International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

4G Enabled IoT Monitoring

[1]Janani H, Anusha D M, M Grace, [2]Gouri M S

T John Institute of Technology

ABSTRACT

This project addresses connectivity challenges in industrial settings by integrating an ESP32 microcontroller with a GSM module for 4G internet access. This setup enables wireless communication with cloud services, ensuring reliable data transmission in remote or network- constrained environments. A demonstration application will validate the solution by transmitting sensor data to a cloud platform, showcasing its practicality for industrial IoT applications and remote monitoring.

Keywords: GSM, ESP32, AT command library, Network tools.

1. INTRODUCTION

In today's industrial environments, reliable data communication is crucial for monitoring and controlling various processes remotely. However, many industrial setups face challenges due to the unavailability or unreliability of traditional networks like Wi-Fi or Ethernet, especially in remote or mobile locations. To address this issue, cellular networks such as 4G provide an alternative solution for uninterrupted connectivity.

This project "4G-Enabled IoT Monitoring System" aims to bridge this gap by integrating an ESP32 microcontroller with a GSM/4G module. The system is designed to collect sensor data from industrial equipment and transmit it to a cloud platform over a 4G connection. This ensures continuous monitoring and control even in environments where conventional networks are not feasible.

The project also includes a demonstration of a small application that showcases real-time data transfer, monitoring, and remote-control capabilities. By leveraging 4G technology, the system provides a robust and scalable solution for industries that require reliable communication for IoT devices, enabling better efficiency and oversight in various industrial processes.

1.1 MOTIVATION

The project is motivated by the need for reliable connectivity in industrial environments where Wi- Fi or Ethernet is unavailable. By integrating an ESP32 microcontroller with a GSM module for 4G connectivity, the solution ensures continuous sensor data transmission to a cloud platform. This enables real-time monitoring and control, overcoming connectivity challenges and supporting remote industrial operations efficiently.

1.2 PROBLEM STATEMENT

In industrial environments where traditional Wi-Fi or Ethernet networks are unavailable or unreliable, devices need an alternative way to communicate with the cloud or remote servers. To solve this, an ESP32 microcontroller must be integrated with a GSM module to facilitate a 4G internet connection for data communication. The ESP32 should be able to send and receive data via 4G, ensuring continuous connectivity for remote monitoring and control. A small application will be developed to demonstrate the 4G connection by sending sensor data to a cloud platform, confirming that data communication is working correctly.

1.3 OBJECTIVES

The basic objectives of the project include the following This system ensures reliable connectivity in industrial environments by using an ESP32 microcontroller and a GSM module to provide continuous 4G internet access where Wi-Fi or Ethernet is unavailable. Designed for industrial applications, it enables seamless, real-time sensor data transmission to a cloud platform for ongoing.

The ESP32 collects data from sensors like temperature, humidity, or pressure, transmitting it over the 4G network to ensure constant access to critical information, supporting immediate response and decision-making. A small-scale application will demonstrate the system's capabilities, showcasing how it efficiently handles and transmits sensor data to enable real-time monitoring in simulated industrial conditions. This solution is cost-effective and scalable, providing an affordable alternative to traditional networks. With low-cost components and cloud integration, it reduces infrastructure expenses, supporting easy scaling across multiple remote sites, and enhancing industrial automation and operational control in locations.

2. Literature Survey

1. explains an automated environmental monitoring system built around a PIC16F877A microcontroller and LM35 temperature sensor, designed specifically for critical environments like server rooms and data centres where temperature stability is crucial for equipment longevity and performance. The system continuously monitors ambient temperature and triggers multiple alerts (LCD display, LED indicator, buzzer alarm, and SMS notification via GSM module) when temperature falls outside the programmed safe range of -10°C to 30°C, eliminating the need for constant manual monitoring. system was programmed using C language (version 1.05) and Arduino 1.0.5 software, with the code implementing high and low temperature sensitivity thresholds, alert system logic, and GSM communication protocols, all while maintaining real-time monitoring capabilities. Testing demonstrated the system's reliability across different times of day (with recorded temperatures of 21.45°C in morning, 28.38°C in afternoon, and 23.06°C at night), proving its effectiveness as a comprehensive solution for automated temperature monitoring in critical environments where maintaining stable temperatures is essential for operational integrity.

