iSWaMS: IoT Smart Waste Management & Environmental Monitoring System with LoRa and Cloud Integration

Guntupalli Danush, Nagaraju Marella, Saba Afreen Khatoon
IoT & Autonomous Systems
Indian Institute of Information Technology, Sri City, Chittoor, India
{omio25s00020@iiits.in, omio25s00030@iiits.in, omio25s00045@iiits.in}

***Abstract*—This project presents an ESP32-based Smart Waste Management and Environmental Monitoring System designed to optimize logistics and ensure safety. The architecture comprises a Smart Bin transmitter and a local Receiver node. Key metrics—including bin levels, environmental conditions, flame detection, and RFID user access—are gathered and serialized via JSON. Data is transmitted using a cyclic protocol over LoRa for local monitoring and Wi-Fi/MQTT to the ThingsBoard cloud, ensuring efficient bandwidth usage and real-time remote analytics.**

***Index terms*—IoT, Waste Management, ESP32, Real-Time Monitoring, LoRa, MQTT**

I. INTRODUCTION

Rapid urbanization has led to significant challenges in waste management and environmental safety [2]. Traditional waste collection methods are often inefficient, leading to overflowing bins or unnecessary fuel consumption with collection trucks visiting empty bins [1]. Furthermore, public bins are often subject to vandalism or fire hazards.

This project addresses these issues by developing an intelligent system that:

1. *Monitors Bin Status:* Real-time fill level detection using ultrasonic sensor (HC-SR04).
2. *Ensures Safety:* Immediate detection of fire (flame sensor) and environmental hazards like high temperature and high humidity (DHT22).
3. *Provides Security*: RFID-based access control to log users and prevent unauthorized dumping and jeopardizing.
4. *Enables Remote Monitoring:* Dual-channel communication using LoRa for local alerts and MQTT for cloud dashboard visualization [3], [4].

II. SYSTEM ARCHITECTURE

A. Transmitter Node (Smart Bin)

The core controller is the ESP32. It interfaces with a suite of sensors including the HC-SR04 (level), DHT22 (environment), PIR (motion), Flame sensor, and GPS module. It uses an RC522 RFID reader for user authentication. Data is processed and serialized into JSON format.

