ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

DEPARTMENT OF SOFTWARE ENGINEERING

COURSE:EMBEDDED SYSTEM

COURSE CODE: SWEG4102

SECTION: B

FINAL REPORT: MOBILE CONTROLLED WATER SPRINKLER

GROUP 3

Group members	**ID.NO**
1. Betselot Tesfa	ETS0327/14
2. Bezawit Edilu	ETS0334/14
3. Biruk Lemma	ETS0368/14
4. Diana Luel	ETS0490/14
5. Etsub Girma	ETS0563/14
6. Eyerusalem Rufael	ETS0572/14
7. Frezer Metasebia	ETS0679/14

SUBMITTED TO: Inst. Kassahun

SUBMISSION DATE : May 19, 2025

TABLE OF CONTENTS

Mobile Controlled Water Sprinkler

Abstract

The Mobile Controlled Water Sprinkler system is an innovative Internet of Things (IoT) solution designed to automate irrigation through remote monitoring and control. By leveraging the ESP32 microcontroller, a soil moisture sensor, a relay module, and a water pump, the system enables users to monitor soil moisture levels and control irrigation via the Blynk IoT platform. The project integrates hardware and software components to create a reliable, user-friendly, and efficient irrigation system that promotes water conservation and reduces manual effort. Extensive testing validated the system's functionality, demonstrating its potential for applications in home gardening, small-scale agriculture, and educational settings. This report details the design, implementation, and successful outcomes of the project, highlighting its contributions to smart agriculture and embedded systems education.

Introduction

Water is a critical resource for agriculture and gardening, yet traditional irrigation methods often lead to inefficiencies, such as overwatering or inconsistent watering schedules. Manual irrigation is labor-intensive and impractical for individuals with busy schedules or large areas to maintain. With the advent of embedded systems and IoT technologies, automated irrigation systems have emerged as a transformative solution, enabling precise water management and remote accessibility. The Mobile Controlled Water Sprinkler system addresses these challenges by combining an ESP32 microcontroller, a soil moisture sensor, a relay-controlled water pump, and the Blynk IoT platform to create a smart irrigation system.

This project was developed as part of an academic assignment on embedded systems, aiming to demonstrate the practical application of microcontroller programming, sensor integration, and IoT communication. The system allows users to monitor soil moisture levels in real-time and control a water pump remotely via a mobile application, ensuring optimal irrigation with minimal effort. The use of affordable components like the ESP32 and open-source software like Blynk makes the system

accessible to hobbyists, students, and small-scale farmers. This report provides a comprehensive overview of the project, including its objectives, hardware and software design, implementation details, and the highly successful results achieved during testing.

Objective

The Mobile Controlled Water Sprinkler system was developed with the following objectives:

1. **Real-Time Monitoring**: To accurately measure soil moisture levels using a sensor and display the data in real-time on a mobile application, enabling users to make informed irrigation decisions.
2. **Remote Control**: To allow users to activate or deactivate a water pump remotely via the Blynk IoT platform, providing convenience and flexibility.
3. **Water Efficiency**: To promote efficient water usage by enabling precise control over irrigation, reducing wastage, and ensuring plants receive adequate hydration.
4. **System Integration**: To seamlessly integrate hardware components (ESP32, moisture sensor, relay, pump) with software (Blynk and C++ code) to create a robust embedded system.
5. **Reliability and Scalability**: To design a reliable system that operates consistently under various conditions and can be scaled for larger applications or enhanced with additional features.
6. **Educational Demonstration**: To serve as a practical example of embedded systems and IoT for academic purposes, showcasing sensor interfacing, microcontroller programming, and cloud-based communication.

