

Advanced Satellite Telemetry System Using ESP32 for Real-Time Data Acquisition and Remote Monitoring

¹Omkar Lonarkar, ²Pranav Kachavs, ³Mayur Jamkar, ⁴Govardhan Chavan

^{1,2,3}*Students of E&TC, Shri Shivaji Polytechnic Institute Parbhani*

⁵*Professor, Department of E&TC, Shri Shivaji Polytechnic Institute, Parbhani*

Abstract- Satellite telemetry plays a crucial role in modern communication systems for monitoring remote and inaccessible environments. This paper presents the design and development of an advanced satellite telemetry system using the ESP32 microcontroller integrated with Internet of Things (IoT) technology. The system is capable of acquiring real-time data such as temperature, humidity, motion, current, voltage, and geographical location using multiple sensors.

The acquired data is processed by the ESP32 and transmitted wirelessly to a cloud-based platform using Wi-Fi communication protocols. The system enables real-time monitoring, remote accessibility, and efficient data visualization through mobile applications. Compared to traditional telemetry systems, the proposed system offers reduced cost, improved flexibility, and enhanced performance. The results demonstrate that the system is reliable, scalable, and suitable for applications in satellite communication, environmental monitoring, industrial automation, and defense sectors.

Keywords- Telemetry, ESP32, IoT, Wireless Communication, GPS, Embedded Systems, Real-Time Monitoring

I. INTRODUCTION

Telemetry is defined as the science of automatic data collection and transmission from remote or inaccessible locations to a central monitoring system. It is widely used in satellite communication, weather monitoring, defense systems, and industrial automation. Traditional telemetry systems were based on wired communication and analog transmission techniques, which had several limitations such as high installation cost, maintenance complexity, and lack of real-time data accessibility. With the rapid development of IoT and wireless communication technologies, telemetry systems have evolved into smart, efficient, and scalable systems.

This project focuses on the design and implementation of an IoT-based telemetry system using ESP32. The system is designed to monitor multiple parameters and transmit data wirelessly to a cloud platform. The proposed system provides real-time monitoring, data logging, and remote access capabilities.

II. LITERATURE SURVEY

Earlier research in telemetry systems involved the use of microcontrollers such as Arduino and Raspberry Pi. Arduino-based systems are simple and cost-effective but require additional modules for Wi-Fi communication. Raspberry Pi provides higher processing power but consumes more energy and is not suitable for low-power embedded systems.

Recent advancements have focused on using ESP32 due to its integrated Wi-Fi and Bluetooth features, low power consumption, and high efficiency. Researchers have implemented telemetry systems using IoT platforms such as Blynk and ThingSpeak for real-time data visualization.

However, many existing systems lack integration of multiple sensors and efficient real-time communication. The proposed system addresses these limitations by combining multiple sensing units with efficient wireless data transmission.

III. SCOPE OF THE PROJECT

The scope of the proposed telemetry system includes: Real-time monitoring of environmental parameters such as temperature and humidity Measurement of electrical parameters like current and voltage Motion detection using gyroscope sensors, Location tracking using GPS module, Wireless data transmission using Wi-Fi, Cloud-based monitoring and data visualization

This system can be applied in various fields such as:
 Satellite telemetry systems
 Environmental monitoring stations
 Industrial automation systems
 Smart agriculture
 Defense and surveillance systems

IV. METHODOLOGY

The development of the system follows a systematic approach:

Selection of appropriate sensors and hardware components
 Design of circuit connections and hardware implementation
 Programming of ESP32 using Embedded C in Arduino IDE
 Integration with IoT platform for remote monitoring
 Testing, calibration, and performance evaluation
 the methodology ensures accurate data acquisition, efficient processing, and reliable data transmission.

V. DETAILS OF DESIGN, WORKING AND PROCESSES

5.1 Hardware Design

The hardware system consists of ESP32 microcontroller interfaced with multiple sensors: DHT11 sensor for temperature and humidity measurement MPU6050 sensor for motion and orientation detection INA219 sensor for current and voltage measurement GPS module for location tracking. All sensors are connected using communication protocols such as I2C and UART.

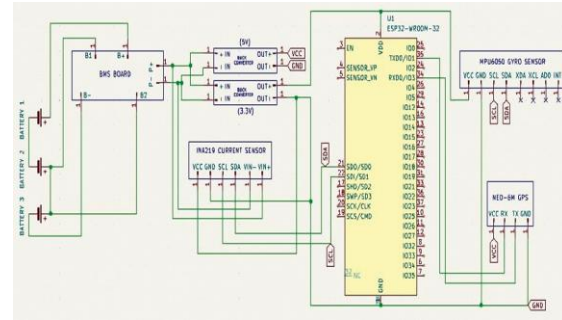
5.2 Software Design

The software is developed using Arduino IDE with Embedded C programming. The ESP32 is programmed to read sensor data, process it, and transmit it to the cloud platform.

5.3 Working Process

Sensors continuously collect real-time data
 ESP32 reads sensor values using communication protocols
 Data is processed and formatted
 ESP32 connects to Wi-Fi network
 Data is transmitted to cloud server
 User monitors data through mobile application

5.4 Circuit diagram



5.5 Communication Process

The system uses IoT communication protocols such as HTTP and MQTT for data transmission. These protocols ensure efficient and reliable communication between the device and the cloud server.

VI. RESULTS AND APPLICATIONS

Results

The system was tested under different environmental conditions and showed accurate and stable performance. Sensor readings were consistent and reliable. Data transmission to the cloud was successful with minimal delay.

Applications

- Satellite telemetry
- Environmental monitoring
- Industrial automation
- Smart agriculture
- Defense systems

VII. CONCLUSION AND FUTURE SCOPE

Conclusion

The proposed telemetry system using ESP32 provides a reliable and cost-effective solution for real-time monitoring. The system integrates multiple sensors and enables efficient wireless communication using IoT technology.

Future Scope

- Integration of LoRa and 5G communication
- Implementation of AI-based data analysis
- Solar-powered system design
- Addition of advanced sensors

VIII. APPENDIX

- Sensor datasheets
- ESP32 pin configuration
- Circuit diagram details

REFERENCES

- [1] ESP32 Datasheet
- [2] Arduino Official Documentation
- [3] Blynk IoT Platform
- [4] Research papers on telemetry systems
- [5] Sensor datasheets