© May 2024| IJIRT | Volume 10 Issue 12 | ISSN: 2349-6002

Auto Irrigation System Using ESP32

Rutuja Sangade[1], Raman.S.Gandewar[2], Adwita.A.Gandhi[3], Siddhant.U.Ghat[4], Prathamesh.S.Ghalsasi[5], Abhay.A.Gharage[6], Pranav.S.Godbole[7]

[1,2,3,4,5,6,7]*Department of Engineering, Sciences and Humanities (DESH) Vishwakarma Institute of Technology, Pune, 411037, Maharashtra, India*

Abstract: **The Auto irrigation System is an advanced solution designed to address the challenges of maintaining optimal soil moisture levels for plants in various environments. This project focuses on creating a reliable, automated irrigation system that prioritizes plant well-being by delivering precise water amounts when needed.**

The system integrates advanced sensors and actuators to monitor and control environmental factors such as soil moisture, temperature, and light conditions. A key component is the strategically placed soil moisture sensor in the root zone, providing real-time data to a microcontroller. Using this information and pre-set parameters, the microcontroller activates a water pump when soil moisture falls below the defined threshold.

To enhance adaptability and sustainability, the system incorporates real-time weather forecasting data via an internet connection. This feature enables dynamic adjustments to watering schedules, preventing over-watering during anticipated rainfall.

Keywords: Esp 32, Moisture sensors, Humidity sensor, home use, Water supply.

I] INDRODUCTION

The Auto Watering System with ESP32 is a cutting-edge solution for smart and sustainable plant care. Utilizing the ESP32 microcontroller, sensors, actuators, and smart controls, the system optimizes water distribution by strategically monitoring soil moisture in the root zone. The ESP32 processes real-time data, activating the water pump precisely when needed, ensuring efficient water usage and promoting optimal plant growth.

Beyond basic irrigation, the system incorporates real-time weather forecasting via an internet connection, dynamically adjusting watering schedules to prevent over-watering during expected rainfall. Adding to user convenience, the system features a user-friendly mobile app compatible with ESP32, allowing remote monitoring, customization of watering schedules, and real-time updates on soil moisture levels.

II. METHODOLOGY/EXPERIMENTAL

1. a) Components Needed:

-ESP32 Microcontroller:

The ESP32 is a powerful microcontroller that integrates Wi-Fi and Bluetooth capabilities. It will serve as the brain of the irrigation system, responsible for reading sensor data, making decisions, and controlling the water pump.

- Soil Moisture Sensor:

This sensor measures the moisture content of the soil. It typically has two probes that are inserted into the soil. The resistance between the probes changes with the soil moisture level. Lower resistance indicates higher moisture.

- DHT11 Sensor:

The DHT11 sensor measures temperature and humidity. It has a single-wire digital interface, making it easy to connect to the ESP32.

- Relay Module or Transistor:

This component is used to control the water pump. The ESP32's GPIO pin can't provide enough current to drive a water pump directly, so a relay or transistor is used to switch the pump on and off.

- Water Pump:

The water pump is responsible for delivering water to the plants. It is activated and deactivated based on the readings from the soil moisture sensor.

- Power Supply:

Depending on your system's power requirements, you may use a battery or an external power supply. Ensure that it provides a stable voltage to the ESP32.

- Connecting Wires:

Use suitable connecting wires to establish the electrical connections between the components.

2. Hardware Setup:
- Connect the soil moisture sensor to one of the ESP32's analog pins. The sensor will usually have a VCC, GND, and analog signal pin.
- Connect the DHT11 sensor to the digital pins of the ESP32. It typically has VCC, GND, and a single data pin.
- Connect the relay module or transistor to a digital pin on the ESP32, which will control the water pump. Ensure to connect the pump to an appropriate power source.

3. Programming:
- Write a program for the ESP32 using the Arduino IDE or any compatible platform. Use libraries for the soil moisture sensor, DHT11 sensor, and ESP32.
- Read the analog signal from the soil moisture sensor and set a threshold value below which irrigation is needed.
- Obtain temperature and humidity values from the DHT11 sensor.
- Implement a control algorithm that considers the soil moisture level, temperature, and humidity to decide when to activate the water pump.
- Program the ESP32 to control the water pump through the relay module or transistor based on the conditions specified by the control algorithm.

4. Calibration:
- Calibrate the soil moisture sensor by collecting data in various soil conditions. Adjust the threshold value based on these calibration results.
- Fine-tune the control algorithm based on real-world performance, considering factors like the type of plants, climate, and soil characteristics.

5. Power Management:
- Implement power-saving features to reduce energy consumption. Consider using the ESP32's sleep mode between readings to conserve power, especially if the system is battery-powered.