2. explains the development of applications using ESP32 microcontrollers, which are powerful System-on-Chip (SoC) devices featuring integrated Wi-Fi, Bluetooth, dual-core processors up to 240MHz, and various peripherals, making them suitable for IoT and smart home automation. Three main development platforms are discussed: Arduino Core for ESP32 (best for beginners and simple projects), Express if IoT Development Framework (for complex professional projects), and Micro Python (for Python-based development), each offering different advantages and complexity levels.

The researchers focused on implementing different display options with ESP32, testing various sizes including 0.91" OLED (128x32 pixels), 0.96" OLED (128x64 pixels), and 2.4" LCD (320x240

pixels), each connected through I2C or SPI interfaces to show status information and data visualization The study demonstrated that smaller displays (0.91" and 0.96") are best suited for showing basic status information like network connection and measurement cycles, while larger 2.4"displays can handle more complex visualizations including graphical waveforms of measured values. The research emphasizes the ESP32's versatility in environmental monitoring, IoT applications, and data measurement systems, with the choice of display and development platform depending on the specific requirements of the project and the developer's expertise level.

3. explains an IoT-enabled real-time machine status monitoring platform for Cloud Manufacturing (CMF), which emphasizes resource sharing and operational efficiency. IoT technologies like RFID and wireless communication capture machine statuses in real- time, with the information processed and visualized via cloud services accessible on smartphones. The paper outlines a hierarchical architecture featuring smart machines, smart connections, data models, and a smart view layer for end-user interaction. A case study is presented using a laboratory testbed with milling machines, robots, and IoT devices to demonstrate the system's effectiveness.

The solution addresses traditional challenges such as machine identification, data collection, and task tracking, while improving production planning and decision-making through real-time data visualization. The paper concludes with contributions to SOA-based CMF frameworks, IoT deployment strategies, and optimized production operations, suggesting future research in areas like behavioral modeling, production optimization, and data analytics.

4. explains the authors explore the impact of Wi- Fi technology on IoT devices across applications like smart cities and healthcare. They discuss RFID and WSN technologies for device tracking and data gathering, highlighting challenges such as energy consumption and communication interference. Examine how IoT advancements have improved electricity management for grid stations and consumers. Additionally, security issues related to IoT in smart grids are addressed. The analysis includes the advantages and limitations of wearable sensor devices.

5. explains the energy efficiency of LoRa-based networks, particularly using the LoRaWAN protocol, is extensively studied. A model detailed in examines how parameters like acknowledged transmission, spreading factor, and payload size affect sensor node energy consumption. An optimization study reveals a trade-off between communication range, spreading factor, and transmission power. With 7620 possible parameter combinations, efficient transmission parameter selection is crucial, though LoRa Wan’s static configurations limit flexibility. For high communication ranges, specific settings like 20 dBm transmission power and SF of 12 are recommended. Additionally, multi-hop LoRa networks have been explored using a modified TSCH MAC protocol, enhancing power efficiency through time division and channel hopping mechanisms. A proposed fog network architecture aims to improve connectivity and data transmission among IoT devices via.

