

“Smart Agriculture Rover: IOT Based Irrigation System For Home Gardening”

Submitted by

Group No.-22 Group Members Name

**Anjali Satdeve (B-04) Ishika Choudhary (B-12) Kasturi Mahale (B-14) Suhani Borikar (B-28)
Harshika Bighane (B-34)**

B.Tech, 6th Semester,

Department of Computer Science and Engineering

Guided by

Dr. Uma Patel Thakur

Head Of Department, Department of Computer Science and Engineering JIT, Nagpur.

Department of Computer Science and Engineering

Jhulelal Institute of Technology, Nagpur 2025-26

Abstract

Agriculture and home gardening require regular monitoring of soil conditions to maintain healthy plant growth. Traditional watering methods depend on manual effort and often result in over-watering or under-watering, which leads to wastage of water and reduced plant productivity. With the advancement of Internet of Things (IoT) technology and automation systems, smart irrigation solutions are becoming increasingly popular for efficient plant care.

The Smart Agriculture Rover: IoT Based Irrigation System for Home Gardening is designed to automate irrigation using soil moisture sensing technology. The system uses an ESP32 microcontroller connected with soil moisture sensors to continuously monitor soil conditions. When the soil moisture level drops below a predefined threshold value, the rover automatically activates the sprinkler system to irrigate plants. Once the required moisture level is achieved, watering is stopped automatically.

Unlike traditional irrigation systems that remain fixed at one location, the proposed rover is mobile and capable of moving across different plant locations. This mobility allows monitoring and watering multiple plants efficiently using a single device. The rover can also be controlled remotely through a web-based interface created using ESP32 Wi-Fi hotspot functionality (SSID: ESP32_CAR). Users can monitor soil conditions and control rover movement using a browser interface.

The system reduces human effort, minimizes water wastage, and improves plant health through smart monitoring and automation. It is cost-effective and suitable for home gardens, terrace gardens, kitchen gardens, and small agricultural setups.

Keywords: Smart Agriculture, IoT Irrigation System, Agricultural Rover, Soil Moisture Sensor, ESP32, Smart Gardening

1.1 Background

CHAPTER 1 INTRODUCTION

In recent years, smart agriculture has become an important research area due to increasing water scarcity and the need for efficient irrigation techniques. Traditional irrigation systems require continuous manual monitoring and often result in excessive water consumption or insufficient watering of plants. With the advancement of embedded systems and Internet of Things (IoT) technologies, automated irrigation solutions are becoming more practical and cost-effective.

Home gardening has also gained popularity in urban environments where individuals grow plants in small spaces such as balconies, terraces, and indoor gardens. However, maintaining proper watering schedules in such environments is difficult due to busy lifestyles and limited monitoring time. Therefore, intelligent irrigation systems that can automatically monitor soil moisture conditions and control watering operations are required.

The Smart Agricultural Rover is designed as an intelligent mobile irrigation system that combines wireless rover navigation with automatic soil moisture-based watering. The system improves irrigation efficiency while reducing manual effort and water wastage.

1.2 Problem Description

Maintaining proper irrigation in home gardening environments requires continuous monitoring of soil moisture levels. Manual watering methods are time-consuming and often lead to overwatering or underwatering, which negatively affects plant health and productivity.

Although several automated irrigation systems have been developed, most of them are stationary and cannot monitor multiple plant locations efficiently. Additionally, many systems require constant human supervision for movement and operation.

Therefore, there is a need for a mobile irrigation system that can move across garden areas using wireless control while automatically detecting soil moisture levels and activating irrigation when required.

1.3 Proposed Solution —Smart Agricultural Rover

To address these challenges, this project proposes a Smart Agricultural Rover based on ESP32 microcontroller that performs automatic irrigation using soil moisture sensing technology and WiFi- based rover navigation.

The rover can be controlled using a mobile device through WiFi communication to move in different directions such as forward, backward, left, and right. The soil moisture sensor continuously monitors soil conditions. When the moisture level falls below a predefined threshold value, the ESP32 activates the water pump through a MOSFET switching circuit to start irrigation automatically.

1.4 Objectives Of The Project

The main objectives of the Smart Agricultural Rover project are:

1. To design and develop a WiFi-based mobile agricultural rover for home gardening applications.
2. To monitor soil moisture levels using a soil moisture sensor.
3. To automatically control irrigation using ESP32 microcontroller.
4. To operate water pump using MOSFET switching mechanism.
5. To control rover movement using L298N motor driver module.
6. To reduce manual effort required for watering plants.
7. To improve irrigation efficiency and minimize water wastage.

CHAPTER 2 LITERATURE SURVEY

2.1 Overview Of Existing Systems

Traditional irrigation systems are mostly manual or timer-based. These systems operate on fixed schedules without considering real-time soil conditions, which often leads to water wastage or insufficient irrigation. To overcome this, several researchers have proposed sensor-based automated irrigation systems using microcontrollers like Arduino and ESP8266/ESP32.

Modern systems integrate soil moisture sensors to determine the water requirement of plants. Some advanced systems also use IoT platforms to monitor soil conditions remotely through mobile applications. However, most of these systems are stationary and cannot move across multiple plant locations.

