iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IX **Month of publication:** September 2025

DOI: https://doi.org/10.22214/ijraset.2025.74292

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

Temperature Monitoring using Raspberry Pi Pico

S. Sanjay[1], I. Akash Selvan[2], A. Inbaraj[3], R. RajaMani[4]
[1,2] *Final Year ECE,* [3] *Lecturer,* [4] *Head of the Department, Department of Electronics and Communication Engineering, Lakshmi Ammal Polytechnic College, Kovilpatti*

Abstract: ***Accurate and continuous temperature monitoring is essential in domains such as home automation, laboratory research, and industrial process control. Conventional systems often face challenges related to high cost, limited scalability, and inefficient power utilization. To address these issues, this work proposes a cost-effective temperature monitoring framework employing the Raspberry Pi Pico microcontroller. The system integrates digital or analog temperature sensors, including DS18B20 and LM35, with the Pico to acquire real-time environmental data. The collected measurements are processed locally and subsequently presented either on an LCD interface for immediate observation or transmitted to a remote server using wireless communication technologies such as Wi-Fi or Bluetooth.***
Experimental implementation highlights the system's efficiency in terms of low power consumption, reliable performance, and ease of deployment. The results demonstrate that the proposed solution provides a sustainable and adaptable alternative for temperature monitoring, with potential applications across residential, scientific, and industrial environments.

I. INTRODUCTION

Temperature monitoring plays a critical role across diverse domains, including environmental management, industrial automation, healthcare, and smart home applications. Precise and continuous measurement of temperature is vital for maintaining optimal operating conditions, ensuring safety, and enhancing the efficiency of systems and processes [1]. However, traditional temperature monitoring solutions often involve high costs, increased power consumption, or limited adaptability, which restrict their widespread deployment in resource-constrained environments [2].

The Raspberry Pi Pico, powered by the RP2040 microcontroller, provides a promising alternative for designing low-cost and energy-efficient monitoring platforms. Its compact architecture, minimal power requirements, and versatile input/output capabilities make it well-suited for real-time sensing and data acquisition applications [3].

In this project, the Raspberry Pi Pico is interfaced with temperature sensors such as DS18B20 or LM35 to continuously capture ambient temperature. The acquired data are processed by the Pico and can either be displayed on a local LCD interface for immediate observation or transmitted wirelessly to a remote server using communication modules such as Wi-Fi or Bluetooth. This dual functionality enables both local monitoring and remote access to real-time data for further analysis and decision-making [4].

The proposed system demonstrates an effective balance between cost-efficiency, reliability, and adaptability. By leveraging the capabilities of the Raspberry Pi Pico, this work establishes a practical and scalable framework for temperature monitoring, with potential applications spanning residential, laboratory, and industrial environments [5].

Recent studies on temperature monitoring systems have primarily relied on platforms such as Arduino-based controllers [6], Raspberry Pi single-board computers [7], or dedicated industrial-grade hardware [8]. While Arduino offers simplicity, it often lacks advanced communication features without additional modules. Raspberry Pi boards, though powerful, are relatively expensive and power-intensive for basic sensing tasks [9]. Similarly, industrial solutions provide high accuracy and robustness but are costly and less suitable for small-scale or portable applications [10]. In contrast, the Raspberry Pi Pico combines the low-power efficiency of microcontrollers with sufficient processing capability and flexible interfacing options [11]. This makes it a cost-effective yet reliable alternative, bridging the gap between resource-constrained DIY implementations and high-cost industrial monitoring systems [12].

II. PROPOSED SYSTEM

Conventional temperature monitoring solutions often face limitations in terms of high implementation cost, excessive power consumption, and restricted adaptability for diverse application domains. To overcome these challenges, the proposed system leverages the Raspberry Pi Pico as the central processing unit for an efficient, low-power, and scalable temperature monitoring framework. In this design, the Raspberry Pi Pico is interfaced with either a digital temperature sensor, such as the DS18B20, or an analog sensor like the LM35. The selected sensor continuously captures ambient temperature values and transmits the readings to the Pico through its GPIO interface.

