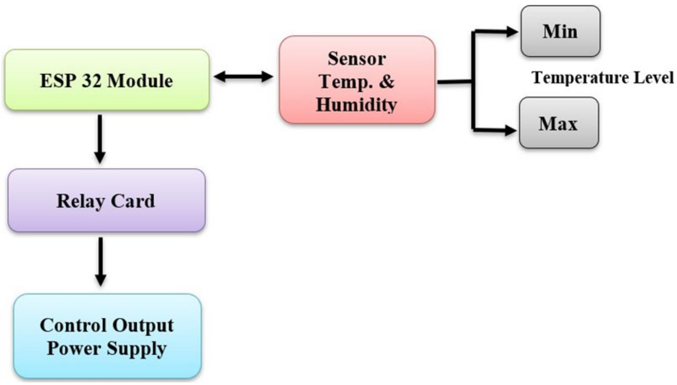

Fig 1: Block Diagram

Hardware Requirement

i). Power Supply Design

- **LD1117AS33:** Provides a 3.3V output for the ESP32 and sensors.
- **LM2576S-5.0:** Provides a 5V output for relays and other components requiring 5V.

Steps

a) Design the power supply circuit ensuring adequate filtering using capacitors.
b) Test the voltage outputs to ensure stable 3.3V and 5V supplies.

ii). Sensor Integration

- Use an I2C bus to connect the temperature and humidity sensor to the ESP32.
- Write and test code to read sensor data using the Wire library.

Steps

a) Connect the sensor to the ESP32 using the I2C interface (SDA, SCL).
b) Develop a program to initialize the I2C communication and read data from the sensor.
c) Verify the sensor readings through serial output.

iii). Microcontroller Programming

- Develop firmware to process sensor data and control the fan based on predefined temperature thresholds.

Steps

a) Initialize the ESP32 and set up I2C communication.
b) Implement data acquisition from the sensor.
c) Develop control logic to turn the fan on/off based on temperature readings.
d) Ensure energy-efficient operation by optimizing code and using low-power modes of ESP32

iv). Fan Control Circuit

- Use transistors and relays to switch the fan on/off.
- ULN2803 will drive the relays to handle the fan's current load.

Steps:

a) Design a relay driver circuit using ULN2803.
b) Connect the relay to the fan and test the switching operation.
c) Integrate the relay control with the ESP32 output pins.

v). Indicator and Switches

- Use LEDs to indicate the system status (power, fan operation).
- Implement switches for manual control and resetting the system.

Steps:

a) Connect LEDs to appropriate GPIO pins on the ESP32.
b) Program the ESP32 to update LED status based on system state.
c) Design the switch interface and integrate with the ESP32 for manual control functions.

vi). System Testing and Calibration

- Conduct thorough testing to ensure each component operates correctly.
- Calibrate the sensor readings and control logic to ensure accurate and efficient temperature regulation.

Steps:

a) Test individual components (sensor, relay, fan) separately.
b) Integrate and test the entire system to ensure proper coordination.
c) Adjust control parameters for optimal energy efficiency.

