iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

IOT ENABLED EMBEDDED PLATFORM FOR SMART WASTEWATER TREATMENT MONITORING

MR. K. Vinod Kumar
Assistant Professor, J.B. Institute of Engineering and Technology

P. Manohar Reddy, K. Abhinav, P. Ramanjaneyulu, P. Vishnu
B. Tech final year students, J.B. Institute of Engineering and Technology

ABSTRACT
The escalating challenges of industrialization and urbanization have led to a significant increase in wastewater production, necessitating advanced treatment and monitoring solutions. Traditional wastewater management systems often rely on manual sampling and periodic laboratory analysis, which are inherently slow, labor-intensive, and prone to human error. Such delays in detecting fluctuations in water quality parameters—like acidity, turbidity, or flow irregularities—can lead to environmental non-compliance, equipment damage, and public health risks. This project presents the design and implementation of an IoT-Enabled Embedded Platform specifically engineered for real-time monitoring and active management of wastewater treatment processes. The system is built around a high-performance ESP32 microcontroller, which serves as the central gateway for data acquisition and cloud communication. To provide a comprehensive profile of water health, the platform integrates a multi-sensor array consisting of a pH sensor for chemical balance, a turbidity sensor for suspended solids, a DS18B20 temperature sensor for thermal variations, and a Hall-Effect flow meter for measuring real-time flow rates and cumulative water volume. The embedded system utilizes a Basic Machine Learning (Logic-based Classification) approach to categorize water quality into three distinct states: CLEAN, MODERATE, or TREAT. This classification is displayed locally on an I2C LCD and transmitted wirelessly to a Thinger.io cloud dashboard. A critical feature of this platform is the Remote Chemical Dosing Capability. Controlled via an active-low relay module, an administrator can manually trigger a dosing pump from the remote dashboard to inject treatment chemicals when the system identifies a "TREAT" or "MODERATE" condition. This creates a "Human-in-the-Loop" safety mechanism, ensuring that chemical resources are used efficiently and only when verified by an operator. Experimental results demonstrate that the system provides high-precision data logging and reliable remote control with minimal latency. By transitioning from manual oversight to an IoT-driven framework, this platform enhances operational efficiency, reduces maintenance costs, and ensures a proactive response to wastewater contamination, contributing to smarter and more sustainable environmental management.

Keywords:
Waste Water, ESP32, Clean, Moderate, Treat , LCD

INTRODUCTION

Background of the Study: Water is the most fundamental natural resource required for the sustenance of life, industrial growth, and ecological balance. However, the rapid pace of global industrialization and uncontrolled urban expansion has led to the generation of massive volumes of wastewater. This wastewater, often contaminated with heavy metals, organic pollutants, and hazardous chemicals, poses a severe threat to natural water bodies and public health if left untreated. Conventional wastewater treatment facilities often struggle with the dynamic nature of influent quality. Monitoring these facilities has traditionally been a manual process, where technicians collect physical samples and transport them to laboratories for chemical analysis. While accurate, this "offline" approach suffers from significant time lags, often taking hours or days to produce results. In a modern industrial context, a delay of even a few hours in detecting a chemical spill or a filtration failure can result in irreparable environmental damage and heavy regulatory fines.

Problem Statement: The primary challenge in contemporary wastewater management is the lack of real-time, actionable data. Most existing systems are "reactive" rather than "proactive." Without continuous monitoring of critical parameters like pH, turbidity, and flow rates, it is impossible to implement precise chemical dosing. Furthermore, many automated systems operate on fixed timers, which leads to two major inefficiencies:

ISSN: 2456-9348
Impact Factor: 8.232

International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

Under-dosing: Failing to neutralize highly acidic or alkaline waste because the dosing volume was insufficient for the actual contamination level. © 2026, IRJET

OBJECTIVES

- To achieve a successful implementation, the project focuses on the following specific objectives:
- To interface a multi-sensor array (pH, Turbidity, DS18B20, and Flow) with a 32-bit microcontroller architecture.
- To develop an algorithm that converts analog sensor signals into standardized environmental units (pH units, NTU, Celsius, and Liters).
- To implement a fail-safe communication protocol between the ESP32 and the IoT Cloud over a Wi-Fi network.
- To design a dual-interface system consisting of a local 16x2 I2C LCD for onsite viewing and a remote web-dashboard for global control. To validate the performance of the chemical dosing relay under different simulated water quality conditions.