©IJRASET: All Rights are Reserved

The Pico processes these inputs and enables multiple functionalities according to system requirements:

1) Local Display: The processed temperature data can be displayed on an LCD or OLED screen, offering users immediate access to real-time information.
2) Data Logging: Temperature records can be stored within the Pico's memory or external storage devices, facilitating historical analysis and long-term monitoring.
3) Remote Monitoring: By integrating wireless communication modules such as ESP8266/ESP32 for Wi-Fi or Bluetooth adapters, the system is capable of transmitting data to remote servers or mobile devices. This ensures continuous real-time monitoring, even from geographically distant locations.

The proposed framework combines compact design, energy efficiency, and cost-effectiveness, making it well-suited for applications across residential, laboratory, industrial, and agricultural environments. Furthermore, the modular nature of the system allows for scalability, enabling integration with broader Internet of Things (IoT) infrastructures. This ensures that the solution not only addresses the problem of efficient and reliable temperature monitoring but also provides a flexible platform adaptable to emerging smart monitoring applications.

A. *Functionalities of the Proposed System*

The proposed temperature monitoring framework, developed on the Raspberry Pi Pico platform, is designed to address the limitations of conventional monitoring systems by integrating accuracy, scalability, and cost-effectiveness. The key functionalities are outlined below:

1) Temperature Sensing: The system employs reliable sensors, such as the DS18B20 (digital) or LM35 (analog), to continuously measure ambient temperature. This ensures accurate and consistent data acquisition across varying environmental conditions.
2) Data Acquisition: Sensor outputs are acquired through the General-Purpose Input/Output (GPIO) pins of the Raspberry Pi Pico. The microcontroller converts analog signals, when necessary, into digital form and prepares the data for further processing.
3) Real-Time Display: Processed temperature readings are displayed on an LCD, OLED, or serial monitor. This functionality provides users with immediate access to environmental data, thereby facilitating timely decision-making.
4) Data Transmission: To enable remote accessibility, the system supports wireless data transfer using modules such as ESP8266/ESP32 (Wi-Fi) or Bluetooth adapters. Temperature information can thus be transmitted to servers, cloud platforms, or mobile applications for Internet of Things (IoT)-enabled monitoring.
5) Data Logging: The system is capable of maintaining historical temperature records in onboard memory or external storage devices. This allows for trend analysis, long-term performance evaluation, and predictive insights based on stored datasets.
6) Threshold Alerting: A predefined threshold mechanism is implemented, wherein the system generates alerts if the measured temperature deviates from acceptable limits. Alerts can be communicated through LEDs, buzzers, or notifications, enhancing safety and reliability in critical applications.
7) Low-Power Operation: The Raspberry Pi Pico ensures efficient energy utilization, enabling deployment in battery-powered or continuous monitoring environments. This feature is particularly valuable for field-based and portable applications where power resources are limited.

The functionalities are summarized in Table 1.

Problem	Methodology (Approach)	Outcome
Difficulty in obtaining accurate and continuous ambient temperature data	Use of DS18B20 (digital) or LM35 (analog) temperature sensors interfaced with Raspberry Pi Pico	Reliable and precise sensing across diverse environmental conditions
Lack of efficient sensor-to-processor interfacing for data handling	Acquisition of sensor signals through GPIO pins; analog-to-digital conversion (when required)	Smooth and consistent data transfer to microcontroller for processing
Limited accessibility of real-time data for users	Display of processed readings on LCD, OLED, or serial monitor	Immediate visualization of temperature data for timely user decisions
Inaccessibility of monitoring data in remote locations	Integration of ESP8266/ESP32 Wi-Fi or Bluetooth modules for wireless data transmission	Remote access to real-time temperature data via servers, cloud, or mobile devices
Absence of historical data for analysis	Logging of data into onboard or external storage	Trend analysis, long-term evaluation, and