Significance and Limitation of the Project

Significance

The Mobile Controlled Water Sprinkler system offers numerous benefits, making it a valuable contribution to smart agriculture and embedded systems education:

- **Water Conservation**: By allowing precise control over irrigation, the system minimizes water wastage, aligning with global efforts to conserve water resources in agriculture.
- **Time and Effort Savings**: Remote control via a mobile app eliminates the need for manual watering, making it ideal for busy individuals or those managing multiple garden areas.
- **Cost-Effectiveness**: The use of affordable components like the ESP32 (priced at approximately $5–10) and the free Blynk platform ensures the system is accessible to a wide audience, including students and small-scale farmers.
- **Scalability and Flexibility**: The system's modular design allows for the addition of sensors (e.g., temperature or light sensors) or integration with automated irrigation logic, making it adaptable for larger agricultural applications.
- **Educational Impact**: The project provides hands-on experience with embedded systems, IoT, and sensor integration, serving as a practical learning tool for students studying electronics, computer engineering, or IoT technologies.
- **Environmental Benefits**: Efficient irrigation reduces runoff and soil erosion, contributing to sustainable gardening practices.
- **User Accessibility**: The Blynk app's intuitive interface ensures that even non-technical users can operate the system effectively, broadening its potential user base.

Limitations

Despite its strengths, the system has some constraints that warrant consideration:

- **Internet Dependency**: The system relies on a stable Wi-Fi connection for communication with the Blynk server, which may be a challenge in areas with unreliable internet access.
- **Power Requirements**: Continuous operation of the ESP32 and water pump requires a reliable power source, which could be a limitation in off-grid settings unless paired with a battery or solar power.

- **Sensor Calibration**: The soil moisture sensor's accuracy depends on proper calibration and may vary with different soil types (e.g., sandy vs. clay soils), requiring adjustments for optimal performance.
- **Limited Automation**: The current design relies on manual control via the Blynk app, though future iterations could incorporate automated irrigation based on moisture thresholds.
- **Scale Constraints**: While suitable for small gardens, scaling the system for large agricultural fields would require additional pumps, sensors, and infrastructure.
- **Component Durability**: Prolonged exposure to outdoor conditions may affect the longevity of components like the moisture sensor or relay unless weatherproof enclosures are used.

Hardware Part of the Project

The Mobile Controlled Water Sprinkler system was built using a carefully selected set of hardware components, each chosen for its functionality, affordability, and compatibility with the ESP32 microcontroller. The components and their roles are detailed below:

- **ESP32 Microcontroller**: The ESP32, a powerful dual-core microcontroller with built-in Wi-Fi and Bluetooth, serves as the system's brain. It processes sensor data, controls the relay, and communicates with the Blynk server. Its 32-bit architecture and 240 MHz clock speed ensure efficient performance for real-time applications.
- **Soil Moisture Sensor**: This capacitive or resistive sensor measures soil moisture by detecting changes in electrical resistance or capacitance. Connected to analog pin 34 of the ESP32, it provides raw analog values (0–4095) that are converted to a moisture percentage for user display.
- **Relay Module**: A single-channel 5V relay module controls the high-voltage water pump by switching it on or off. Connected to digital pin 25, the relay isolates the low-voltage ESP32 circuit from the pump's high-voltage circuit, ensuring safety and reliability.

- **Water Pump**: A 5V or 12V submersible water pump delivers water through a connected pipe to the sprinkler system. The pump's flow rate (typically 100–200 liters per hour) is suitable for small-scale irrigation.
- **Water Pipe**: A flexible PVC or silicone pipe channels water from the pump to the irrigation area, ensuring efficient water distribution without leaks.
- **Diode (1N4007)**: A flyback diode is placed across the relay's coil to protect the ESP32 from voltage spikes caused by the relay's inductive load, enhancing circuit longevity.
- **Jumper Wires**: Male-to-female and male-to-male jumper wires connect the components to the ESP32 and board, ensuring secure and organized wiring.
- **LED Lights**: Two LEDs (green and red) indicate system status. The green LED lights up when the ESP32 is connected to the Blynk server, while the red LED indicates pump activation, providing visual feedback.
- **Mounting Board**: A prototyping board or custom PCB holds the ESP32, relay, and other components, ensuring a stable and compact assembly.
- **Charger Cables**: Two USB cables are used—one to power the ESP32 (5V, typically via a USB power bank or adapter) and another to supply power to the water pump (5V or 12V, depending on the pump's requirements).

