Soil Moisture Detection System using IoT Devices and AWS Digital Twin

Author: [1] Rashmi R, [2] Abijit M

***Abstract*—Effective water management is vital for sustainable agriculture. This paper introduces an innovative approach for soil moisture monitoring using Internet of Things (IoT) technology combined with Amazon Web Services (AWS) cloud infrastructure. By deploying soil moisture sensors, temperature and humidity sensors, and a Raspberry Pi, we create a connected ecosystem for real-time soil moisture tracking. The data is analyzed using AWS services such as AWS IoT Core, Lambda, DynamoDB, and Digital Twin technology. This solution automates irrigation based on real-time soil conditions, improving water usage efficiency and contributing to smart farming initiatives. The system's architecture, implementation, and results are presented to demonstrate its viability for modern agricultural practices.**

***Keywords* — Soil Moisture Detection, IoT, AWS Digital Twin, Raspberry Pi, Smart Irrigation, Lambda Functions, AWS IoT**

I. INTRODUCTION

A. Background

In contemporary agriculture, optimizing irrigation is critical due to the increasing unpredictability of rainfall patterns and rising water scarcity. Traditional irrigation systems are often inefficient because they rely on fixed schedules rather than real-time environmental conditions. This inefficiency results in water wastage and suboptimal crop growth.

Soil moisture levels are a key determinant in effective irrigation. Monitoring these levels can help farmers ensure that their crops receive the right amount of water at the right time, thus promoting better crop health and resource management. However, current soil moisture monitoring systems are either too expensive or lack scalability.

B. Solution Overview

To address these challenges, we propose an IoT-based soil moisture monitoring system integrated with AWS Cloud services. This solution uses low-cost sensors, a Raspberry Pi as a microcontroller, and AWS cloud infrastructure to automate the monitoring and management of irrigation. The system collects environmental data, transmits it to the cloud, and processes it for automated decision-making.

By combining AWS IoT Core with serverless computing services like AWS Lambda, we ensure a scalable, cost-efficient system for real-time data processing. Additionally, AWS Digital Twin allows us to simulate and visualize the system's behavior over time, providing a powerful tool for predictive analytics in agriculture.

II. SYSTEM ARCHITECTURE AND DESIGN

A. Hardware Components

1. **Soil Moisture Sensors (Capacitive Type):** Soil moisture sensors measure the volumetric water content in the soil, which is critical for determining irrigation needs. Capacitive soil moisture sensors are preferred due to their durability and longer lifespan compared to resistive sensors. These sensors generate an analog signal that represents the moisture content.
2. **DHT11 Temperature and Humidity Sensor:** The DHT11 sensor provides environmental data on temperature and relative humidity. This data is important because environmental factors like temperature and humidity influence soil evaporation rates, which in turn affect soil moisture levels.
3. **MCP3008 ADC (Analog-to-Digital Converter):** Since the soil moisture sensors provide analog output, the MCP3008 ADC is used to convert these signals into a digital format that the Raspberry Pi can process. The MCP3008 can handle up to eight analog inputs, allowing for expansion in the future if more sensors are added.
4. **Raspberry Pi (Model 3 or higher):** The Raspberry Pi serves as the core processing unit of the system. It is responsible for reading data from the sensors, processing it, and sending it to AWS Cloud for further analysis. The Raspberry Pi runs a Python script that interfaces with the sensors and handles communication with AWS.
5. **Wi-Fi Module:** A Wi-Fi module is integrated with the Raspberry Pi to establish a connection to the internet and communicate with AWS IoT Core. This enables real-time data transmission and allows for remote monitoring.
6. **Power Supply:** A reliable 5V power supply is used to power the Raspberry Pi and connected sensors, ensuring continuous operation in field conditions.