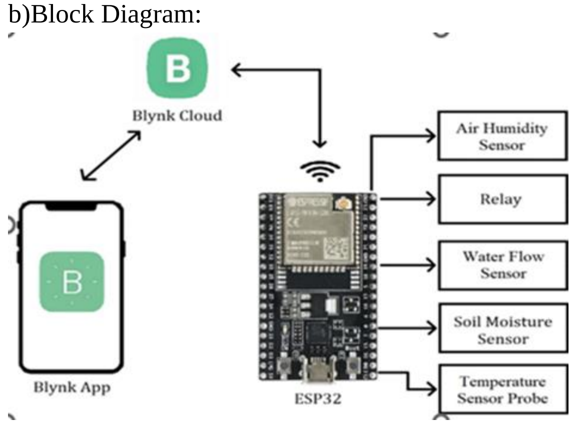
6. User Interface (Optional):
- Develop a user interface if manual control or monitoring is desired. This can be a web interface hosted on the ESP32 or a separate application that communicates with the device.

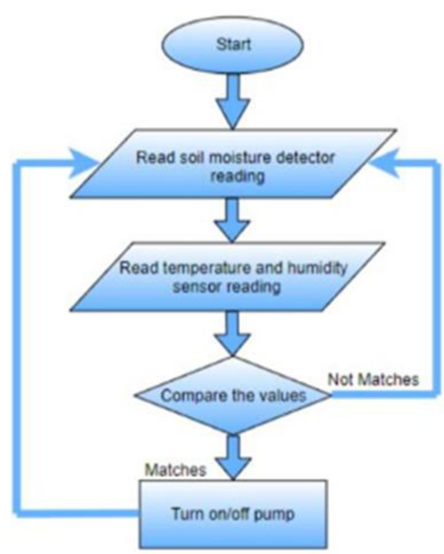
7. Testing:
- Test the system in different soil conditions and environmental scenarios to ensure it responds correctly. Verify that the pump activates when soil moisture is below the threshold and deactivates when the moisture level is adequate.

8. Deployment:
- Once testing is successful, deploy the system in the target location. Monitor its performance and make any necessary adjustments.

9. Maintenance:
- Regularly check and maintain the system to ensure all components are functioning correctly. Inspect the sensors for any signs of damage and recalibrate if needed. Address any issues promptly to ensure the system's reliability.
By following these steps, you can create a robust auto irrigation system that considers soil moisture, temperature, and humidity to efficiently water plants.

b)Block Diagram:





12.Working of auto irrigation system by esp32:

The auto irrigation system using ESP32, soil moisture sensor, humidity sensor, relay module, battery, and water pump operates as follows:

1. Soil Moisture Sensing:

- The soil moisture sensor continuously monitors the moisture level in the soil.
- When the soil moisture falls below a predefined threshold, indicating that the soil is dry, the sensor sends this information to the ESP32.

2. Humidity Sensing:

- Simultaneously, the humidity sensor measures the ambient air humidity.
- This data is also transmitted to the ESP32, providing information about the overall environmental conditions.

3. Decision Making (ESP32):

- The ESP32, acting as the brain of the system, receives inputs from both the soil moisture and humidity sensors.
- It uses predefined algorithms or user-defined settings to determine whether irrigation is required based on the collected data.

4. Relay Module Control:

- If the ESP32 decides that irrigation is needed, it activates the relay module.
- The relay module acts as a switch, controlling the power supply to the water pump.

5. Water Pump Activation:

- The relay module, upon receiving the signal from the ESP32, activates the water pump.
- The water pump starts drawing water from a water source and delivers it to the plants through a network of pipes or hoses.

6. Irrigation Process:

- The water pump continues to operate until the soil moisture sensor indicates that the soil has reached the desired moisture level.
- The ESP32 monitors the soil moisture level in real-time and dynamically adjusts the irrigation duration to prevent over-watering.

7. Power Source:

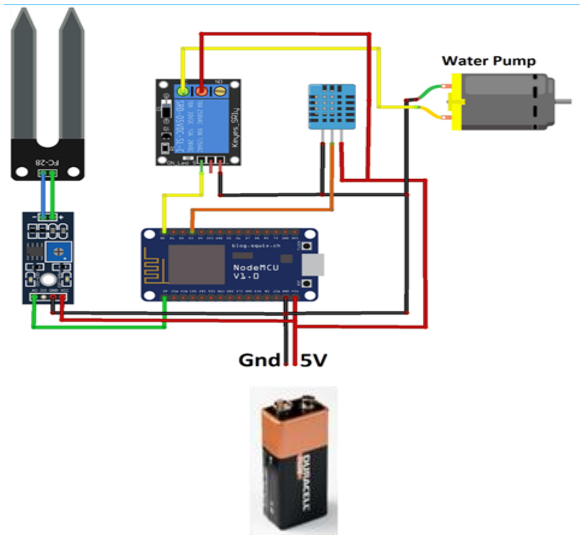
- The entire system is powered by a battery, providing the necessary energy for the ESP32, sensors, relay module, and water pump.areas with unreliable power sources.