Hardware Assembly

The hardware was assembled with precision to ensure reliability and safety:

1. The ESP32 was mounted on the board, with its pins accessible for connections.
2. The soil moisture sensor was connected to analog pin 34, with its VCC and GND pins wired to the ESP32's 3.3V and ground rails, respectively.
3. The relay module was connected to digital pin 25 for control, with its VCC and GND pins powered by the ESP32's 5V and ground rails. A diode was soldered across the relay's coil to suppress voltage spikes.
4. The water pump was wired to the relay's normally open (NO) and common (COM) terminals, with a separate power supply (via the second charger cable) providing the necessary voltage.

5. LEDs were connected to additional digital pins (e.g., pins 26 and 27) with current-limiting resistors (220 ohms) to indicate connection and pump status.
6. Jumper wires were used to complete all connections, with care taken to avoid loose or exposed wires.
7. The water pipe was attached to the pump's outlet, and the pump was submerged in a water reservoir for testing.
8. The entire assembly was tested for continuity and insulation to prevent short circuits or electrical hazards.

The hardware design prioritized modularity, allowing for easy troubleshooting and future upgrades, such as adding more sensors or a weatherproof enclosure.

Software Part of the Project

The software component of the Mobile Controlled Water Sprinkler system consists of two primary elements: the Blynk IoT platform and a C++ program running on the ESP32. These components work together to enable remote control, real-time monitoring, and seamless communication between the hardware and the user's mobile device.

Blynk IoT Platform

Blynk is a versatile IoT platform that provides a drag-and-drop interface for creating mobile applications to control and monitor IoT devices. For this project, the Blynk app was configured with the following widgets:

- **Button Widget (Virtual Pin V0)**: A toggle button allows the user to send a binary value (1 or 0) to the ESP32, controlling the relay and, consequently, the water pump. The button was styled as a switch for intuitive operation.
- **Gauge Widget (Virtual Pin V1)**: A gauge displays the soil moisture percentage (0–100%) in real-time, updating every 2 seconds. The gauge was customized with a green-to-red color gradient to visually indicate moisture levels (green for wet, red for dry).

- **Status Indicator**: A built-in Blynk feature shows the connection status between the ESP32 and the Blynk server, ensuring users are aware of the system's operational state.