METHODOLOGY

The methodology of this project is based on an integrated hardware-software framework designed to provide high-fidelity monitoring and remote control of wastewater parameters. The implementation follows a modular approach, dividing the system into four functional blocks: the Perception Layer (Sensing), the Processing Layer (ESP32 Intelligence), the Communication Layer (Wi-Fi/IoT Cloud), and the Actuation Layer (Chemical Dosing Control). This structured design ensures that each component can be calibrated and tested independently before full system integration Hardware Implementation and Interfacing. The hardware execution involves interfacing a multi-parameter sensor array with the ESP32 microcontroller. The ESP32 was selected for its 12-bit ADC resolution and dual-core architecture, which allows for simultaneous sensor sampling and cloud synchronization. Analog Sensor Integration (pH and Turbidity)The pH Sensor and Turbidity Sensor are interfaced with the ESP32's Analog-to-Digital Converter (ADC) pins (GPIO 34 and 35). Because these sensors operate on a 5V supply but the ESP32 logic is 3.3V, a voltage divider or careful scaling in the firmware is implemented to prevent damage to the microcontroller. The raw analog values are smoothed using an averaging filter to eliminate electronic noise before being converted into environmental units (pH units and Turbidity percentages). Digital Sensor Integration (Temperature and Flow)Temperature (DS18B20): This sensor uses the OneWire protocol, requiring only a single data pin (GPIO 4). A 4.7k ohm pull-up resistor is installed between the VCC and Data lines to maintain signal integrity. Flow Meter (YF-S201): The flow sensor is connected to a digital interrupt pin (GPIO 27). The implementation utilizes Hardware Interrupts to count every pulse generated by the internal Hall-effect turbine. This ensures that the system accurately captures high-velocity flow without losing data packets during the main loop execution. Actuation and Local Display The Active-Low Relay is connected to GPIO 25. In this configuration, the ESP32 pulls the pin to logic '0' (GND) to activate the chemical dosing pump. For local diagnostics, an I2C 16x2 LCD is used, significantly

ISSN: 2456-9348
Impact Factor: 8.232

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

reducing the wiring complexity by using only two wires (SDA and SCL) for communication.

Software Architecture and Firmware Development:
The firmware is developed using the Arduino framework, structured to handle real-time multitasking.
Data Acquisition and Signal Processing The loop begins by requesting a temperature conversion from the DS18B20. Simultaneously, the pH and turbidity voltages are read. The flow rate is calculated every 1000 milliseconds using the formula: The cumulative volume is then updated by integrating the flow rate over time ().
Basic Machine Learning (Logic-Based Classification)The "intelligence" of the system is implemented via a Three-Level Decision Tree. The firmware evaluates the sensor inputs against predefined environmental safety thresholds:
Level 1 (CLEAN): Validated if and .
Level 2 (MODERATE): Triggered if OR.
Level 3 (TREAT): Triggered if OR .
This classification is updated every second and sent to both the local LCD and the cloud dashboard to assist the administrator in decision-making. IoT Cloud Integration and Remote Control . The communication layer is implemented using the Thinger.io platform. Bi-Directional Data Transfer. The implementation utilizes two types of IoT resources: Output Resources (Monitoring): The ESP32 pushes a JSON-like object containing pH, Temperature, Turbidity, Flow Rate, and Water Quality status to the cloud .Input Resources (Command): A "Pump Switch" resource is defined. When the Admin toggles a switch on the web dashboard, the Thinger.io server sends a command packet to the ESP32, which immediately changes the state of the relay pin .Manual Override Logic To ensure safety, the system is programmed with a Manual Override priority. While the "ML" logic classifies the water and displays the "TREAT" warning, the actual physical activation of the relay is reserved for the Admin via the dashboard. This prevents accidental chemical discharge due to a temporary sensor glitch. System Calibration and Testing Implementation concludes with a rigorous calibration phase: pH Calibration: The sensor is tested in Buffer 4.0 and Buffer 7.0 solutions to determine the linear slope for the voltage-to-pH conversion Turbidity Calibration: The sensor is tested in clear distilled water (set as 0%) and highly opaque samples to map the percentage range accurately Connectivity Stress Test: The ESP32 is monitored over a 24-hour period to ensure that the Wi-Fi connection remains stable and that data packets