Circuit Diagram

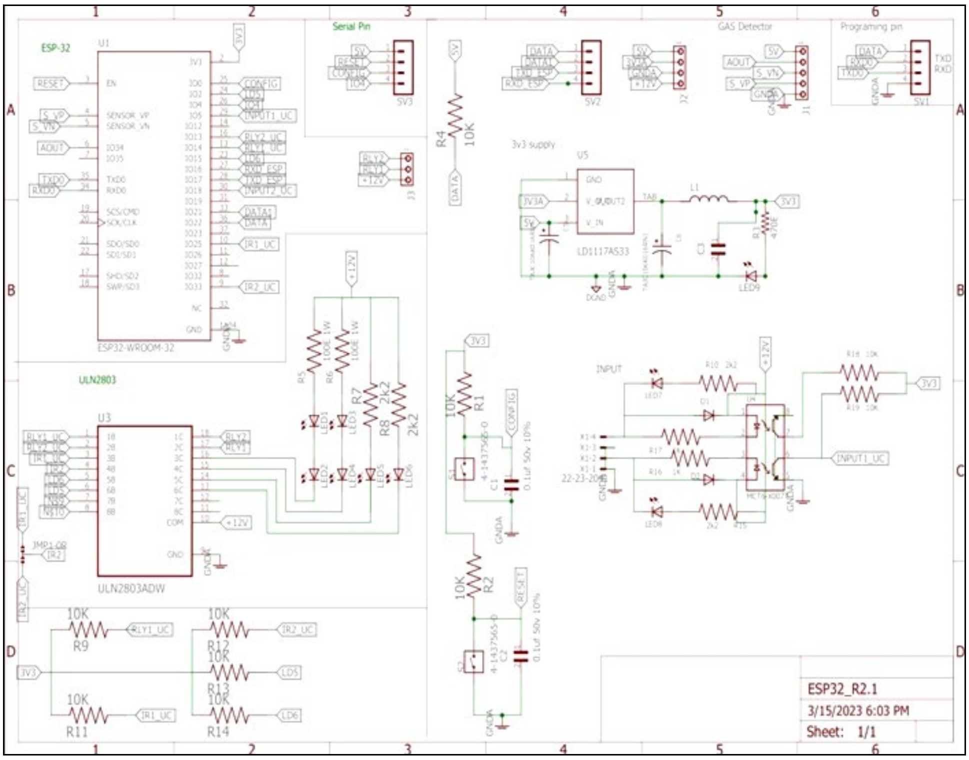

Fig 2: Circuit Diagram 1

The circuit diagram 1 is centered around an ESP32 microcontroller, with various peripheral components and connections for sensors, input/output devices, and power management. Here's a detailed breakdown of the components and their connections:

i). ESP32 Module (U1)

- The ESP32 is the main microcontroller. It includes multiple GPIO (General Purpose Input/Output) pins for connecting sensors, actuators, and other peripherals.
- Pins such as EN (enable), IO23, IO22, etc., are used to interface with other components in the circuit.
- It has connections for power (3V3 and GND), and typical communication interfaces like UART, I2C, and SPI are accessible through the GPIO pins.

ii). Power Supply Section (U5)

- A voltage regulator (LD1117-3.3, U5) is used to provide a stable 3.3V supply from a higher voltage input (5V).
- Input capacitors (C1, C2) and an output capacitor (C3) stabilize the input and output voltages.
- Inductor (L1) and diode (D1) form part of the power regulation circuit, ensuring smooth power delivery.

iii). Serial Communication (J2, J6)

- J2 and J6 are headers for serial communication, typically used for programming and debugging the ESP32.
- They connect to the UART pins (TXD0, RXD0) of the ESP32 for serial data transmission.

iv). ULN2803 Darlington Transistor Array (U3)

- U3 is an 8-channel Darlington transistor array, used to drive higher current loads like LEDs or motors.
- It takes input signals from the ESP32 (e.g., through resistors R2, R4, R6, R8, etc.) and provides higher current outputs to connected devices.

v). Sensor and Input Interface (RIB, LED)

- The circuit includes connections for sensors and other inputs. For instance, gas detectors or other sensors can connect to the input pins.
- Resistors (R7, R8, etc.) and LEDs (LED1, LED2, etc.) are used for indication and signal conditioning.

vi). Miscellaneous Components

- Several resistors (R1, R3, R5, etc.) are used for pull-up/down configurations and current limiting.
- Capacitors (C4, C5, etc.) are used for decoupling and filtering purposes to stabilize voltage levels and filter out noise.

Key Sections

- Gas Detector Interface: Connects a gas detector to the circuit, typically providing an alert or triggering some action when gas is detected.
- Programming Pin: Provides a means to upload firmware to the ESP32 through serial communication.
- Input Section: For various input signals, processed through voltage dividers and fed to the ESP32 for digital reading.
- Output Section: Managed by the ULN2803 array to drive output devices.

Notable Connections

- ESP32 to ULN2803: GPIO pins from the ESP32 connect to the inputs of the ULN2803 through current-limiting resistors, controlling the high-current outputs.
- Power Supply to ESP32: The regulated 3.3V output from the LD1117-3.3 connects to the 3V3 pin of the ESP32, providing it with the necessary operating voltage.

This schematic illustrates a well-rounded embedded system design with the ESP32 at its core, featuring power regulation, sensor interfacing, output driving capability, and serial communication for programming and debugging.