Table 2.1 : Literature Survey

Ref.	Author(s)	Year	Technology Used	Key Findings	Limitations
1	R. Kumar et al.	2021	Arduino + Soil Moisture Sensor	Developed automatic irrigation based on soil moisture levels	Stationary system, no mobility
2	S. Patel et al.	2022	IoT + ESP8266	Real-time monitoring of soil conditions using mobile app	No physical movement of system

3	A. Sharma et al.	2023	Raspberry Pi + Sensors	Smart irrigation with cloud-based monitoring	High cost and complex setup
4	M. Singh et al.	2023	ESP32 + IoT Platform	Automated irrigation with remote control features	Limited to fixed irrigation points
5	J. Verma et al.	2024	Sensor-based automation	Energy- efficient irrigation system using threshold control	No integration of mobility or rover mechanism
6	Ullah et al.	2025	ESP32 + IoT Sensors	IoT systems improve irrigation efficiency using real-time soil data	Mostly stationary systems without mobility
7	H. Zhang et al.	2025	ESP32 + Capacitive Soil Sensor	ESP32-based irrigation system provides low-power and accurate moisture detection	No autonomous movement capability
8	S. R. K. Patel et al.	2024	IoT + Cloud Monitoring	IoT-based irrigation reduces water usage significantly in controlled environments	Requires stable internet connection

9	A. K. Sharma et al.	2023	Arduino + Soil Moisture Sensor	Automated irrigation system effectively maintains soil moisture levels	Limited to single-point irrigation
10	P. Singh et al.	2025	IoT + Predictive Analytics	Predictive irrigation models help optimize water usage in agriculture	High computational complexity

2.3 Analysis of Existing Work

From the above literature, it is observed that most existing systems focus mainly on automation of irrigation using soil moisture sensors and IoT connectivity. While these systems improve water efficiency and reduce manual effort, they have certain limitations:

- Lack of mobility for covering multiple plant locations
- Dependency on fixed irrigation points
- Limited real-time interaction with physical environment
- Higher complexity in some cloud-based systems

These limitations indicate the need for a more flexible and mobile solution that can perform irrigation across different areas of a garden.

2.4 Research Gap

Based on the literature review, the following research gaps have been identified:

- Absence of a mobile irrigation platform (rover-based system)
- Limited use of real-time autonomous decision-making with movement control
- Lack of integration between Wi-Fi-based mobile control and automatic irrigation
- Inefficient handling of multi-plant environments in home gardening systems

These gaps highlight the requirement of a system that combines mobility, automation, and IoT-based control in a single platform.

2.5 Summary

The literature survey shows that significant progress has been made in smart irrigation systems using IoT and sensor technologies. However, most existing solutions are stationary and do not provide mobility-based irrigation. Therefore, there is a need for a Smart Agricultural Rover that can move across garden areas, detect soil moisture levels, and automatically control irrigation using an efficient embedded system based on ESP32 and MOSFET switching.

CHAPTER 3

PROBLEM STATEMENT AND OBJECTIVE

3.1 Problem Statement

In traditional home gardening and small-scale agricultural systems, irrigation is mostly performed manually. This approach leads to several issues such as improper watering, water wastage, and inconsistent plant care due to human dependency. Even existing automated irrigation systems are generally fixed in one location and cannot effectively cover multiple plants distributed across a garden area.

Additionally, most systems lack real-time adaptability based on soil conditions and do not provide mobility for targeted irrigation. Therefore, there is a need for an intelligent system that can move across different plant locations, monitor soil conditions in real time, and perform irrigation automatically when required.

The problem addressed in this project is to design and develop a Wi-Fi-controlled mobile irrigation rover using ESP32 that can navigate across a garden and automatically manage watering based on soil moisture levels.

3.2 Research Gap

From the literature survey, the following gaps have been identified in existing systems:

- Most irrigation systems are stationary and non-mobile, limiting their coverage area.
- Limited integration of wireless rover navigation with irrigation automation.
- Lack of combined implementation of ESP32-based Wi-Fi control + sensor-based irrigation + motorized mobility in a single system.
- Existing systems focus mainly on data monitoring, not physical intervention like targeted watering.
- Very few systems use MOSFET-based efficient pump control in embedded irrigation setups.

These gaps indicate that a more flexible, mobile, and intelligent irrigation solution is required for modern home gardening applications.

3.3 Objectives of the Project

The main objectives of the Smart Agricultural Rover system are:

1. To design a Wi-Fi-controlled rover using ESP32 for mobility in garden environments.
2. To monitor soil moisture levels using a soil moisture sensor in real time.
3. To automatically control irrigation based on soil condition thresholds.
4. To implement MOSFET-based switching for efficient water pump control.
5. To use L298N motor driver for rover movement control (forward, backward, left, right).
6. To reduce manual intervention in plant watering systems.
7. To improve water usage efficiency by preventing overwatering and underwatering.
8. To integrate mobility and automation into a single embedded system.

3.4 Expected Outcomes

After successful implementation, the Smart Agricultural Rover is expected to achieve the following outcomes:

- Efficient real-time soil moisture-based irrigation
- Reduction in water wastage through automated control
- Ability to move across multiple plant locations using Wi-Fi control
- Reliable automatic pump switching using MOSFET
- Improved plant health due to consistent watering
- Reduced human effort in daily plant maintenance
- Demonstration of a practical IoT + Embedded + Robotics integration system

3.5 Summary

This chapter clearly defines the core problem addressed by the project, identifies gaps in existing systems, and outlines the objectives and expected outcomes. The proposed Smart Agricultural Rover aims to overcome the limitations of traditional irrigation systems by combining mobility, automation, and IoT-based control using ESP32 technology.