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

Hardware Assembly and Circuit Integration: The physical implementation of the system requires a structured assembly to ensure electrical stability and protection against the corrosive nature of a wastewater environment. The hardware is housed in an IP65-rated waterproof enclosure to shield the ESP32 and sensitive circuitry from moisture and chemical vapors.
Power Distribution: A 5V/2A DC power adapter provides regulated power to the system. While the ESP32 operates at 3.3V via its onboard regulator, the Turbidity Sensor and Relay Module require a stable 5V rail to function accurately. A common ground (GND) is established across all components to prevent floating voltages that could cause sensor drift.
Signal Isolation: To protect the ESP32 from back-EMF (Electromotive Force) generated by the dosing pump motor, the Active-Low Relay is opto-isolated. This ensures that the high-voltage AC/DC side of the pump is physically and electrically separated from the low-voltage DC control side.
LCD Interface: The 16x2 LCD is connected using an I2C Serial Interface Adapter (PCF8574). This reduces the required GPIO pins from six to just two (SDA and SCL), preserving pins for the flow meter and temperature sensors.

RESULTS AND DISCUSSION

Experimental Setup and Testing Environment

The performance of the **IoT-Enabled Wastewater Monitoring System** was evaluated in a controlled laboratory environment followed by a simulated industrial runoff test. The **ESP32** was connected to a stable 2.4GHz Wi-Fi network (SSID: "IOT"). To simulate various contamination levels, three distinct water samples were prepared:
1.Sample A (Neutral): Tap water with balanced pH and low turbidity.
Sample B (Acidic/Turbid): Water mixed with dilute acetic acid and fine silt.
3.Sample C (Alkaline/Highly Turbid): Water mixed with sodium bicarbonate and concentrated suspended solids.

Sensor Performance and Data Accuracy

The accuracy of the perception layer is fundamental to the system's reliability. The sensors were monitored over a 24-hour continuous cycle to check for drift.

- **pH Sensor Accuracy:** After the calibration of the analog signal, the sensor showed a deviation of only ± 0.15 pH units when compared to a standard laboratory pH meter.
- **Temperature Stability (DS18B20):** By solving the previous "-127" error with a **4.7kΩ pull-up resistor**, the temperature readings remained stable between 24°C and 32°C with no packet loss.
- **Turbidity Mapping:** The mapping function correctly identified clear water as $0\text{-}5\%$ and highly opaque "sludge-like" water as $85\text{-}95\%$, proving the efficiency of the linear scaling algorithm.

Flow Rate and Volumetric Validation

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

The **YF-S201 Flow Sensor** was tested by passing a known volume of 5 liters of water through the system at varying speeds.

- **Measurement:** The system recorded 5.12 liters.
- **Accuracy:** The error rate was approximately 2.4%, which is well within the acceptable industrial tolerance for secondary monitoring systems. This confirms that the **Hardware Interrupt** logic effectively captured all pulses without CPU lag.

Basic ML Classification Results

The "Basic ML" decision tree successfully categorized the water samples in real-time. The classification was updated on the **I2C LCD** and the **Thinger.io Dashboard** simultaneously.

Water Sample	pH Observed	Turbidity %	Classification	System Action
Sample A	7.2	8%	**CLEAN**	Monitoring Only
Sample B	5.8	32%	**MODERATE**	Alert Triggered
Sample C	9.8	72%	**TREAT**	Critical Alert

IoT Connectivity and Remote Actuation

The communication latency between the ESP32 and the **Thinger.io Cloud** was measured to be less than **250ms**.

- **Dashboard Control:** When the Admin toggled the "Pump Switch," the **Active-Low Relay** responded nearly instantly, activating the chemical dosing pump.
- **Data Logging:** The Thinger.io "Data Bucket" successfully logged 2,400 data points over the test period, allowing for the generation of trend graphs showing the relationship between pH spikes and flow volume.

Discussion of Findings

The results confirm that a high-end SCADA system is not always necessary for effective wastewater management. The **ESP32** demonstrated sufficient processing power to handle four sensors, a local display, and a cloud uplink simultaneously.

A key observation was the **Safety of the "Human-in-the-Loop" model**. While the system correctly identified the "TREAT" status for Sample C, the fact that the pump required an Admin command prevented any accidental over-dosing that might occur in a purely autonomous system due to temporary sensor turbulence.