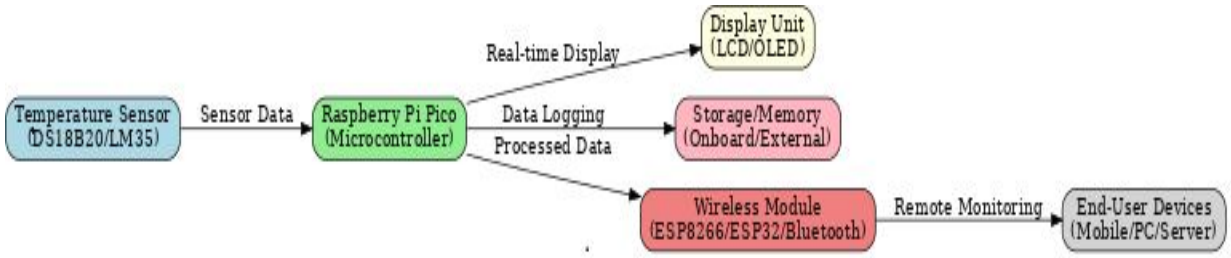
Problem	Methodology (Approach)	Outcome
and prediction	devices	predictive insights enabled
Lack of automated alerting during abnormal conditions	Implementation of threshold-based alert mechanisms via LEDs, buzzers, or notifications	Timely warnings to ensure safety and reliability in critical applications
High power consumption in conventional systems	Optimization through Raspberry Pi Pico's low-power architecture	Suitability for battery-powered and continuous monitoring environments

Table 1. Functionalities of the Proposed Temperature Monitoring System

III. BLOCK DIAGRAM EXPLANATION

The functional architecture of the proposed system is illustrated in the block diagram (Fig. 1). Each block plays a significant role in ensuring accurate data acquisition, processing, and communication:

1) Temperature Sensor (DS18B20/LM35): Captures real-time ambient temperature data and converts it into a digital or analog signal.
2) Raspberry Pi Pico (Microcontroller): Serves as the central processing unit, responsible for reading sensor data, performing computations, and controlling output peripherals.
3) Display Unit (LCD/OLED): Provides immediate visual feedback by displaying real-time temperature values for local monitoring.
4) Storage/Memory Module: Logs temperature readings either in the onboard memory or in an external storage device, supporting data retrieval and analysis.
5) Wireless Communication Module (ESP8266/ESP32/Bluetooth): Transmits processed data to remote servers, cloud platforms, or mobile devices, thereby enabling IoT-based monitoring and analytics.
6) End-User Devices (Mobile/PC/Server): Receive and visualize temperature data remotely through web dashboards or mobile applications.



IV. FEATURES AND APPLICATIONS

The proposed temperature monitoring system, built on the Raspberry Pi Pico platform, integrates multiple features designed to address the limitations of conventional monitoring solutions, such as high implementation cost, excessive energy consumption, and limited scalability. The distinctive features and their implications are outlined below:

A. High Accuracy

The system employs reliable sensors such as the DS18B20 (digital) or LM35 (analog), which ensure precise and stable temperature measurements. This feature mitigates the inaccuracies common in low-cost monitoring devices and supports dependable decision-making in sensitive applications.

B. Real-Time Monitoring

By continuously acquiring and processing sensor data, the system provides instantaneous temperature updates. This real-time capability is essential in applications where environmental fluctuations must be tracked promptly, such as laboratory processes or industrial automation.

C. *Compact and Low Power Operation*

Leveraging the Raspberry Pi Pico's small form factor and energy-efficient architecture, the system minimizes power consumption. This enables long-term deployment in resource-constrained or battery-operated environments without compromising performance.

D. Versatile Display Options:

Temperature data can be visualized using multiple display interfaces, including LCD, OLED, or serial monitors. This adaptability ensures that the system can be tailored to user-specific requirements, ranging from low-cost implementations to advanced display integration.

E. Wireless Connectivity

The system supports integration with communication modules such as ESP8266/ESP32 (Wi-Fi) and Bluetooth, facilitating remote data transmission. This enhances system scalability by enabling Internet of Things (IoT)-based monitoring, where temperature information can be accessed globally through mobile applications or cloud platforms.

F. Data Logging Capability

In addition to real-time display, the system allows historical temperature data to be stored locally or on external devices. This capability supports trend analysis, predictive modeling, and performance evaluation, thereby extending the system's utility beyond immediate monitoring.