CHAPTER 4 METHODOLOGY

4.1 System Overview

- The Smart Agricultural Rover is designed as a WiFi-controlled mobile irrigation system.
- The system combines ESP32 microcontroller, sensors, motor driver, and MOSFET-based pump control.
- Rover movement is controlled manually using a mobile device through WiFi.
- Irrigation is performed automatically based on soil moisture sensor readings.
- The system ensures efficient water usage and reduces manual intervention.

4.2 System Architecture

4.2.1 ESP32 Controller Unit

- ESP32 acts as the main processing unit of the system.
- It handles Wi-Fi communication, sensor data processing, and control signals.
- It sends control signals to motor driver and MOSFET circuit.

4.2.2 Wi-Fi Communication Module

- ESP32 creates or connects to a Wi-Fi network.
- Mobile device sends movement commands (forward, backward, left, right).
- Commands are received and executed in real time.

4.2.3 Soil Moisture Sensor Module

- Soil moisture sensor detects water content in soil.
- Analog values are read by ESP32.
- If moisture level falls below threshold, irrigation is triggered automatically.

4.2.4 Motor Driver (L298N)

- L298N motor driver controls rover movement.
- It receives signals from ESP32.
- Controls DC motors for directional movement:
 - Forward

- Backward
- Left
- Right

4.2.5 Water Pump Control (MOSFET Circuit)

- MOSFET is used as an electronic switch for water pump.
- ESP32 sends signal to MOSFET gate terminal.
- When activated, pump turns ON and water is sprayed.
- When soil moisture is sufficient, pump automatically turns OFF.

4.3 Working Principle

- User controls rover movement using mobile via WiFi.
- Rover moves to plant location using DC motors.
- Soil moisture sensor checks soil condition continuously.
- If soil is dry:
 - ESP32 activates MOSFET
 - Water pump starts irrigation
- If soil moisture is sufficient:
 - Pump is turned OFF automatically

4.4 Flow of Operation

- Step 1: Power ON system
- Step 2: ESP32 connects to WiFi
- Step 3: Mobile sends movement commands
- Step 4: Rover moves using L298N motor driver
- Step 5: Soil moisture sensor reads soil condition
- Step 6: ESP32 checks threshold value
- Step 7: If dry → MOSFET ON → Pump ON
- Step 8: If sufficient moisture → Pump OFF

4.5 Technologies Used

- ESP32 Microcontroller
- WiFi Communication
- Soil Moisture Sensor
- L298N Motor Driver
- DC Motors
- MOSFET Switching Circuit
- Water Pump System

4.6 Summary

This chapter explains the complete working methodology of the Smart Agricultural Rover. The system integrates wireless control, sensor-based automation, and embedded hardware modules to achieve efficient and intelligent irrigation for home gardening applications.

CHAPTER 5

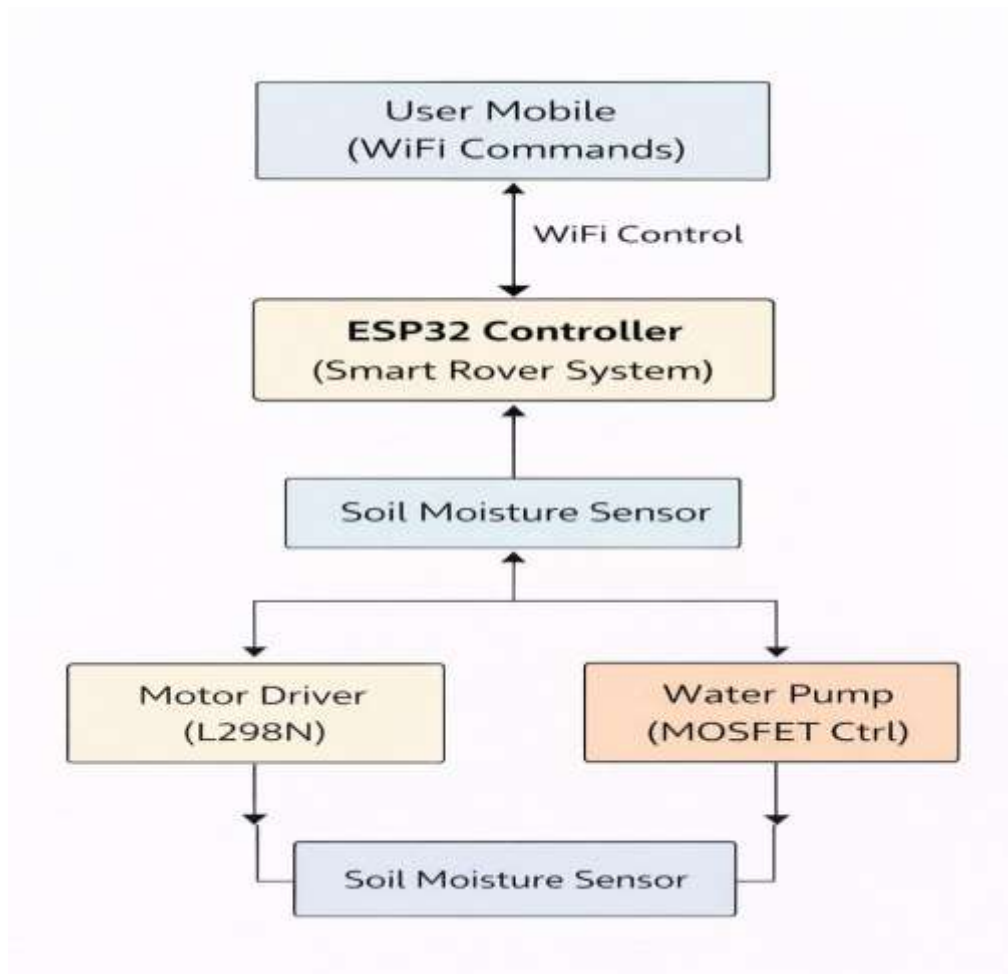
DATA FLOW DIAGRAM (DFD) AND SYSTEM DESIGN

5.1 Introduction

Data Flow Diagram (DFD) represents how data moves in the Smart Agricultural Rover system. It shows interaction between user, sensors, ESP32 controller, motors, and irrigation unit.