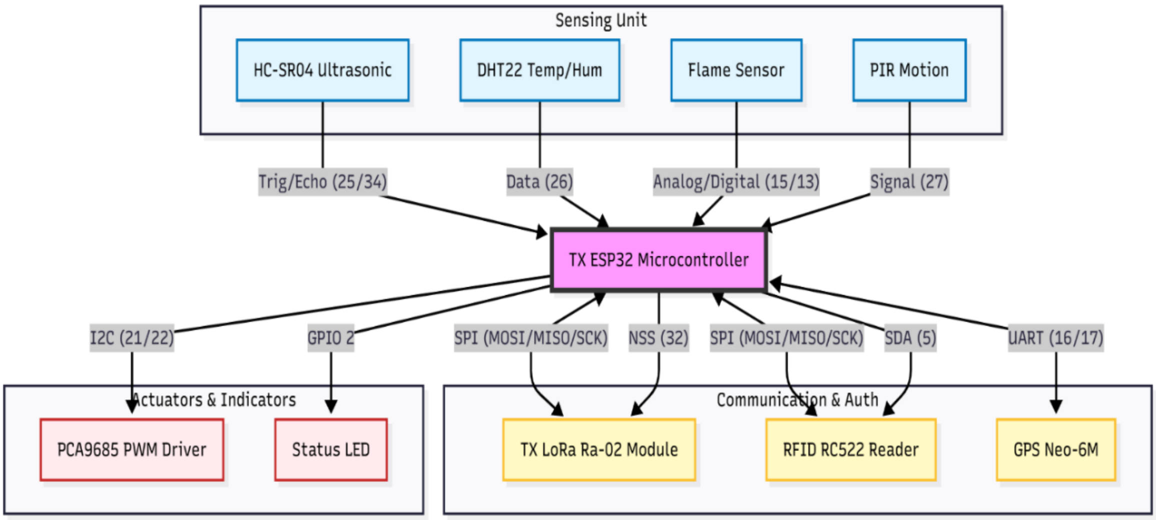

Fig. 1. Transmitter Block Diagram

- *Central Processing Unit: ESP32 (30-pin Dev Module).*
- *Sensors:*
    - *HC-SR04:* Trigger (Pin 25), Echo (Pin 34). Measures distance to waste.
    - *DHT22:* Pin 26. Measures Temp/Humidity.
    - *Flame Sensor:* Digital (Pin 13) & Analog (Pin 15). Detects fire spectrum.
    - *GPS (Neo-6M):* RX/TX (Pins 16/17). Provides Lat/Lng.
    - *PIR:* Pin 27. Detects human presence.
- *Communication:*
    - *LoRa (Ra-02):* SPI Interface (NSS 32, RST 14, DIO0 4).
    - *RFID (RC522):* Shared SPI Bus (SDA 5, RST 33).
- *Actuators:*
    - *PCA9685:* I2C (Pins 21/22). PWM driver for potential servo control.
    - *LED:* Pin 2. Bin level full status indicator.

B. Receiver Node (Monitoring Station)

A second ESP32 unit is equipped with a LoRa receiver, an I2C LCD display and a buzzer. It listens for incoming packets, de-serializes the JSON data, and cycles through status screens (Environment, Bin Level, GPS, RFID). Critical alerts (Fire, Bin Full) trigger interrupt-driven display overrides.

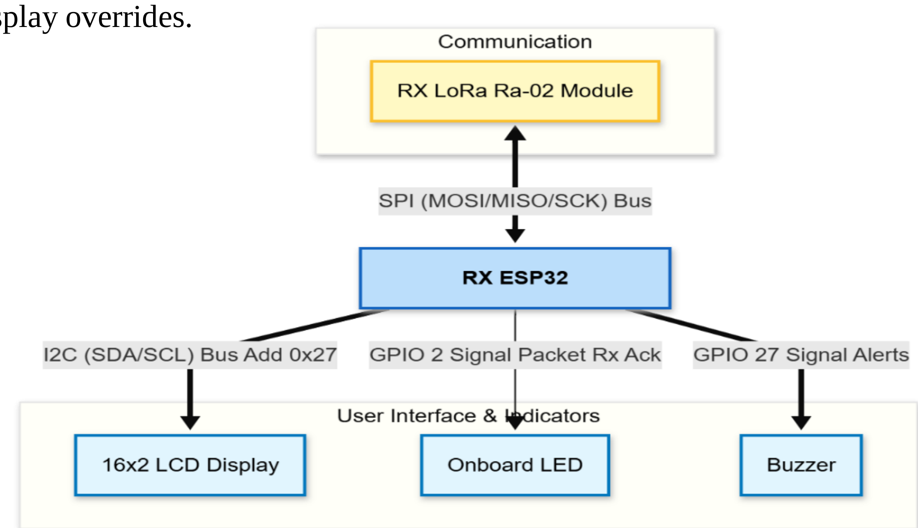

Fig. 2. Receiver Block Diagram

- *Central Processing Unit:* ESP32.
- *Communication:* LoRa (Ra-02) via SPI.
- *Display:* 16x2 LCD via I2C (Add 0x27).
- *Indicators:* On-board LED for packet reception acknowledgment and buzzer for alerts

C. Cloud Interface

The system connects to the ThingsBoard cloud platform via MQTT (Message Queuing Telemetry Transport) [3]. This allows for historical data logging and a graphical dashboard for municipal administrators [2].