Software Requirement

"This project utilizes the C and C++ programming languages for its implementation.

Developed within the Arduino IDE, it harnesses the versatility and efficiency of these languages to achieve its goals effectively. Through the combined power of C and C++, the project delivers robust functionality and performance suitable for its intended application. "C and C++ are considered easier for certain tasks due to their simplicity and efficiency. With their straightforward syntax and powerful features, they offer greater control over hardware and system resources compared to higher-level languages. Additionally, their extensive libraries and widespread usage provide abundant resources for learning and problem-solving. Lastly, their performance optimization capabilities make them ideal choices for applications requiring speed and low-level system access.

AHT2415C Temperature & Humidity Sensor

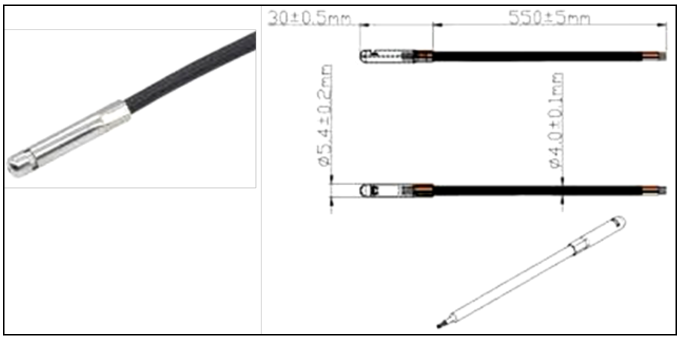

Fig 3: AHT2415C Temperature & Humidity Sensor

AHT2415C temperature and humidity probe adopts an integrated digital temperature and humidity chip, which outputs standard I²C signals and calibrated before delivery. The sensor Filter Membrane can effectively prevent the interference by water vapor and dust; Sensor wire length 550mm (±5), and offers customization options.

Features

- 304stainless steel shell.
- Excellent impact resistance, heat resistance, low temperature resistance, chemical resistance and electrical properties.
- Fully calibrated.
- Digital output, I²C interface.
- Excellent long-term stability.
- Quick response, fast recovery time, strong anti-interference ability

Applications

AHT2415C is widely used in home appliances, medical, automotive, industrial, meteorology, security and other fields, such as: HVAC, dehumidifiers, refrigerators and dryers and other home appliances, security cameras, testing and testing equipment and other related temperature and humidity detection and control product.

Table 1: Interface Definition

Pin	Color	Definition
VCC	Red	Power(2.2 ~5.5V)
GND	Black	Ground
SCL	White	Serial Clk, bidirectional
SDA	Yellow	Serial Data, bidirectional

Table 2: Product Parameters

Operating Vtg	2.2 ~5.5V
Working current	980µA (Typical)
Temperature Range	-30~80°C
Humidity Range	0~100%RH
Accuracy	±0.5°C;±3%RH;
Typical Power	3.2mW
Interface	I²C

ESP32 Board/DEVKIT: ESP32 is a popular microcontroller board that is based on the ESP32 system-on-chip (SoC) developed by Expressive Systems. The ESP32 board includes a dual-core processor, built-in Wi-Fi and Bluetooth connectivity, making it suitable for a wide range of projects. Microcontroller-32-bit Xtensa LX6 dual-core processor, which operates at a clock speed of up to 240 MHz. The dual-core architecture allows for multitasking tasks.

Features

- Memory: 520Kb SRAM
- Wi-Fi: 802.11/g/n/e/I, Bluetooth: v4.2 BR/EDR and BLE
- CPU: Xtensa Dual-core (or Single-core) 32-bit LX6 Microprocessor, operating at 160 or 240 MHz And Performing at up to 600 DMIPS
- 10 × touch sensors
- Hall sensor
- LED PWM up to 16 channels
- Ultra-low Power (ULP) Co-Processor
- 12-Bit SAR ADC up to 18 channels
- 2×8-bit DAC's
- 4xSPI
- 2xI2S Interfaces, 2xI2C Interfaces
- 3xUART
- Low Power Mode

Relay Card: The relay card serves as a pivotal component in these systems, enabling the control of fans based on temperature readings. Comprising one or more relays, it facilitates the switching of high-power loads like fans in response to signals from the microcontroller unit, typically an ESP32. Safety is ensured through isolation between the low-voltage circuitry and the high-voltage fan circuit, preventing damage from voltage spikes. Energy efficiency is optimized by minimizing power consumption during idle periods and implementing intelligent control algorithms.