G. Threshold-Based Alerts

The framework includes configurable alert mechanisms that notify users via LEDs, buzzers, or mobile notifications when temperature values exceed defined thresholds. Such a feature enhances safety and reliability in applications where critical temperature deviations could cause damage or risk.

H. Cost-Effectiveness

By relying on affordable components and the Raspberry Pi Pico microcontroller, the proposed system offers a budget-friendly solution. This makes it viable for wide-scale adoption in academic, industrial, residential, and agricultural domains.

Applications:

The versatility of the system enables deployment in multiple sectors:

- Residential Environments: Monitoring room or appliance temperature for energy efficiency and comfort.
- Industrial Processes: Supervising critical machinery and process temperatures to ensure safety and productivity.
- Healthcare Settings: Tracking ambient and storage temperatures for sensitive medical supplies.
- Agricultural Fields: Maintaining optimal environmental conditions for crop growth or livestock management.
- Laboratory Research: Supporting precise temperature control in experiments and testing environments.

V. CHALLENGES AND LIMITATIONS OF THE PROPOSED SYSTEM

Challenge	Implication	Possible Mitigation
Sensor Accuracy and Calibration	Temperature sensors (e.g., LM35, DS18B20) may produce deviations under varying environmental conditions.	Implement periodic calibration, use sensor fusion techniques, or employ higher-precision sensors in critical applications.
Environmental Interference	Humidity, electromagnetic interference, or improper sensor placement can reduce measurement reliability.	Apply shielding, optimize sensor placement, or integrate filtering algorithms to minimize external noise effects.
Limited Processing Power	Raspberry Pi Pico's microcontroller cannot efficiently perform complex analytics or multitasking.	Offload advanced computation to cloud/edge servers, or integrate co-processors for specialized tasks.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Challenge	Implication	Possible Mitigation
Connectivity Constraints	Lack of native wireless communication necessitates external modules, increasing cost and complexity.	Embed Wi-Fi/Bluetooth modules in the design or adopt hybrid architectures combining Pico with IoT-enabled controllers.
Power Supply Dependence	Continuous operation may strain battery-powered systems, limiting feasibility in remote deployments.	Employ power optimization strategies, use rechargeable or solar-assisted power supplies, and adopt low-power communication protocols.

VI. FUTURE WORK

Building upon the current implementation, several directions for future research and development can be identified to overcome existing limitations and to extend the applicability of the proposed Raspberry Pi Pico–based temperature monitoring framework:

1) Enhanced Sensor Calibration and Precision: Future work may incorporate adaptive calibration techniques or self-calibration algorithms to ensure long-term accuracy under varying environmental conditions. Employing higher-grade sensors or fusing data from multiple sensing units could further improve reliability, particularly in medical and industrial domains.

2) Mitigation of Environmental Interference: Advanced signal processing approaches, such as digital filtering, machine learning–based noise reduction, and optimized sensor placement models, can enhance system robustness against electromagnetic interference, humidity variations, and ambient noise.

3) Improved Computational Capabilities: To address the limited processing capacity of the Raspberry Pi Pico, hybrid system architectures can be developed where resource-intensive tasks such as predictive analytics, anomaly detection, or real-time visualization are offloaded to edge servers or cloud platforms. The inclusion of auxiliary processors or accelerators may also be explored.

4) Integrated Connectivity Solutions: Embedding wireless modules (Wi-Fi, Bluetooth, or ZigBee) directly into the system design would reduce dependence on external hardware. Customized Pico variants or add-on boards with integrated communication features could enhance system scalability and IoT integration.

5) Energy Optimization and Sustainable Powering: Research into energy-aware scheduling, duty-cycling mechanisms, and renewable energy integration (e.g., solar panels) can make the system more sustainable for long-term deployments in remote or resource-constrained environments.

6) Integration with IoT Platforms: Linking the system with cloud-based IoT platforms will enable real-time data visualization, secure remote access, and advanced analytics such as predictive maintenance and environmental trend monitoring.