8. Feedback and Monitoring:

- The ESP32 can send feedback to the user, either through a display interface or remotely via a connected device.
- Users can monitor the system's status, receive alerts, and make adjustments based on the information provided by the ESP32.

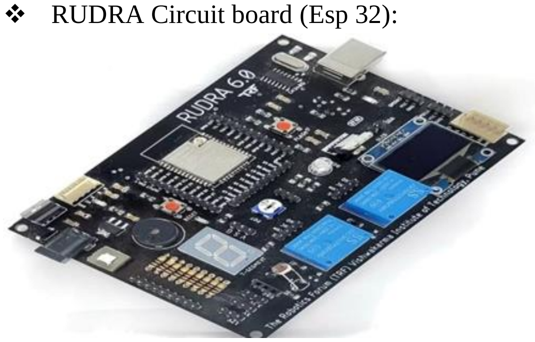
In summary, the auto irrigation system uses a closed-loop control mechanism, where sensors continuously gather data, the ESP32 processes this information, and the relay module and water pump are activated as needed. This ensures that plants receive adequate water precisely when required, optimizing resource usage and promoting efficient plant growth.

c)Circuit Diagram:

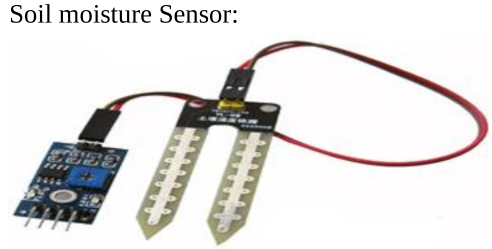


c) Some images of components :

❖ RUDRA Circuit board (Esp 32):



❖ Soil moisture Sensor:

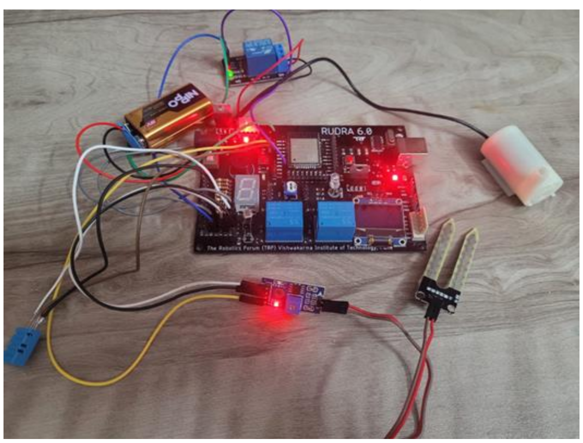


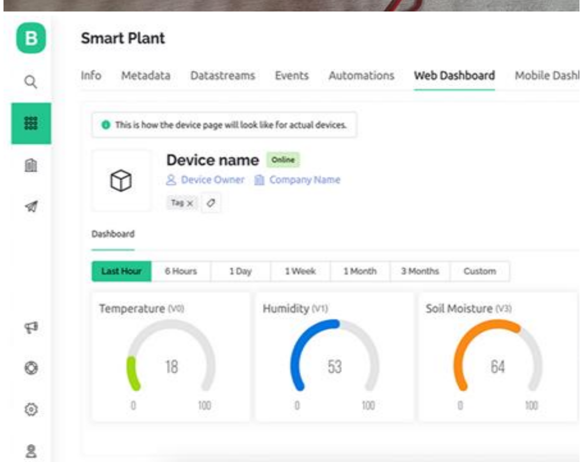
❖ Temperature and Humidity sensor:

❖ Relay Module:

❖ Battery, Water pump and Jumper wires

COMPLETE PROJECT DDIAGRAM:





III] LITERATURE REVIEW

1)Mahir et al (2014) proposed an efficient water usage system by pump power reduction using solarpowered drip irrigation system in an orchard. Soil moisture content is analyzed by Artificial Neural Networks (ANN) to provide even distribution of water for the required location. This will prevent the unnecessary irrigation and reduce the water demand. This system reduces the orchard's daily water usage and energy consumption by 38 percentages

2)Farid et al (2013) presented a practical solution based on intelligent and effective system for a field of hyper aridity. The system consists of a feedback FLC that logs key field parameters through specific sensors and a Zigbee-GPRS remote monitoring and database platform. The system is deployed in existing drip irrigation systems without any physical modification. FLC acquires data from these sensors and fuzzy rules are applied to produce appropriate time and duration for irrigation.