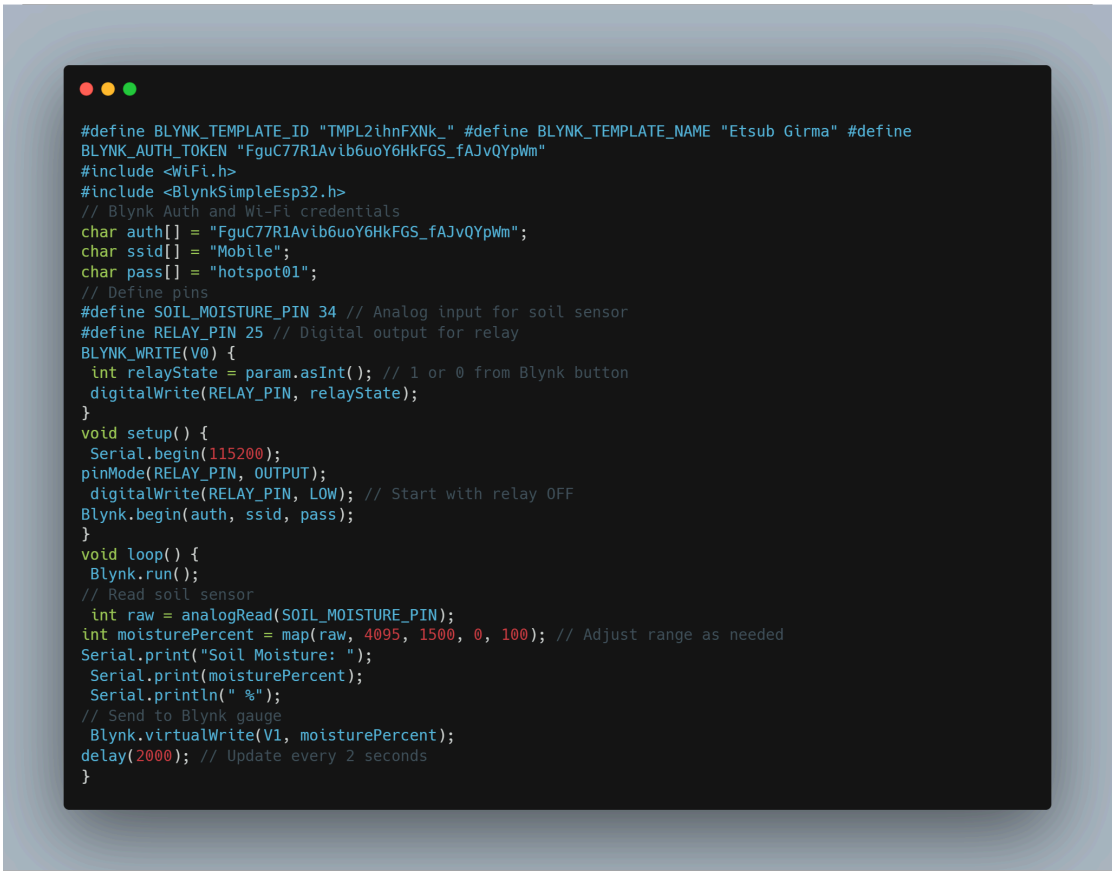
The Blynk app was set up using the provided template ID, template name, and authentication token, ensuring secure communication with the ESP32. The app's user-friendly interface made it accessible to users with minimal technical expertise, enhancing the system's practicality.

C++ Code for ESP32

The ESP32 was programmed using the Arduino IDE with the Blynk library (BlynkSimpleEsp32.h) and Wi-Fi library (WiFi.h). The C++ code handles the following tasks:

- Initializes the ESP32 and establishes a Wi-Fi connection to the Blynk server.
- Reads analog data from the soil moisture sensor and converts it to a percentage.
- Sends moisture data to the Blynk app for display on the gauge.
- Listens for user input from the Blynk app to control the relay and water pump.
- Provides serial output for debugging and monitoring.

The C++ code, developed using the Arduino IDE, is as follows:



```
#define BLYNK_TEMPLATE_ID "TMPL2ihnFXNk_" #define BLYNK_TEMPLATE_NAME "Etsub Girma" #define
BLYNK_AUTH_TOKEN "FguC77R1Avib6uoY6HkFGS_fAJvQYpWm"
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
// Blynk Auth and Wi-Fi credentials
char auth[] = "FguC77R1Avib6uoY6HkFGS_fAJvQYpWm";
char ssid[] = "Mobile";
char pass[] = "hotspot01";
// Define pins
#define SOIL_MOISTURE_PIN 34 // Analog input for soil sensor
#define RELAY_PIN 25 // Digital output for relay
BLYNK_WRITE(V0) {
 int relayState = param.asInt(); // 1 or 0 from Blynk button
 digitalWrite(RELAY_PIN, relayState);
}
void setup() {
 Serial.begin(115200);
pinMode(RELAY_PIN, OUTPUT);
 digitalWrite(RELAY_PIN, LOW); // Start with relay OFF
Blynk.begin(auth, ssid, pass);
}
void loop() {
 Blynk.run();
// Read soil sensor
 int raw = analogRead(SOIL_MOISTURE_PIN);
int moisturePercent = map(raw, 4095, 1500, 0, 100); // Adjust range as needed
Serial.print("Soil Moisture: ");
 Serial.print(moisturePercent);
 Serial.println(" %");
// Send to Blynk gauge
 Blynk.virtualWrite(V1, moisturePercent);
delay(2000); // Update every 2 seconds
}
```