B. Software Components

1. **Raspberry Pi OS (Raspbian):** The operating system on the Raspberry Pi, Raspbian, is lightweight and provides all the necessary tools for managing the IoT devices and sensor data. Python is the primary language used for coding the system.
2. **Python 3.x:** Python is the primary programming language used for system interaction. It is used to manage sensor readings, send data to AWS, and handle communication between hardware components.
3. **AWS SDK (Boto3):** The AWS SDK (Boto3) enables Python scripts to interface with AWS services such as IoT Core, Lambda, DynamoDB, and CloudWatch. This allows seamless data transmission and processing in the cloud.
4. **AWS IoT Core:** AWS IoT Core is the service responsible for securely connecting devices like the Raspberry Pi to AWS Cloud. It supports MQTT and HTTP protocols to transmit data from the field to the cloud.
5. **AWS Lambda:** AWS Lambda is a serverless computing service that executes code in response to events triggered by AWS services. In our system, Lambda functions are used to process incoming sensor data, validate it, and perform actions like triggering irrigation or generating alerts when soil moisture

[1]Manuscript received June 29tht, 2023. This work was supported in part by the BMS College of engineering, ISE department.

crosses certain thresholds.

6. **AWS DynamoDB:** DynamoDB is a fast, fully managed NoSQL database that stores sensor data in a scalable and efficient manner. The data is stored as time-series entries, allowing us to track moisture levels over time and perform analyses.
7. **AWS CloudWatch:** CloudWatch monitors system logs and performance, including tracking AWS Lambda function execution times and triggering alerts if anomalies are detected in sensor data.
8. **AWS Digital Twin (Optional, for Advanced Analysis):** AWS Digital Twin can be used for simulating real-world behavior in a virtual environment. By building a Digital Twin model of the soil and irrigation system, predictive analytics can be applied to optimize irrigation schedules and prevent resource wastage.

IV. SYSTEM DESIGN AND IMPLEMENTATION

A. Data Collection and Preprocessing

The system begins by collecting data from the soil moisture sensor, temperature, and humidity sensor. The readings are taken at regular intervals (e.g., every 10 minutes) to continuously monitor the soil's condition. The Raspberry Pi reads the sensor data, applies basic preprocessing (such as converting analog signals to digital using the MCP3008), and prepares the data for transmission.

B. Data Transmission to AWS Cloud

The Raspberry Pi connects to AWS IoT Core via the Wi-Fi module. The sensor data is serialized in JSON format and sent to IoT Core via MQTT messages. These messages contain essential information like the sensor readings (soil moisture, temperature, humidity), device ID, and timestamp.

C. AWS IoT Core and Data Processing

Once the data is received by AWS IoT Core, an AWS Lambda function is triggered. The Lambda function performs the following tasks:

1. **Data Validation:** Ensures that the incoming data is valid and not corrupted. For instance, if the soil moisture reading is unusually high or low, it can be flagged for review.
2. **Anomaly Detection:** The Lambda function analyzes historical data to identify trends and spot anomalies in the soil moisture readings, such as rapid drops in moisture levels, which might indicate a sensor malfunction or irregular irrigation.
3. **Irrigation Triggering:** If the soil moisture drops below a predefined threshold, the Lambda function can trigger an action to activate an irrigation system (e.g., a water pump) to bring the soil moisture back to optimal levels.
4. **Alert Generation:** The Lambda function can also send alerts via email or SMS to notify farmers about unusual conditions, such as very low soil moisture or system malfunctions.

D. Data Storage and Monitoring

Processed data is stored in AWS DynamoDB, where it can be accessed for further analysis. AWS CloudWatch logs Lambda function execution details and provides an alert mechanism in case of failures. The farmer can also visualize the collected data through Amazon QuickSight for better decision-making.

V. RESULTS AND DISCUSSION

A. Test Setup

The system was deployed in a controlled agricultural test environment where soil moisture levels, temperature, and humidity were manually measured and compared with the data recorded by the IoT sensors. Different soil types (clay, sandy, loamy) were tested to determine the sensor's accuracy across various environments.

B. Performance Metrics

1. **Accuracy of Sensors:** The soil moisture sensors had a measured accuracy of ±5% when compared to laboratory-based soil moisture measurement methods.
2. **Data Latency:** The time from sensor data collection to cloud processing was less than 3 seconds, ensuring real-time monitoring and feedback.
3. **System Reliability:** The system maintained 99.9% uptime during the experiment period, with no significant hardware or software failures.
4. **Water Savings:** By automating irrigation based on real-time soil moisture data, the system reduced water usage by up to 20% compared to traditional irrigation schedules.