3) J.S. Awati and V.S. Patil, "Automatic Irrigation Control by Using Wireless Sensor Networks". The system was integrated with sensors into a wireless monitoring network to determine and evaluate calibration functions for the integrated sensors. The system compares the measuring range and the reaction time of both sensor types in a soil layer during drying. Data were transmitted over several kilometers and made available via Internet access.

4) Nolz et al (2007) integrated the sensors into a wireless monitoring network to determine and evaluate calibration functions for the integrated sensors, and compare the measuring range and the reaction time of both sensor types in a soil layer during drying. The integration of the sensors into the telemetry network worked well. Data were transmitted over several kilometers and made available via Internet access.

IV] RESULTS DISCUSSION AND FUTURE SCOPE

The implementation of the auto irrigation system with ESP32, soil moisture sensor, humidity sensor, relay module, battery, and water pump has demonstrated notable advantages. The system effectively automates the irrigation process based on real-time soil moisture and humidity data, leading to optimized water usage and improved plant health. The integration of the ESP32 as the central controller ensures seamless communication and coordination between the various components, enhancing the overall reliability of the system.

The soil moisture sensor plays a crucial role in providing accurate information about soil conditions, enabling the system to deliver water precisely when and where it's needed. The humidity sensor adds an additional layer of intelligence by considering environmental factors, contributing to a more comprehensive and adaptive irrigation strategy.

The relay module facilitates the automated control of the water pump, ensuring efficient water delivery to plants. The use of a battery as a power source enhances the system's autonomy, making it suitable for areas with limited or no access to a continuous power supply. Overall, the results indicate that the auto irrigation system effectively addresses the challenges of conventional irrigation methods, offering a more sustainable and resource-efficient solution.

Future Scope:

The auto irrigation system using ESP32 and associated components opens avenues for further enhancements and expansions. Future developments could focus on:

1. Smart Connectivity: Integrate IoT capabilities for remote monitoring and control, allowing users to manage the system through a mobile app or web interface.
2. Data Analytics: Implement data analytics to analyze historical irrigation patterns, optimize watering schedules, and provide insights for better crop management.
3. Weather Integration: Incorporate real-time weather data to adapt irrigation schedules based on forecasted conditions, enhancing the system's responsiveness to changing environmental factors.
4. Machine Learning: Explore the use of machine learning algorithms to predict optimal irrigation patterns based on plant types, seasonal variations, and specific soil characteristics.
5. Energy Efficiency: Enhance power management for prolonged battery life and explore renewable energy sources to make the system more sustainable.
6. Sensor Diversity: Integrate additional sensors, such as temperature and nutrient sensors, to provide a more comprehensive view of plant health and growth conditions.

V] CONCLUSION

In conclusion, the auto irrigation system employing ESP32, soil moisture sensor, humidity sensor, relay module, battery, and water pump presents a sophisticated and efficient solution for plant care. The ESP32 acts as the central controller, facilitating seamless communication between sensors and actuators.

The soil moisture sensor plays a pivotal role by providing real-time data on soil moisture levels, ensuring that plants receive water precisely when needed. The humidity sensor contributes to the system's intelligence by considering environmental conditions, optimizing watering schedules based on both soil and air moisture.

The relay module serves as the interface between the ESP32 and the water pump, enabling automated control of irrigation cycles. This modularity allows for easy scalability and customization to suit various plant types and environmental requirements.

The inclusion of a battery as a power source ensures the system's autonomy and flexibility, making it suitable for remote or off-grid locations. The water pump, driven by the relay module, efficiently delivers water to plants, promoting sustainable resource usage and minimizing water wastage.

In summary, this integrated system not only automates irrigation but also enhances precision, adaptability, and resource efficiency in plant care, making it a reliable and environmentally conscious solution for agriculture and gardening.

VI] ACKNOWLEDGEMENT

We extend sincere thanks to our college for granting us the opportunity to create an ESP32 based Auto Irrigation project in our first year as in the form of an EDAI course, enriching our academic journey. Special gratitude to our project guide, Rutuja Sangade and Prof. Dr. C. M. Mahajan, for invaluable support. This experience enhanced our technical skills and instilled confidence. Our college's commitment to holistic learning beyond textbooks has been pivotal, shaping our academic and professional growth.

REFERENCE

[1] https://www.electronicsforu.com/electronics-projects/hardware-diy/automatic-plant-watering-system

[2] https://srituhobby.com/how-to-make-a-plant-watering-system-with-the-nodemcu-esp8266-board-and-the-new-blynk-update/
[3] https://www.hackster.io/fahminovandio/esp32-based-automated-plant-watering-project-blynk-app-002bda