3. PROPOSED SYSTEM

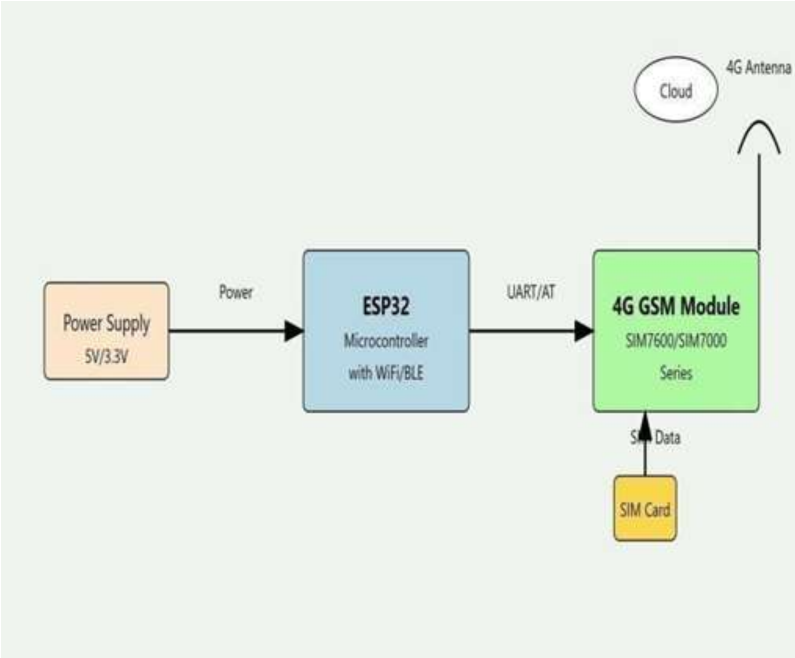


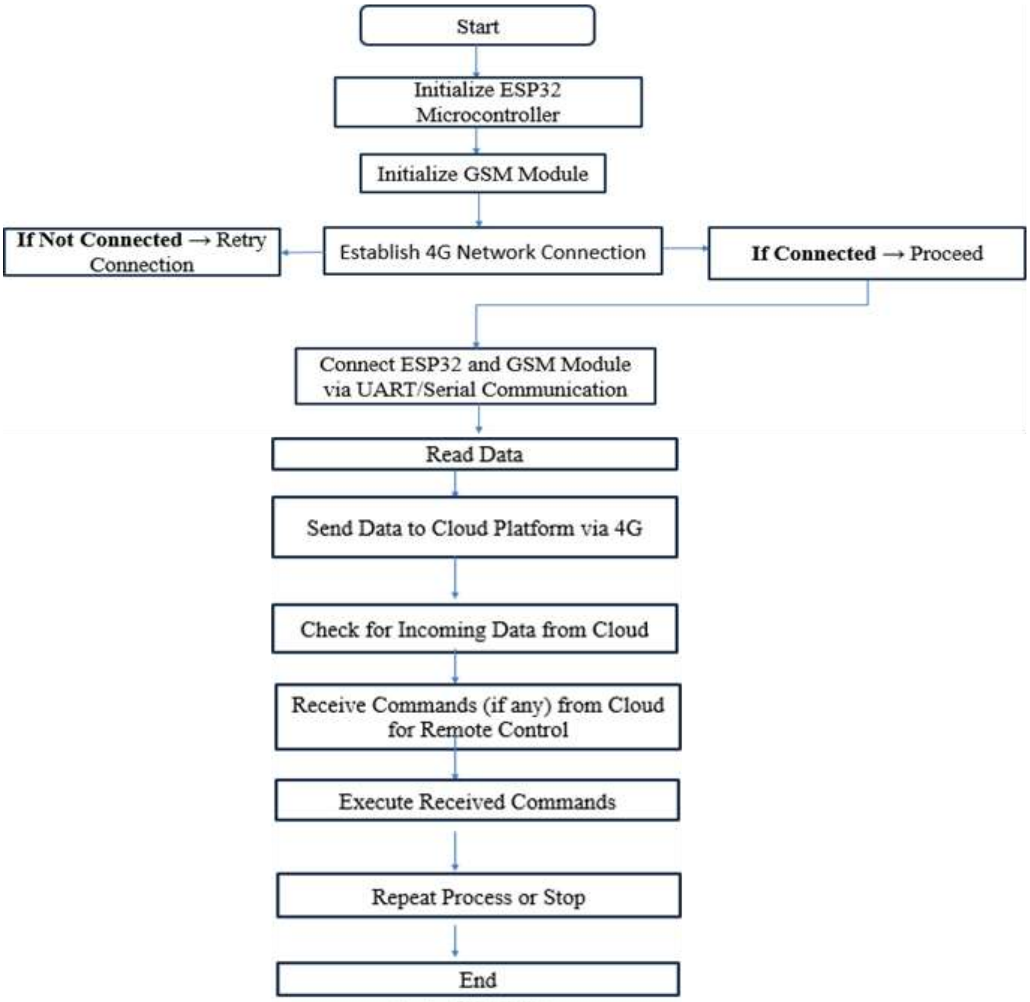
Figure: Block diagram

The diagram represents a setup designed for remote data transmission in industrial environments using an ESP32 microcontroller and a 4G GSM module. The system begins with a 5V/3.3V power supply that powers the ESP32, a microcontroller with built-in Wi-Fi and Bluetooth capabilities. However, instead of relying on Wi-Fi, the system uses a 4G GSM module (such as the SIM7600 or SIM7000 series) to achieve connectivity in areas where Wi- Fi or Ethernet is unavailable.

The ESP32 communicates with the GSM module through UART (Universal Asynchronous Receiver- Transmitter) using AT commands. A SIM card inserted in the GSM module provides cellular network access, enabling data transmission over a 4G network. The module is also connected to a 4G antenna to strengthen network reception. This configuration allows the ESP32 to reliably send sensor or device data to a cloud platform, supporting remote monitoring and control applications essential for industrial IoT solutions.