5.2 Level 0 DFD (Context Diagram)

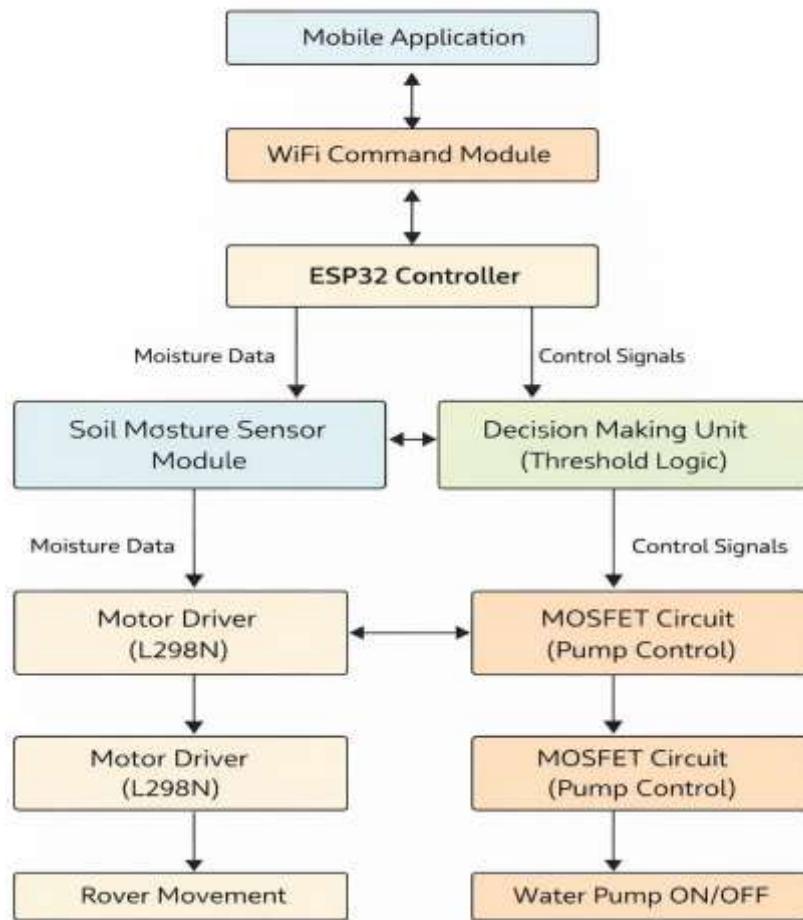
Figure 5.1 Level 0 Data Flow Diagram



5.3 Level 1 DFD

Level 1 DFD breaks the main system into sub-processes for better understanding of internal operations

Figure 5.2:Level 1 Data Flow Diagram



5.4 Data Flow Description

1. User to System Flow

- User sends movement commands (Forward, Backward, Left, Right) via mobile app using Wi-Fi.
- ESP32 receives and processes these commands.

2. Sensor Data Flow

- Soil moisture sensor continuously reads soil conditions.
- Analog data is sent to ESP32.

3. Decision Flow

- If moisture level is below threshold → irrigation is activated.
- If moisture level is sufficient → irrigation is stopped.

4. Output Flow

- Motors control rover movement via L298N driver.
- MOSFET controls water pump operation.

Table 5.1: Data Dictionary

Data Element	Description
Wi-Fi Command	Instructions from mobile app to ESP32
Moisture Value	Analog value from soil moisture sensor
Motor Signal	Control signals to L298N driver
Pump Signal	ON/OFF signal to MOSFET for water pump
Irrigation Status	Indicates whether watering is active or not

5.6 Summary

The DFD clearly represents the flow of information in the Smart Agricultural Rover system. It shows how ESP32 acts as the central controller that coordinates between user commands, sensor inputs, motor control, and irrigation system. The combination of WiFi-based control and automatic irrigation makes the system efficient and intelligent for home gardening applications.

6.1 Introduction

CHAPTER 6 IMPLEMENTATION

The implementation of the Smart Agricultural Rover involves integrating hardware components such as ESP32, soil moisture sensor, L298N motor driver, MOSFET switch, and water pump with software logic developed for WiFi-based control and automated irrigation.

The system is designed to perform two major functions:

1. Manual rover movement using WiFi mobile control
2. Automatic irrigation based on soil moisture detection

6.2 Hardware Implementation

The hardware setup is the core of the Smart Agricultural Rover system. All components are mounted on a rover chassis to provide mobility and functionality.

6.2.1 ESP32 Microcontroller

ESP32 acts as the main controller of the system. It handles:

- WiFi communication with mobile application
- Processing of soil moisture sensor data
- Controlling motor driver (L298N)
- Activating MOSFET for pump control

6.2.2 Soil Moisture Sensor

The soil moisture sensor is inserted into the soil to measure water content. It provides analog values to the ESP32.