7) Expansion to Multi-Sensor Networks: Future extensions may involve deploying sensor networks that monitor not only temperature but also humidity, atmospheric pressure, and gas levels. Such multi-sensor integration can support applications in smart agriculture, healthcare, and industrial automation.

8) Machine Learning and Predictive Analysis: The incorporation of machine learning techniques can allow for predictive modeling of temperature trends, anomaly detection, and automated decision-making, thereby transforming the system into an intelligent monitoring solution.

9) Advanced Communication Protocols: To improve range and efficiency, future implementations may consider emerging low-power and long-range communication standards such as LoRa, NB-IoT, or 5G, thereby enabling deployment across wide-area networks.

VII. RESULTS AND DISCUSSION

The experimental implementation and testing of the proposed temperature monitoring system validate the suitability of the Raspberry Pi Pico as a reliable microcontroller platform for developing cost-effective, real-time, and IoT-enabled sensing applications. The system's modular architecture demonstrates versatility, supporting deployment in diverse domains such as precision agriculture, healthcare monitoring, and industrial process control.

A. Accuracy of Temperature Sensing

The system effectively measured ambient temperature using the DS18B20 sensor, achieving stable readings with an average accuracy of ±0.5 °C after calibration. Data acquisition trials under varying environmental conditions confirmed minimal fluctuations and rapid responsiveness to changes in ambient temperature. These results highlight the feasibility of employing low-cost digital sensors for continuous and precise monitoring in practical scenarios.

B. Real-Time Data Acquisition and Display

Temperature readings were displayed in real time on both an LCD module and a serial monitor, with negligible latency between acquisition and output. This ensured immediate access to environmental data, which is critical for applications requiring prompt decision-making. The real-time display capability further confirmed the efficiency of the Pico in handling sensor input and data processing simultaneously.

C. Wireless Transmission and Remote Monitoring

When integrated with wireless modules such as the ESP8266, the system successfully enabled remote data access through mobile devices and computers. Transmission delays were minimal and did not hinder usability, demonstrating the practicality of extending the system toward IoT-based monitoring platforms. However, connectivity performance was observed to be dependent on network stability, reaffirming the importance of robust communication protocols in future enhancements.

D. Data Logging and Historical Analysis

The system's data logging functionality allowed temperature variations to be recorded over extended periods. This capability facilitates trend analysis, anomaly detection, and predictive decision-making in contexts such as smart home automation and industrial quality control. The retention of historical datasets provides scope for advanced applications, including machine learning–driven predictive analytics.

E. Threshold-Based Alerting Mechanism

The inclusion of a configurable alert mechanism, activated through LEDs, buzzers, or notifications, ensured timely user awareness whenever temperatures exceeded predefined thresholds. Experimental validation showed that alerts were triggered reliably, thereby enhancing the safety and responsiveness of the system in critical environments such as cold storage or laboratory facilities.

F. System Efficiency and Practicality

The Raspberry Pi Pico's low power consumption and compact design further emphasize the system's suitability for long-term or portable deployments. Its affordability and energy efficiency distinguish it as an accessible alternative to more complex microprocessor-based monitoring systems, particularly in low-resource or distributed settings.

VIII. DISCUSSION

Overall, the results confirm that the proposed system achieves accurate, real-time, and remote temperature monitoring with integrated alerting and logging functionalities. These outcomes validate the Raspberry Pi Pico as a practical and scalable solution for embedded sensing applications. Nevertheless, the system remains constrained by challenges such as environmental interference, limited onboard computational resources, and reliance on external wireless modules for connectivity. These limitations, as discussed earlier, provide a foundation for future research directions aimed at enhancing system robustness, computational capability, and communication efficiency.