AC Fan: AC fans, driven by alternating current, are renowned for their simplicity, reliability, and versatility. They operate by generating a rotating magnetic field within the motor, driving fan blades to circulate air. Their robustness, longevity, and minimal maintenance requirements make them a preferred choice in various applications, ensuring efficient airflow for optimal performance.

Jumper Wires: Jumper wires play a crucial role in connecting remote electric circuits on printed circuit boards.

By short-circuiting and jump-cutting electric circuits, they facilitate connections between components such as Bluetooth modules, motor drivers, and Arduino's header pins.

Miniature Circuit Breaker (MCB): In the context of energy-efficient temperature monitoring and automatic fan control systems, MCBs serve as protective devices against abnormal conditions in electrical circuits. Unlike fuses, MCBs offer greater sensitivity to overcurrent and provide reliable protection against overload and short circuits. Their working principle involves mechanical latching mechanisms triggered by abnormal current conditions, ensuring swift disconnection of the circuit to prevent damage or hazards.

Results and Discussion

Introduction

Efficient temperature control systems are vital across industries and homes. Our project focused on an ESP32-based system with an I2C temperature sensor, MQTT protocol for web interface, and relays to regulate an AC fan. Our aim was to achieve optimal energy efficiency while maintaining precise temperature control.

Results

- **Temperature Sensing and Control:** The I2C sensor accurately measured ambient temperature, with minimal deviation. The ESP32-controlled algorithm promptly regulated temperature by activating the AC fan through relays, showing swift response to fluctuations.
- **Energy Efficiency Analysis:** Leveraging MQTT and intelligent algorithms, our system minimized energy usage while ensuring optimal conditions. Remote monitoring and sleep modes of ESP32 further enhanced efficiency during idle periods.
- **Reliability and Performance:** The system displayed high reliability and performance. MQTT integration enabled seamless communication, while relays provided robust fan control. It met design specifications and remained stable under varied conditions.

Discussion

Optimization for Energy Efficiency: Energy-efficient strategies, like MQTT and intelligent control, were pivotal. Future enhancements could focus on refining algorithms and incorporating advanced power management.

Scalability and Adaptability: Our system's modular design allows easy expansion for multi-zone control or integration with smart home devices. MQTT's flexibility enables seamless integration with existing IoT platforms.

Functionality Testing: Testing validated sensor accuracy, fan control algorithms, and web interface responsiveness, all demonstrating consistent and reliable performance.

Challenges and Future Directions: Challenges included protocol optimization and compatibility issues, addressed through iterative testing. Future directions could involve machine learning for predictive control, renewable energy integration, and user-friendly interface development.

This project showcases the effectiveness of energy-efficient temperature control systems and offers avenues for further innovation and improvement.

Conclusion

In conclusion, the ESP32 temperature control project stands as a testament to the seamless integration of cutting-edge technology and sustainable practices. Through meticulous optimization of both software and hardware components, the project exemplifies how effective energy efficiency can be achieved in temperature control systems. Leveraging the ESP32 microcontroller's capabilities, the project harnesses low-power modes and implements efficient coding practices to ensure minimal power consumption without compromising on performance. This strategic utilization of resources not only underscores the project's commitment to environmental sustainability but also enhances its overall functionality and longevity.

Furthermore, the project's emphasis on precision sensor readings and intelligent control algorithms demonstrates a holistic approach to temperature regulation. By continuously monitoring environmental conditions and dynamically adjusting heating or cooling operations, the system effectively maintains a comfortable temperature while conserving energy. This not only benefits end-users by providing a seamless and energy-efficient solution for temperature control applications but also contributes to broader sustainability efforts.

With extended battery life and reduced energy consumption, the ESP32 temperature control project emerges as a practical and eco-friendly choice for various environments, from residential homes to commercial establishments. Its success underscores the importance of prioritizing energy efficiency in IoT applications and serves as a compelling example of how innovative design practices can lead to both environmental stewardship and economic viability.