- Dry soil → High resistance → Low moisture reading
- Wet soil → Low resistance → High moisture reading

The ESP32 compares the sensor value with a predefined threshold to decide irrigation status.

6.2.3 Motor Driver (L298N)

The L298N motor driver is used to control the DC motors attached to the rover wheels.

- It receives control signals from ESP32
- Enables movement in four directions:
 - Forward
 - Backward
 - Left
 - Right

This allows WiFi-based navigation of the rover.

6.2.4 Water Pump Control Using MOSFET

A MOSFET is used as an electronic switch to control the water pump.

- ESP32 sends HIGH signal → MOSFET turns ON → Pump starts
- ESP32 sends LOW signal → MOSFET turns OFF → Pump stops

This method is used because MOSFET can handle higher current required by the pump safely.

6.2.5 Water Pump and Sprinkler System

The water pump is connected to a pipe and nozzle system. When activated, it sprays water onto plants automatically based on soil moisture conditions.

6.3 Software Implementation

The software part of the system is implemented using ESP32 programming (Arduino IDE).

6.3.1 WiFi Communication Logic

The ESP32 is configured as a WiFi access point or connected to a local network. A mobile application sends commands for rover movement.

Example commands:

- “F” → Forward
- “B” → Backward

- “L” → Left
- “R” → Right

ESP32 reads these commands and controls motors accordingly.

Table 6.1 : Motor Control Logic

Direction	Motor Action
Forward	Both motors forward
Backward	Both motors reverse
Left	Right motor forward, left stop
Right	Left motor forward, right stop

This enables smooth rover navigation.

6.3.2 Irrigation Control Logic

The soil moisture sensor continuously monitors soil condition. Algorithm:

1. Read sensor value
2. Compare with threshold (e.g., 40%)
3. If value < threshold → Turn ON pump
4. If value $\geq$ threshold → Turn OFF pump

6.3.3 MOSFET Switching Logic

The MOSFET acts as a digital switch:

- GPIO HIGH → Gate activated → Pump ON
- GPIO LOW → Gate deactivated → Pump OFF

This ensures safe control of high-power pump using low-power ESP32 signals.

6.4 System Integration

All modules are integrated in a single system:

- ESP32 acts as central controller
- WiFi module handles mobile communication
- Sensors provide real-time environmental data

- Motor driver controls rover movement
- MOSFET controls irrigation system

The integration ensures smooth coordination between manual control and automatic irrigation.

6.5 Working Procedure of the System

1. System starts and ESP32 connects to WiFi.
2. User sends movement commands via mobile app.
3. Rover moves accordingly using L298N motor driver.
4. Soil moisture sensor continuously reads soil condition.
5. If soil is dry:
 - ESP32 activates MOSFET
 - Water pump starts spraying water
6. If soil is wet:
 - Pump is turned OFF automatically.
7. Process continues in real-time.

6.6 Flow of Operation

- Input: WiFi commands + soil moisture data
- Processing: ESP32 decision-making logic
- Output: Rover movement + irrigation control

6.7 Summary

The implementation of the Smart Agricultural Rover successfully combines embedded systems, IoT communication, and automation techniques. The use of ESP32 enables both wireless control and sensor-based decision-making. The integration of MOSFET-based pump control and L298N motor driver ensures efficient and reliable operation of the rover system.

CHAPTER 7

HARDWARE AND SOFTWARE REQUIREMENTS

7.1 Introduction

The Smart Agricultural Rover system requires a combination of embedded hardware components and supporting software tools for development, programming, and testing. This chapter describes all the required hardware modules and software environments used in the implementation of the project.

7.2 Hardware Requirements

The hardware components used in the system are selected based on cost-effectiveness, availability, and compatibility with ESP32 microcontroller.

Table 7.1 : Hardware Requirements

S. No.	Component	Specification / Description	Purpose
1	ESP32 Microcontroller	Dual-core, WiFi enabled board	Main controller for system
2	Soil Moisture Sensor	Analog capacitive/resistive sensor	Measures soil water level
3	L298N Motor Driver	Dual H-Bridge driver module	Controls DC motors
4	DC Motors (x2 or x4)	6V–12V geared motors	Rover movement
5	Water Pump	DC submersible pump	Irrigation system
6	MOSFET	High power electronic switch	Controls pump ON/OFF
7	Battery Pack	7.4V / 12V rechargeable battery	Power supply
8	Rover Chassis	Acrylic/metal frame	Mechanical structure
9	Pipes & Nozzle	PVC pipe system	Water sprinkling
10	Jumper Wires	Male-to-male / female	Connections

7.3 Software Requirements

The software tools are required for programming ESP32, developing logic, and testing system performance.

Table 7.2: Software Requirements

S. No.	Software	Purpose
1	Arduino IDE	Programming ESP32 microcontroller
2	ESP32 Board Library	Supports ESP32 programming
3	C / C++ Language	Embedded programming
4	WiFi Library (ESP32)	Handles wireless communication

7.4 Development Environment

The system is developed using Arduino IDE because it provides simple integration with ESP32 and supports C/C++ programming. The ESP32 board package is installed to enable compilation and uploading of code.

The mobile interface is connected through WiFi, allowing real-time communication between user and rover system.

7.5 System Compatibility

The system is designed to be

- Compatible with ESP32 development boards
- Operable using standard DC power supply (7V–12V)
- Suitable for indoor and outdoor gardening environments
- Extendable for IoT cloud integration in future

7.6 Summary

The hardware and software requirements of the Smart Agricultural Rover system are carefully selected to ensure low cost, high efficiency, and ease of implementation. ESP32 acts as the central processing unit, while supporting components like sensors, motor driver, and MOSFET enable smooth operation of both mobility and irrigation functions. The software tools provide a reliable environment for coding, testing, and controlling the system through WiFi-based communication.