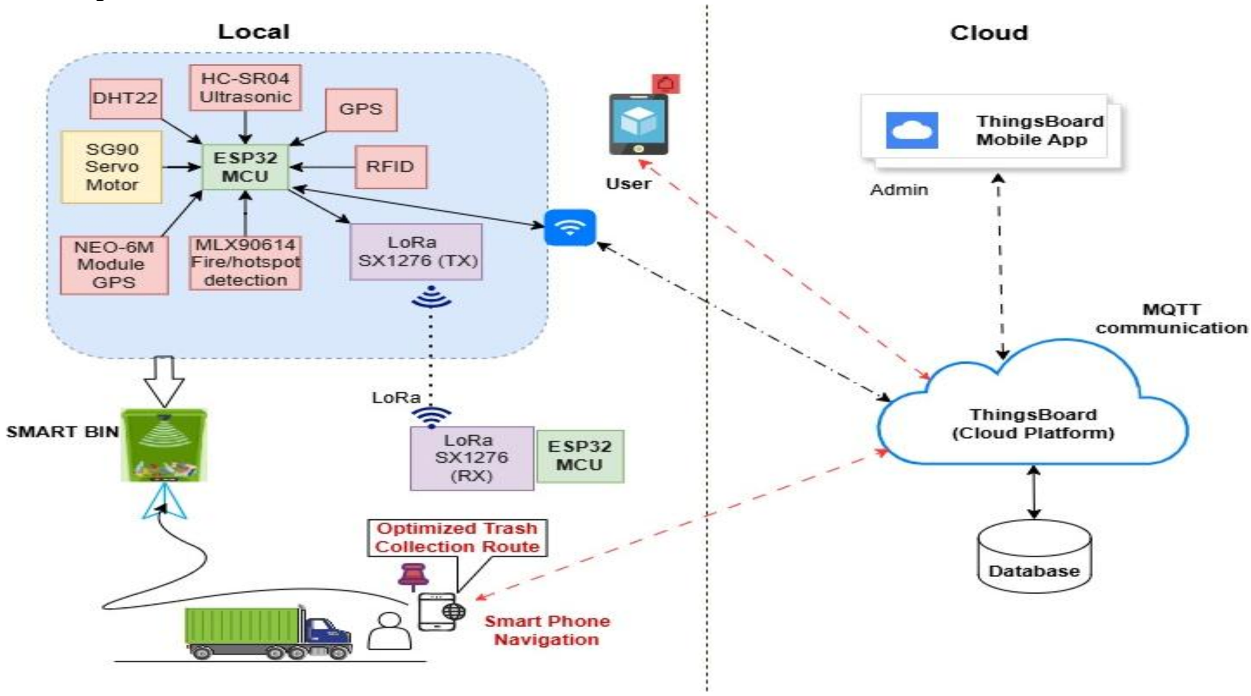

Fig. 3. System Architecture of iSWaMS

III. SOFTWARE IMPLEMENTATION

A. Data Transmission Protocol (Transmitter)

To prevent data collisions and manage the 255-byte bandwidth limit on the LoRa 433MHz band, the firmware implements a Cyclic Transmission Sequence [4]. The loop divides data into five distinct packet types, sent sequentially every 5 seconds:

- *Seq 0 (GPS):* Latitude, Longitude, Connection status.
- *Seq 1 (ENV):* Temperature, Humidity, Flame Analog/Digital values.
- *Seq 2 (BIN):* Ultrasonic distance, Fill percentage, Motion status.
- *Seq 3 (RFID):* Last scanned UID, User Count, Entry/Exit state.
- *Seq 4 (SYS):* System health, PCA9685 status.

This also includes Global Alerts i.e., regardless of the sequence, critical flags (Fire, Motion, High Temp, Bin Full, RFID, Entry, Exit, Timeout) are injected into every LoRa packet to ensure immediate reception at the monitoring station.

B. MQTT Telemetry

While LoRa sends split packets, the Wi-Fi/MQTT handler constructs a single "Mega-Payload" containing all sensor data [3]. This is published to the topic `v1/devices/me/telemetry` on the ThingsBoard cloud.

- *JSON Structure:* The `ArduinoJson` library is used to structure data dynamically.
- *Optimization:* The MQTT buffer size is increased to 1024 bytes to accommodate the large payload of ~400 bytes.

C. Dashboard

The cloud interface utilizes the ThingsBoard platform to provide real-time visualization of MQTT telemetry received from the Smart Bin. The dashboard displays key sensor parameters (from left to right) such as:

1. Bin fill percentage
2. Temperature
3. Humidity
4. Flame intensity

The bin-level indicator provides the current fill status, enabling efficient waste-level monitoring along with it, Gauge widgets are used for temperature and humidity to present environmental variations in an intuitive manner.