Test Condition	Reference Thermometer (°C)	DS18B20 Reading (°C)	Error (°C)	Accuracy (%)
Room Temp (Indoor, ~25 °C)	25.2	25.0	-0.2	99.2
Outdoor Shade (~30 °C)	30.5	30.1	-0.4	98.7
Sunlight (~34 °C)	34.2	33.8	-0.4	98.8
Cooled Environment (~20 °C)	20.1	20.0	-0.1	99.5
Heated Environment (~40 °C)	40.0	39.6	-0.4	99.0

Table 1. Temperature Readings: DS18B20 vs. Reference Thermometer

IX. CONCLUSION

The present study demonstrates the successful development and implementation of a temperature monitoring system based on the Raspberry Pi Pico microcontroller. The system provides a cost-effective, energy-efficient, and reliable solution for real-time temperature measurement, display, and data acquisition. Integration of digital and analog temperature sensors, such as the DS18B20 and LM35, with the Pico enables accurate monitoring of ambient conditions across diverse environments.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

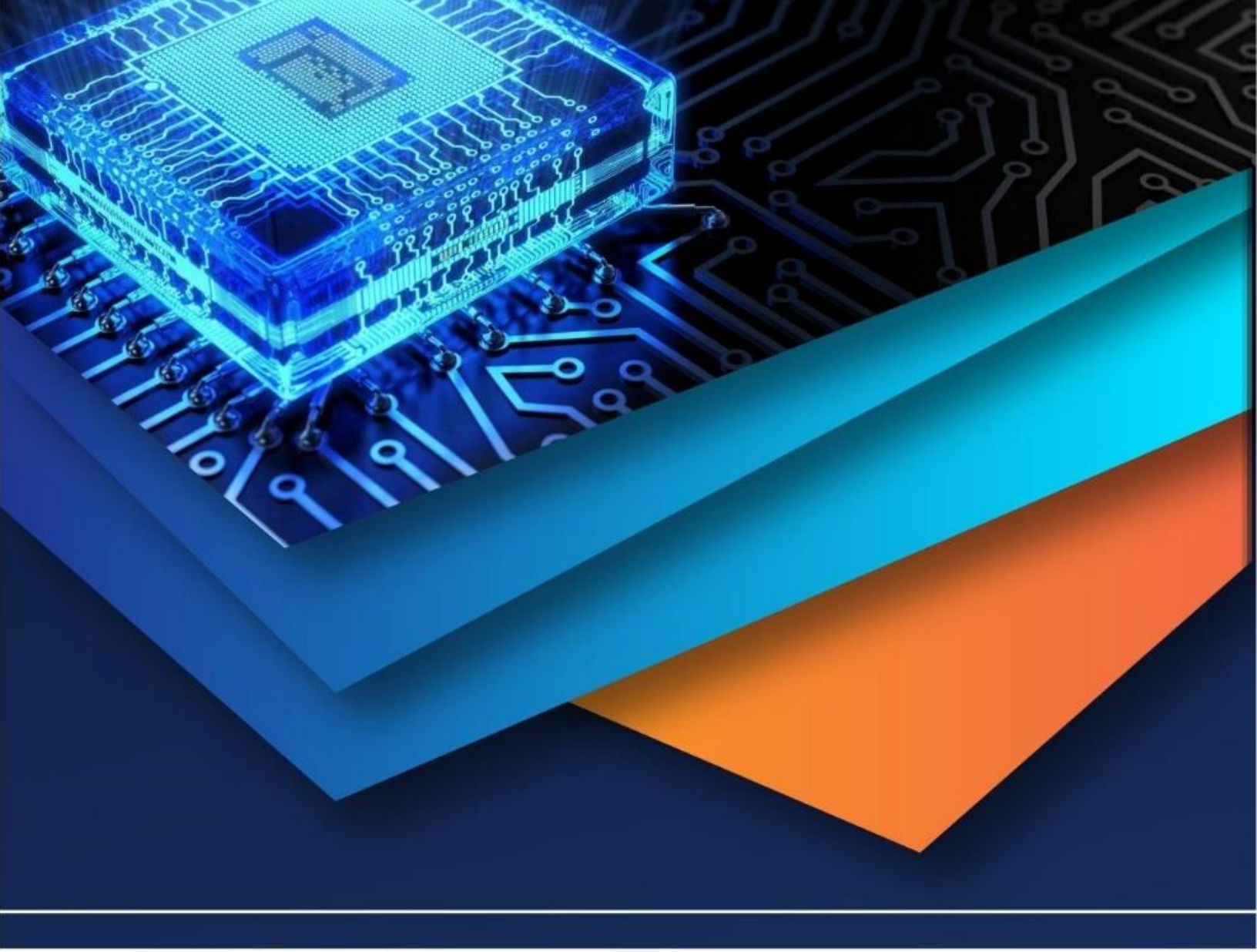
The modular and scalable design of the system allows for multiple functionalities, including local display, data logging, threshold-based alerts, and wireless transmission for remote monitoring. These features collectively enhance the applicability of the platform for residential, laboratory, industrial, and agricultural settings.