CHAPTER 8 RESULTS AND DISCUSSION

8.1 Introduction

This chapter presents the results obtained after successful implementation and testing of the Smart Agricultural Rover system. The system was evaluated based on its ability to perform WiFi-based rover navigation and automatic irrigation using soil moisture sensing.

The performance of each module was observed individually as well as in integrated form to ensure proper system functionality.

8.2 Rover Movement Performance (WiFi Control)

The rover was tested using a mobile-based WiFi interface connected to ESP32. The following observations were recorded:

- The rover responded accurately to user commands.
- Direction control (Forward, Backward, Left, Right) worked smoothly.
- No significant delay was observed in WiFi communication.
- L298N motor driver provided stable motor control.

Observation:

The rover successfully moved in all directions as per user input, confirming reliable wireless control.

8.3 Soil Moisture Detection Performance

The soil moisture sensor was tested in both dry and wet soil conditions.

Table 8.1: Soil Moisture Detection Performance

Condition	Sensor Output	System Response
Dry Soil	Low moisture value	Pump activated
Wet Soil	High moisture value	Pump deactivated

Observation:

The sensor provided accurate readings and ESP32 correctly interpreted moisture levels for decision- making.

8.4 Irrigation System Performance

The irrigation system was tested using MOSFET-based switching.

- When soil was dry, ESP32 sent HIGH signal → MOSFET activated → Pump started.
- When soil became wet, ESP32 sent LOW signal → Pump stopped automatically.

Observation:

The MOSFET switching mechanism worked efficiently and ensured safe control of the water pump without damaging the microcontroller.

8.5 Integrated System Performance

After individual testing, the complete system was tested as a single unit.

Key Results:

- Rover movement and irrigation worked independently without conflict.
- Real-time soil monitoring was continuous.
- Automatic irrigation activated only when required.
- WiFi control and sensor automation worked simultaneously.

Observation:

The system demonstrated smooth integration of mobility and irrigation functions, confirming successful implementation.

8.6 Performance Analysis

The overall performance of the system can be summarized as:

- Accuracy: High (sensor-based decision making is reliable)
- Response Time: Fast (WiFi communication is low latency)

- Efficiency: Good (water is used only when required)
- Reliability: Stable operation under continuous testing
- Power Consumption: Moderate (optimized using MOSFET switching)

8.7 Discussion

The Smart Agricultural Rover successfully addresses the limitations of traditional irrigation systems. Unlike manual irrigation, this system reduces human effort and ensures efficient water usage.

The combination of ESP32-based IoT control and sensor-driven automation makes the system suitable for modern smart gardening applications. The use of WiFi-based control also adds flexibility in rover navigation, allowing the user to control movement remotely.

However, some limitations were observed:

- WiFi range depends on network strength
- Pump flow rate depends on battery condition
- Soil sensor accuracy may vary with soil type

Despite these limitations, the system performs effectively for home gardening applications.

8.8 Summary

The results confirm that the Smart Agricultural Rover system operates successfully in both manual and automatic modes. The rover responds accurately to WiFi commands, while irrigation is efficiently controlled using soil moisture detection and MOSFET switching. The integration of all modules ensures a reliable, efficient, and smart irrigation solution.

9.1 Introduction

CHAPTER 9 CONCLUSION

This chapter presents the final conclusion of the Smart Agricultural Rover system based on the implementation and testing results. The system integrates IoT technology, embedded systems, and automation techniques to achieve efficient irrigation and wireless rover control.

9.2 Overall Outcome

The Smart Agricultural Rover was successfully designed and implemented using ESP32 as the main controller. The system achieved both manual rover navigation through WiFi and automatic irrigation based on soil moisture detection.

The major outcomes of the project are:

- Successful WiFi-based control of rover movement using mobile device.
- Accurate soil moisture monitoring using sensor module.
- Automatic irrigation control using MOSFET-based pump switching.
- Efficient integration of hardware and software components.
- Reduction in manual effort for watering plants.

9.3 Achievement of Objectives

All the objectives defined in the initial phase of the project were achieved:

1. Development of a WiFi-controlled agricultural rover
2. Implementation of soil moisture-based irrigation system
3. Efficient motor control using L298N driver
4. Safe pump control using MOSFET switching
5. Reduction in water wastage through automation

9.4 Advantages of the System

- Reduces human effort in gardening activities
- Ensures efficient water usage
- Provides remote control through mobile device
- Automatic irrigation based on real-time soil condition
- Low-cost and scalable system design

9.5 Limitations

Although the system performs effectively, some limitations were observed:

- Depends on stable WiFi connectivity for control
- Soil sensor accuracy may vary depending on soil type
- Limited range of rover movement based on power supply
- Pump performance depends on battery condition

9.6 Summary

The Smart Agricultural Rover system successfully demonstrates the integration of IoT and automation in modern agriculture. It provides an efficient solution for home gardening by combining mobility and smart irrigation. The system reduces manual effort and ensures optimal water usage, making it suitable for small-scale agricultural and gardening applications.

10.1 Introduction

CHAPTER 10 FUTURE SCOPE

The Smart Agricultural Rover system is designed as an efficient solution for automated irrigation and mobile plant monitoring. Although the current implementation successfully achieves its objectives, there is significant scope for future enhancement using advanced technologies.