Summary of Results

The implementation proved that:

1. **Hardware Stability:** Proper pull-up resistors and power isolation are mandatory for industrial-grade reliability.
2. **Software Efficiency:** Interrupt-driven flow sensing is significantly more accurate than polling.
3. **Cloud Utility:** Remote monitoring allows for centralized management of multiple treatment nodes, reducing the need for constant on-site physical presence.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who supported and guided me throughout the development of this project titled “IoT-Enabled Embedded Platform for Smart Wastewater Monitoring and Management. ”First and foremost, I thank my project guide for providing clear direction, technical insights, and consistent feedback at every stage of the project. Their guidance helped in structuring the system effectively and overcoming practical challenges during implementation. I also extend my thanks to the faculty members of the department for their valuable suggestions and for providing the necessary academic support and resources required to complete this work. I am grateful to my friends and peers who contributed through discussions, troubleshooting, and sharing ideas, which improved the overall quality of the project.

Finally, I would like to acknowledge the availability of open-source tools, platforms, and documentation that played a crucial role in building and testing the system efficiently. If you want something stronger (more technical tone, less generic college style), say it—I can rewrite it to sound like a serious engineering report instead of a typical student submission.

ISSN: 2456-9348
Impact Factor: 8.232

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

CONCLUSION

Wastewater Treatment successfully demonstrate the integration of low-cost embedded hardware with cloud-based intelligence. By utilizing the **ESP32 microcontroller**, the project provides a robust solution for real-time monitoring of critical environmental parameters including pH, temperature, turbidity, and flow rate.

The core success of this project lies in its ability to transform raw sensor data into actionable information through **Basic Machine Learning (Logic-Based Classification)**. The system effectively categorizes water quality into **CLEAN, MODERATE, and TREAT** levels, allowing for a simplified user interface that non-technical operators can easily interpret. Furthermore, the **Human-in-the-Loop** model—implemented via the **Thinger.io** dashboard—ensures that chemical dosing is only performed when necessary and verified by an administrator, thereby optimizing chemical usage and reducing operational costs.Technically, the project resolved significant hardware challenges, such as the digital bus stability of the **DS18B20** and the accurate volumetric tracking of the **YF-S201** flow sensor using interrupt-driven logic. The final prototype serves as a reliable, scalable, and affordable alternative to expensive industrial SCADA systems, making advanced wastewater management accessible to smaller industrial units and housing societies.

REFERENCES

[1] The development of this **IoT-Enabled Embedded Platform** is based on established research in environmental engineering, embedded systems design, and Internet of Things protocols. The following references provide the technical foundation for the sensors, communication logic, and water quality standards used in this project.
[2] **Espressif Systems (2023).** *ESP32 Series Datasheet v4.1.* Technical specifications for the dual-core 32-bit MCU, ADC characteristics, and Wi-Fi stack integration.
[3] **Dallas Semiconductor (2019).** *DS18B20 Programmable Resolution 1-Wire Digital Thermometer.* Detailed protocol for OneWire communication and temperature sensing logic.
[4] **Thinger.io Documentation (2024).** *Thinger.io Server API and Client Library Reference.* Implementation of bi-directional data streams and device resource mapping for ESP32.
[5] **Sensorex (2022).** *pH Electrode Theory and Practice.* Technical guide on the Nernst Equation and the calibration of analog pH probes using buffer solutions.
[6] **Kaur, M., & Singh, P. (2022).** "Real-time Water Quality Monitoring System using IoT and Machine Learning." *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering.* Discusses the transition from manual sampling to rule-based classification.
[7] **Zhang, L., et al. (2021).** "A Low-Cost Smart Water Quality Monitoring System Based on ESP32." *IEEE Sensors Journal.* Analysis of ADC stability and Wi-Fi interference in 32-bit microcontrollers.
[8] **Vijaykumar, S., & Ramaswamy, R. (2023).** "Design of an Automated Chemical Dosing System for Industrial Wastewater Treatment." *Journal of Environmental Management.* Exploration of human-in-the-loop control systems for chemical optimization.
[9] **World Health Organization (WHO) (2022).** *Guidelines for Drinking-water Quality: Fourth Edition.* Standardized thresholds for pH and Turbidity levels used in the "Basic ML" classification logic.
[10] **ISO 7027-1:2016.** *Water quality — Determination of turbidity.* International standards for the measurement of suspended solids using light scattering techniques.
[11] **Arduino Reference (2024).** *Interrupts and Timer Logic in ESP32.* Documentation for attachInterrupt() and IRAM_ATTR used in the flow meter pulse counting.
[12] **OneWire Library (v2.3.7).** *Paul Stoffregen's OneWire Communication Protocol.* Technical implementation of the digital bus timing required to fix the -127 temperature error.
[13] **LiquidCrystal_I2C Library.** *PCF8574 I2C LCD Driver.* Implementation guide for mapping serial data to a 16x2 character display.