Future Scope

i). **Enhanced User Interface:** Further development of the user interface to include advanced features such as data visualization, historical temperature trends, & personalized settings customization. This would provide users with more insights into temperature control & enhance their overall experience.

ii). **Integration of Machine Learning:** Exploration of machine learning algorithms to predict temperature variations based on historical data & user behavior patterns. This predictive capability could further optimize energy consumption & improve system's efficiency over time.

iii). **Multi-Zone Temperature Control:** Expansion of the project to support multi-zone temperature control, allowing users to regulate temperatures in different areas or rooms independently. This would cater to more complex environments, such as large buildings or smart homes, and offer greater flexibility and customization options.

iv). **Integration with Smart Home Systems:** Integration with existing smart home systems and platforms, such as Amazon Alexa or Google Home, to enable voice-controlled temperature adjustments and seamless integration with other smart devices in the home ecosystem.

v). **Remote Diagnostics and Maintenance:** Implementation of remote diagnostics and maintenance capabilities to detect and troubleshoot issues remotely, reducing downtime and maintenance costs. This could include automated alerts for system malfunctions or predictive maintenance based on performance analytics.

vi). **Energy Harvesting Solutions:** Exploration of energy harvesting solutions, such as solar panels or kinetic energy harvesters, to power the ESP32 temperature control system using renewable energy sources. This would further enhance the project's sustainability and reduce reliance on external power sources.

vii). **Deployment in Industrial Settings:** Adaptation of the project for deployment in industrial settings, such as manufacturing plants or warehouses, where precise temperature control is critical for product quality and process efficiency. This would involve robust industrial-grade hardware and tailored control algorithms to meet specific industry requirements.

viii). **Research on Alternative Cooling Methods:** Research into alternative cooling methods, such as passive cooling or phase-change materials, to supplement or replace traditional active cooling systems. Exploring innovative cooling solutions could further reduce energy consumption and environmental impact.

ix). **Community Engagement and Education:** Engaging with the community through workshops, seminars, & educational materials to raise awareness about energy-efficient temperature control practices & promote sustainability. Encouraging collaboration and knowledge-sharing among stakeholders could foster innovation and drive further advancements in field.

By pursuing these future scope initiatives, the ESP32 temperature control project can continue to evolve and address emerging challenges in energy efficiency, sustainability, and user experience, ultimately contributing to a more efficient, sustainable, and comfortable future

References

1. https://www.mdpi.com/1424-8220/23/15/6739
2. https://docs.espressif.com/projects/esp-idf/en/latest/esp32s2/api-reference/peripherals/temp_sensor.html#:~:text=The%20ESP32%2DS2%20has%20a,compensate%20for%20the%20temperature%20measurement.
3. https://www.amazon.in/Electronics-ESP32-Controller-Electronic-WiFi/dp/B0BXP671TL
4. https://www.leipole.net/article/dc-fan-and-its-working-principle.html#:~:text=News,DC%20fan%20and%20its%20working%20principle,that%20the%20fan%20blades%20rotate.
5. https://www.globalspec.com/learnmore/industrial_computers_embedded_computer_components/industrial_computing/relay_boards#:~:text=Relay%20boards%20are%20computer%20boards,of%20several%20onboard%20relay%20channels.
6. "ESP32 Projects Using the ESP32 IoT WiFi BLE Module" by Agus Kurniawan-This book covers various ESP32 projects, including IoT applications and energy-efficient designs, which could provide valuable insights for your project.
7. "Practical Internet of Things with ESP32: Build and Deploy Advanced IoT Projects Using the ESP32" by Agus Kurniawan-This book delves into advanced IoT projects using ESP32, including remote monitoring and energy optimization techniques, aligning closely with your project goals.
8. "Internet of Things with ESP32" by Mauro Califano-This book offers a comprehensive guide to building IoT applications with ESP32, exploring topics such as connectivity, sensors, and energy-efficient design principles.
9. "Building Smart Drones with ESP8266 and Arduino" by Syed Omar Faruk Towaha-Although focused on drones, this book provides valuable insights into IoT, wireless communication, and energy-efficient design practices applicable to your project.
10. Designing Embedded Systems with ESP32" by Yuan, Fan-This book covers fundamental principles of designing embedded systems with ESP32, which can be helpful for understanding hardware integration and optimization for energy efficiency.