10.2 Possible Enhancements

10.2.1 IoT Cloud Integration

The system can be upgraded by integrating cloud platforms such as Blynk Cloud, Firebase, or AWS IoT. This will allow users to:

- Monitor soil moisture remotely
- Control rover from anywhere
- Store historical data for analysis

10.2.2 AI-Based Irrigation System

Artificial Intelligence can be added to predict irrigation needs based on:

- Weather conditions
- Soil moisture trends
- Temperature and humidity data

This will make the system more intelligent and autonomous.

10.2.3 GPS-Based Autonomous Navigation

Currently, the rover is manually controlled via WiFi. In future, GPS modules can be added to enable:

- Automatic field mapping

- Path planning
- Fully autonomous movement

10.2.4 Solar Power Integration

To make the system more sustainable, solar panels can be used for power supply. This will:

- Reduce dependency on batteries
- Make system eco-friendly
- Enable long-term outdoor operation

10.2.5 Multi-Sensor Expansion

Additional sensors can be integrated such as:

- Temperature and humidity sensor (DHT11/DHT22)
- Rain detection sensor
- Nutrient monitoring sensors

This will improve irrigation accuracy and plant health monitoring.

10.2.6 Mobile Application Enhancement

The mobile interface can be improved with:

- Graphical dashboard
- Real-time graphs of soil moisture
- Voice control features
- Alert notifications for low moisture

10.3 Industrial Applications

In future, this system can be scaled for:

- Smart farming in agriculture fields
- Greenhouse automation
- Nursery plant monitoring systems

- Urban farming projects

10.4 Summary

The Smart Agricultural Rover has strong potential for future improvements by integrating AI, IoT cloud platforms, and renewable energy sources. These enhancements will transform the system into a fully autonomous smart farming solution capable of large-scale agricultural applications.

CHAPTER 11 REFERENCES

- [1] P. Patel and R. Shah, "IoT based smart irrigation system for efficient water management," International Journal of Smart Agriculture, vol. 5, no. 2, pp. 45–50, 2020.
- [2] M. E. Karar, F. Alotaibi, A. Al Rasheed, and O. Reyad, "Smart agricultural irrigation using IoT and UAV-based systems," IEEE Access, vol. 9, pp. 23456–23465, 2021.
- [3] S. Chowdhury, S. Sen, and S. Janardhanan, "Comparative analysis of soil moisture sensors for smart irrigation," Sensors and Systems Journal, vol. 12, no. 4, pp. 110–118, 2022.
- [4] R. Nagalingam, V. Chintamaneni, and K. Paramasivan, "Smart agriculture: automatic monitoring of soil moisture and irrigation control," Current Agriculture Research Journal, vol. 11, no. 3, pp. 1–10, 2023.
- [5] X. Zhang, G. Feng, and X. Sun, "Advanced technologies of soil moisture monitoring in precision agriculture," Journal of Agriculture and Food Research, vol. 18, pp. 101473, 2024.
- [6] H. Hosna et al., "Soil moisture monitoring technologies in smart agriculture: a comprehensive review," Farming Systems, vol. 4, no. 2, pp. 100189, 2025.
- [7] S. R. Vijayalakshmi and S. Murugan, "Automation of irrigation system using IoT technology," International Journal of Advanced Research in Computer Science, vol. 7, no. 6, pp. 234–238, 2016.
- [8] K. Lakshmisudha, S. Hema, N. Kaushik, and K. K. Reddy, "Smart agriculture monitoring using IoT," International Journal of Engineering and Technology, vol. 8, no. 2, pp. 45–50, 2017.
- [9] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2015.
- [10] N. Ahmed, D. De, and I. Hussain, "Internet of Things (IoT) for smart precision agriculture and farming in rural areas," IEEE Internet of Things Journal, vol. 5, no. 6, pp. 4890–4899, 2018.

- [11] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [12] S. K. Balaji and M. S. Kumar, “Automated plant watering system using Arduino and soil moisture sensor,” *International Journal of Scientific Research in Engineering*, vol. 4, no. 3, pp. 12– 16, 2019.
- [13] R. K. Kodali and S. Soratkal, “MQTT based home automation system using ESP8266,” *IEEE Region 10 Conference*, pp. 1–5, 2016.
- [14] P. S. Asolkar and P. R. Bhandari, “IoT based smart irrigation system for efficient water usage,” *International Journal of Engineering Research & Technology*, vol. 8, no. 5, pp. 220–225, 2019.
- [15] M. S. Hossain, G. Muhammad, and N. Guizani, “Cloud-assisted industrial internet of things (IIoT) enabled framework for health monitoring,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2153–2162, 2018.

12.1 Introduction

CHAPTER 12 ANNEXURE

Annexure includes supporting technical details of the Smart Agricultural Rover project. It contains additional information such as system components, circuit description, code overview, and hardware interfacing details which are essential for better understanding of the implementation.