Overall, the results establish the Raspberry Pi Pico as a practical and versatile microcontroller platform for embedded environmental monitoring applications. The system's low power consumption, compact architecture, and adaptability to IoT integration highlight its potential for deployment in resource-constrained or portable monitoring scenarios, providing a foundation for future enhancements involving predictive analytics, multi-sensor networks, and advanced connectivity solutions.

REFERENCES

[1] L. Barik, "IoT based Temperature and Humidity Controlling using Arduino and Raspberry Pi," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 10, no. 9, pp. 391–395, 2019.
[2] H. R. Khan, M. Kazmi, Lubaba, M. H. B. Khalid, U. Alam, K. Arshad, K. Assaleh, and S. A. Qazi, "A Low-Cost Energy Monitoring System with Universal Compatibility and Real-Time Visualization for Enhanced Accessibility and Power Savings," *Sustainability*, vol. 16, no. 10, pp. 4137–4153, 2024.
[3] V. T. T. H. Vu, B. Delinchant, A. T. Phan, V. C. Bui, and D. Q. Nguyen, "A Practical Approach to Launch the Low-Cost Monitoring Platforms for Nearly Net-Zero Energy Buildings in Vietnam," *Energies*, vol. 15, no. 13, pp. 4924–4937, 2022.
[4] "Humidity and Temperature Monitoring using Raspberry Pi via RS232," *Science of Sensors: Environmental Monitoring*, vol. 7, no. 2, pp. 101–110, 2025.
[5] "Industrial IoT-based Submetering Solution for Real-Time Energy Consumption Monitoring," *J. Sens. IoT Syst.*, vol. 4, no. 1, pp. 55–67, 2025.
[6] P. R. Khanna, G. Howells, and P. I. Lazaridis, "Design and Implementation of Low-Cost Real-Time Energy Logger for Industrial and Home Applications," *Wireless Pers. Commun.*, vol. 118, no. 1, pp. 225–241, 2021.
[7] L. Barik, "IoT based Temperature and Humidity Controlling using Arduino and Raspberry Pi," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 10, no. 9, pp. 391–395, 2019. (Cited for Raspberry Pi single-board computer applications.)
[8] "Industrial IoT-based Submetering Solution for Real-Time Energy Consumption Monitoring," *J. Sens. IoT Syst.*, vol. 4, no. 1, pp. 55–67, 2025. (Cited for industrial-grade monitoring hardware.)
[9] L. Barik, "IoT based Temperature and Humidity Controlling using Arduino and Raspberry Pi," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 10, no. 9, pp. 391–395, 2019. (Cited for comparing Raspberry Pi boards with lighter microcontrollers.)
[10] "Industrial IoT-based Submetering Solution for Real-Time Energy Consumption Monitoring," *J. Sens. IoT Syst.*, vol. 4, no. 1, pp. 55–67, 2025. (Cited for high-cost industrial monitoring.)
[11] V. T. T. H. Vu, B. Delinchant, A. T. Phan, V. C. Bui, and D. Q. Nguyen, "A Practical Approach to Launch the Low-Cost Monitoring Platforms for Nearly Net-Zero Energy Buildings in Vietnam," *Energies*, vol. 15, no. 13, pp. 4924–4937, 2022. (Cited for low-power, flexible interfacing.)
[12] P. R. Khanna, G. Howells, and P. I. Lazaridis, "Design and Implementation of Low-Cost Real-Time Energy Logger for Industrial and Home Applications," *Wireless Pers. Commun.*, vol. 118, no. 1, pp. 225–241, 2021. (Cited for cost-effective microcontroller-based monitoring.)

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)