4. METHODOLOGY

The methodology flowchart outlines the steps to establish and maintain a connection between an ESP32 microcontroller and a 4G GSM module for data transmission to a cloud platform. The process begins with initializing the ESP32 and GSM module. The system then attempts to establish a 4G network connection. If successful, it proceeds; otherwise, it retries until connected. The ESP32 and GSM module communicate via UART/Serial.



Once connected, the ESP32 reads data from sensors or devices, then sends this data to the cloud. It also checks for incoming data or commands from the cloud for remote control. If any commands are received, the ESP32 executes them. This process continues in a loop, allowing for continuous data monitoring and control. The system either repeats this process or stops based on the application's requirements. This methodology enables remote monitoring and control in real-time using cellular connectivity.

5. Circuit and its explanation

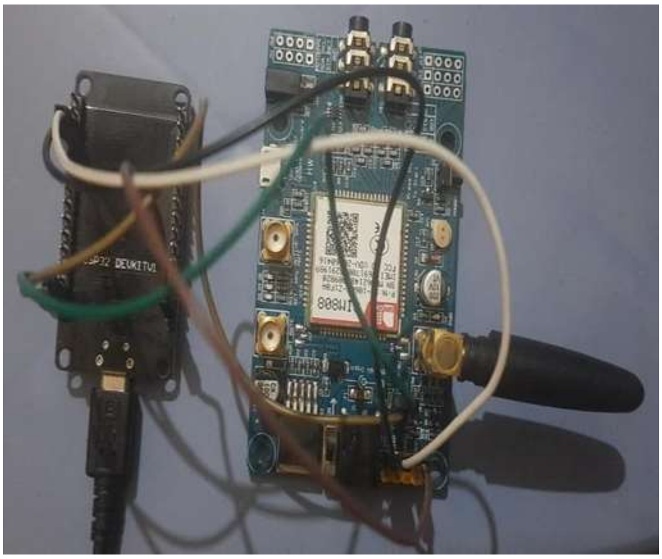


Figure: Circuit

Components in the Circuit:

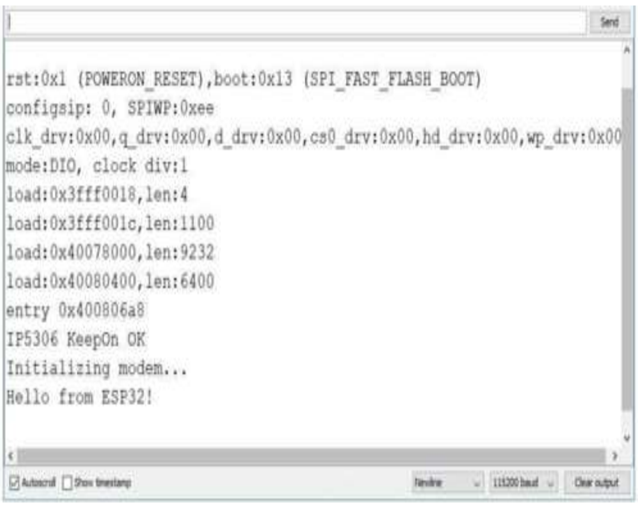
- ESP32
- GSM MODULE (SIM800L)
- SIM
- JUMPER WIRES
- POWER SUPPLY

Connections:

1. **ESP32 Pins:**
    - **3.3V Pin:** Supplies power to the sensor via its VDD pin.
    - **GND Pin:** Connected to the GND of the sensor.
2. **GSM Pins:**
    - **TXD Pin:** Connected to the ESP32 RXD Pin.
    - **RXD Pin:** Connected to the ESP32 TXD Pin.

- The DS18B20 measures the temperature and sends the data to the ESP32 through the data line.

6. Implementation:



After a few seconds, you should receive an SMS on the recipient's phone number.

Working:

- The ESP32 powers the DS18B20 sensor through its 3.3V and GND pins.

- The ESP32 communicates with the DS18B20 using GPIO 13 via the 1-Wire protocol.

7. OUTPUT

Data sent to the local server of the production:

- **IoT Machine Monitoring System:** The display shows a software interface for monitoring machines, tracking operations, and managing machine status in real time.
- **Tabs and Functionality:** The top of the interface has tabs like *Inspection, Production, Report, Maintenance,* and *Machine Idle* for various operational modes.
- **Network Status:** A status indicator at the top- right corner shows *Up* and *Down*, signifying the machine's connectivity status.
- **Manual Control Panel:** Below the monitor, a custom-built switch panel with four rotary knobs allows manual control of machine parameters or settings.
- **Workspace Setup:** Visible cables and an antenna suggest IoT connectivity, and the reflection of a person indicates an industrial or manufacturing environment.
- **Purpose:** The system facilitates real-time machine tracking, operator input, and status management to improve efficiency and maintenance.

CONCLUSION

In conclusion, the integration of the ESP32 microcontroller with a GSM module for 4G connectivity provides a robust solution for data communication in industrial environments lacking reliable Wi-Fi or Ethernet access. This project successfully demonstrates the capability of the ESP32 to transmit and receive sensor data via 4G, ensuring continuous connectivity for remote monitoring and control. By sending real-time data to a cloud platform, the application not only validates the functionality of the 4G connection but also showcases the potential for scalable and flexible monitoring solutions in diverse industrial applications. This implementation paves the way for enhanced operational efficiency and data accessibility, ultimately contributing to improved decision-making and management in challenging environments.

REFERENCE

[1] Development of a GSM-Based Environmental Monitoring System 1Oke A.O., +2Awokola J.A.,3Amusan E.A. 1,+2,3 Department of Computer Science and Engineering. Ladoke AkintolaUniversity of Technology, Ogbomoso, Oyo State,Nigeria

[2] https://www.researchgate.net/publication/3345727 15_Using_the_ESP32_Microcontroller_for_ Data_Processing

[3]https://www.researchgate.net/publication/318355493_An_IoT-enabled_Real-time_Machine_Status_Monitoring_Approach_for_Cloud_ Manufacturin

[4] https://www.mdpi.com/1533236

[5] https://pmc.ncbi.nlm.nih.gov/articles/PMC8307