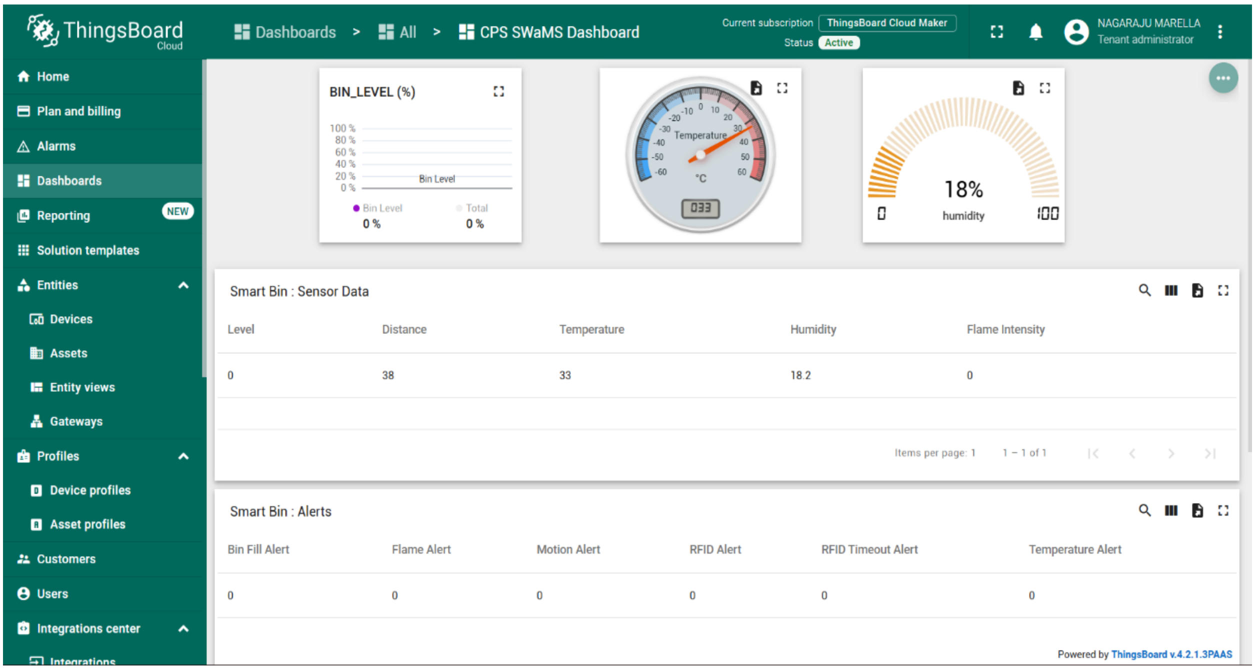

Fig. 4. MQTT Dashboard Interface displaying real-time telemetry from the Smart Bin

- The tabular panels organize information into:

  1. *Sensor Data*: Digital readings of distance, temperature, humidity and flame intensity.
  2. *Alerts*: bin-full state, flame alerts, motion detection, RFID notifications where 0 indicate alert flag as false and 1 indicate alert flag as true.

- The dashboard allows administrators/tenants to:

    1. Monitor live bin status remotely
    2. Verify the presence of any alert conditions
    3. Track environmental changes over time

Overall, the interface effectively supports remote diagnostics and validates the cloud integration capabilities described in Section III-C.

D. RFID Logic & State Machine

The system maintains a local database of users (structured array).

- *New User:* Automatically added to the array upon first tap.
- *State Toggling:* Tapping toggles the user state between `ENTRY` (1) and `EXIT` (2).
- *Timeout:* A global timer resets user states if no activity is detected for 60 seconds, triggering an alert to clear stale data.

E. Receiver Logic

The receiver code constantly parses incoming JSON LoRa packets.

- *Priority Display:* If `alert_flame`, `alert_full`, `alert_temperature`, `alert_rfid` is true, the LCD locks onto the alert screen, overriding normal data cycling.
- *Normal Operation:* If no alerts exist, the LCD rotates every 3 seconds between Environment, Bin Status, GPS, and RFID screens.

IV. SCOPE AND DELIVERABLES

A. Project Scope

The scope of this project is limited to a prototype demonstration of a Smart City solution. It covers:

- Hardware integration of disparate sensors on a shared SPI bus (LoRa + RFID).
- Local range communication up to 1-2km (LoRa theoretical).
- Cloud dashboard integration for visualization.
- Basic automated alerts (LEDs and LCD messages).

B. Deliverables

1. *Smart Bin Prototype:* A fully functional sensing unit capable of measuring waste levels and environmental hazards.
2. *Receiver Unit:* A portable monitoring device displaying real-time metrics.
3. *Firmware Source Code:* Optimized C++ code for ESP32 (Transmitter & Receiver) utilizing FreeRTOS concepts (Wi-Fi handling).
4. *IoT Dashboard:* A ThingsBoard configuration showing graphs for temperature, fill level, and a map widget for GPS location.