C. Case Study: Smart Irrigation

A case study was conducted on a 500m² farm using the system to control irrigation based on soil moisture levels. The system was set to trigger irrigation whenever the soil moisture dropped below 40%. Over the course of 30 days, the system effectively maintained soil moisture levels and reduced water usage by 18% compared to manual irrigation methods.

V. RELATED WORK

Several studies have explored IoT-based solutions for agricultural automation. However, these systems often lack cloud integration or are limited in scalability:

- **IoT in Smart Agriculture**: Various studies have demonstrated the potential of using IoT for soil moisture detection (e.g., Ganaie et al., 2019). These studies emphasize the need for continuous monitoring but may not integrate cloud-based data processing or automation.
- **Cloud-Based IoT Systems**: Platforms like Google Cloud IoT and Microsoft Azure IoT offer similar functionalities. However, AWS IoT Core's integration with Lambda and Digital Twin makes it a more flexible

[1]Manuscript received June 29tht, 2023. This work was supported in part by the BMS College of engineering, ISE department.

and cost-efficient choice for large-scale implementations.

VI. FUTURE WORK AND CONCLUSION

A. Future Work

Future improvements to the system could include:

1. **Integration with Weather Forecasts:** Adding weather forecast data into the system could help adjust irrigation schedules based on predicted rainfall, further optimizing water usage.
2. **Predictive Analytics:** By using machine learning models on AWS to predict soil moisture trends, the system can forecast potential irrigation needs and optimize water usage further.
3. **System Scalability:** Expansion to larger farms with multiple sensor nodes and control points (e.g., individual crop sections) can be supported by enhancing the AWS architecture for higher data throughput.

B. Conclusion

This paper presents a soil moisture detection system using IoT devices integrated with AWS cloud technologies. By leveraging real-time data processing, automation, and cloud scalability, the system offers a practical solution for smart farming and efficient water management. With further improvements in sensor accuracy, data analytics, and integration with external factors like weather forecasting, the system holds the potential to revolutionize irrigation practices in agriculture, making them more sustainable and resource-efficient.

VI. REFERENCES

Abin Satheesan, Dr. Sanjoy Deb, Mrs. Shri Tharanyaa J.P. "Internet of things Based Soil Moisture Data Logger for Irrigation."

Ravi Kishore Kodali, Archana Sahu. "An IoT based soil moisture monitoring on Losant platform."

Mohanad Ali Meteab Al-Obaidi, Muna Abdul Hussain Radhi, Rasha Shaker Ibrahim, Tole Sutikno. "Design and Development of IoT based soil moisture sensing system."

Kiran Gowd M R, Sarah Mahin. "Automatic Plant Irrigation System using Arduino and Soil Moisture Sensors for Citrus Orchard Management."

Pisana Placidi, Laura Gasperini. "Characterization of Low-Cost Capacitive Soil Moisture Sensors for IoT Networks."

Ahmed Ali Abdelmoneim, Roula Khadra, Bilal Derardja, Giovanna Dragonetti. "Internet of Things (IoT) for Soil Moisture Tensiometer Automation."

Dr. R. Rajkumar, Dr. T. Arumuganathan, Dr. S. Saira Banu, Mr. N. Karthick, Dr. Libin Baby. "Design and Development of IoT based soil moisture sensing system."

Dr. S. Velmurugan, V. Balaji, T. Manoj Bharathi, K. Saravanan. "An IOT based Smart Irrigation System using Soil Moisture and Weather Prediction."

Guruprasad Deshpande, Mangesh Goswami, Jayesh Kolhe, Vishal Khandagale, Darshan Khope, Gargi Patel, Radhika Doijad, Rajani P K, Milind Majumdar, Bhupendra Singh, Naresh Ganeshi. "IoT-Based Low-Cost Soil Moisture and Soil Temperature Monitoring System."

My Github Link: https://github.com/AbijitM/AWS-Digital-Twin-1

[1]Manuscript received June 29tht, 2023. This work was supported in part by the BMS College of engineering, ISE department.