V. CHALLENGES

Despite the successful implementation of the prototype, several practical and technical challenges were encountered during the development of the iSWaMS system:

1. The LoRa module's high transmission current requirement, coupled with high current consumption of NEO-6M for cold starting occasionally caused voltage dips resulting in random shutdown of ESP-32. Additional external power supply is used for improved stability.
2. The NEO-6M module must be positioned in an unobstructed open-air environment, as surrounding walls significantly degrade its ability to obtain a satellite fix and maintain accurate positioning.
3. Integrating both the RC522 RFID module and the LoRa Ra-02 on a shared SPI bus resulted in intermittent communication failures. Careful management of chip-select signals and SPI transaction timing was required to ensure stable operation.
4. Simultaneous handling of Wi-Fi, LoRa, GPS, and sensor polling introduced timing conflicts. The use of FreeRTOS queues and non-blocking loops was necessary to prevent watchdog resets.
5. The 255-byte payload restriction in the SX1278 chipset mandated the design of a cyclic packet transmission sequence. Splitting sensor data into multiple sub-packets increased firmware complexity.
6. Maintaining consistent user ENTRY/EXIT states required timeout mechanisms and robust handling of rapid tag taps.

VI. OBSERVATIONS

Several insights were gathered during the development and testing of the iSWaMS system:

1. LoRa performs reliably for short, frequent telemetry packets, but packet integrity drops when transmitting at maximum payload length. The cyclic payload strategy significantly improved consistency in real-world environments.
2. The ultrasonic sensor accuracy is strongly affected by bin shape and waste texture. Flat waste surfaces produce stable readings, while irregular or soft waste introduces higher variance, requiring software filtering.
3. RFID response time is highly dependent on tag orientation and distance. Placing the RC522 module on an external panel improved user tap detection speed and reduced missed scans.
4. GPS lock times vary significantly indoors**,** often exceeding 30 seconds during cold starts. Outdoor placement drastically improved latency and positional accuracy.
5. The ThingsBoard MQTT telemetry was stable**,** and data visualization graphs clearly represented environmental and bin-level trends, confirming the effectiveness of the cloud interface.
6. Power stability directly influences sensor behaviour. When adequately powered, LoRa, GPS, and Wi-Fi modules showed minimal interference, ensuring consistent multi-protocol operation.

VII. CONCLUSION

This project successfully demonstrates a robust architecture for Smart Waste Management [5]. By combining LoRa for reliable, long-range local communication and Wi-Fi for global cloud connectivity, the system offers redundancy and accessibility [4]. The implementation of specific logic, such as the cyclic transmission counter and the RFID state machine, ensures efficient data handling and user management. This system serves as a scalable foundation for Smart City IoT infrastructure, capable of reducing operational costs and improving urban sanitation [1].

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the research supervisors Dr. Hrishikesh Venkataraman and Dr. Raja Vara Prasad Y for their invaluable guidance, constructive feedback, and continuous encouragement throughout the study. We are also grateful to the Electronics Department of Indian Institute of Information Technology, Sri City, Chittoor for providing the necessary support required for conducting the experiments. We would like to acknowledge the contributions of our colleagues and peers for their insightful discussions and technical assistance. Finally, we thank our families and friends for their constant motivation and support during the research and writing of this paper.

REFERENCES

[1] M. Ghahramani, M. Zhou, A. Molter, and F. Pilla, "IoT-Based Route Recommendation for an Intelligent Waste Management System," *IEEE Internet of Things Journal*, vol. 9, no. 14, pp. 11883–11892, Jul. 2022.

[2] I. Sosunova and J. Porras, "IoT-Enabled Smart Waste Management Systems for Smart Cities: A Systematic Review," *IEEE Access*, vol. 10, pp. 73326–73363, 2022.

[3] K. Jaikumar, T. Brindha, T. K. Deepalakshmi, and S. Gomathi, "IOT Assisted MQTT for Segregation and Monitoring of Waste for Smart Cities," in *Proc. 6th Int. Conf. on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2020, pp. 887–891.

[4] T. J. Sheng *et al.*, "An Internet of Things Based Smart Waste Management System Using LoRa and TensorFlow Deep Learning Model," *IEEE Access*, vol. 8, pp. 148793–148811, 2020.

[5] M. Nasiruddin, S. A. L, G. S. Kumar, and A. N. Raj, "IoT Enabled Waste Management Optimization Framework (IWMOF)," in *Proc. 3rd Int. Conf. on Pervasive Computing and Social Networking (ICPCSN)*, Salem, India, 2023, pp. 1541–1546.