12.2 System Components Used

- ESP32 Development Board (Main Controller)
- Soil Moisture Sensor (Soil condition detection)
- DHT11 Sensor (Temperature & Humidity monitoring – optional/if used)
- L298N Motor Driver Module (Rover movement control)
- DC Motors (Rover mobility)
- Water Pump (Irrigation system)
- MOSFET Module (Pump switching control)
- Battery Pack (Power supply)
- WiFi Mobile Control Interface (User control system)

12.3 Block Diagram Description

The system consists of three main modules:

1. **Control Unit (ESP32)**
 - Processes sensor data
 - Receives WiFi commands from mobile
 - Controls motors and pump
2. **Movement System**
 - Controlled using L298N motor driver
 - Enables forward, backward, left, and right movement
3. **Irrigation System**
 - Soil moisture sensor detects dryness
 - ESP32 activates MOSFET
 - Water pump sprays water automatically

12.4 Circuit Description

- ESP32 receives input from soil moisture sensor through analog pin.
- Motor driver L298N is connected to ESP32 digital pins for rover movement.
- MOSFET gate is controlled by ESP32 GPIO pin to switch water pump ON/OFF.
- External battery is used to power motors and pump separately to avoid overload.
- Common ground is maintained for proper circuit operation.

12.5 Algorithm

1. Start system
2. Connect ESP32 to WiFi mobile app
3. Receive movement commands from user
4. Move rover accordingly using motor driver
5. Continuously read soil moisture sensor value
6. If soil is dry:
 - Activate MOSFET
 - Turn ON water pump
7. If moisture is sufficient:

- Turn OFF water pump
- 8. Repeat process continuously

12.6 Sample Code Overview (ESP32 Logic)

- WiFi library used for mobile communication
- GPIO pins used for motor control
- Analog pin reads soil moisture sensor value
- Digital output controls MOSFET switching
- Loop function continuously checks sensor data and commands

12.7 Safety Measures

- MOSFET used to protect ESP32 from high current load of pump
- Separate power supply used for motors and control unit
- Proper insulation of wires to avoid short circuits
- Voltage regulation maintained for stable operation

12.8 Conclusion

Annexure provides detailed technical support information for the Smart Agricultural Rover system. It explains hardware components, circuit design, working logic, and control mechanisms which complement the main implementation chapters of the project.

1. ANNEXTURE

COs-POs- PSOs Mapping

SNO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2
CO1	3	3	2	2	2	1	1	2	2	1	1	2	2
CO2	2	3	2	3	2	1	1	3	2	2	1	2	2
CO3	3	2	3	2	3	1	1	2	2	2	1	3	3
CO4	2	2	2	2	2	3	3	2	3	2	2	2	2
CO5	2	2	2	2	2	2	2	2	3	3	3	2	2
Average	2.4	2.4	2.2	2.2	2.2	1.6	1.6	2.2	2.4	2.0	1.6	2.2	2.2

** For Entire Course, PO /PSO Mapping; 1 (Low); 2(Medium); 3(High) Contribution to PO/PSO*

PO1	Engineering Knowledge	PO7	Ethics	PSO1	Domain Skills 1
PO2	Problem Analysis	PO8	Individual and Collaborative Teamwork		
PO3	Design/Development of Solutions	PO9	Communication		
PO4	Conduct Investigations of Complex Problems	PO10	Project Management and Finance	PSO2	Domain Skills 2
PO5	Engineering Tool Usage	PO11	Life-Long Learning		
PO6	The Engineer and The World				

JUSTIFICATION FOR MAPPING

SR. NO	PO/PSO MAPPED	JUSTIFICATION
CO1	PO1	Engineering knowledge used in designing ESP32-based system
	PO2	Problem analysis of irrigation issues like overwatering
	PO3	Basic design of smart irrigation solution
	PO4	Initial investigation of soil moisture conditions
	PO5	Use of tools like Arduino IDE and sensors
	PO6	Understanding impact on agriculture field
	PO7	Ethical use of water resources
	PO8	Team collaboration in project planning
	PO9	Communication through report & presentation
	PO10	Basic project planning
	PO11	Learning new IoT technologies
	PSO1	Application of programming & embedded systems
	PSO2	Development of real-world automation system
	PO1	Application of engineering fundamentals
	PO2	Deep analysis of system requirements
	PO3	Designing system architecture
	PO4	Investigation using sensor testing
	PO5	Use of ESP32, L298N, MOSFET

CO2	PO6	Impact on smart agriculture
	PO7	Responsible water usage
	PO8	Strong teamwork in implementation
	PO9	Technical communication
	PO10	Managing project execution
	PO11	Learning system integration
	PSO1	Software-hardware integration
	PSO2	Development of IoT-based solution
CO3	PO1	Strong application of engineering knowledge
	PO2	Problem-solving in irrigation automation
	PO3	Full system design (rover + irrigation)
	PO4	Testing and validation of system
	PO5	Advanced tools usage (ESP32, drivers)
	PO6	Agricultural benefits
	PO7	Sustainable solution
	PO8	Team coordination
	PO9	Presentation skills
	PO10	Implementation planning
	PO11	Continuous learning
	PSO1	Embedded programming skills
	PSO2	Real-time system development
CO4	PO1	Use of technical knowledge
	PO2	Evaluation of system performance
	PO3	Improvement of design
	PO4	Analysis of results
	PO5	Tools usage for testing
	PO6	Social/environmental impact
	PO7	Ethical irrigation practices
	PO8	Team collaboration
	PO9	Effective communication
	PO10	Project management
	PO11	Adaptability to new tech

CO5	PSO1	Application of engineering practices
	PSO2	System optimization
	PO1	Knowledge application
	PO2	Analytical thinking
	PO3	System improvement
	PO4	Evaluation techniques
	PO5	Use of modern tools
	PO6	Impact on environment
	PO7	Ethical responsibility
	PO8	Teamwork
	PO9	Strong communication skills
	PO10	Project management skills
	PO11	Lifelong learning
	PSO1	Software & system skills
	PSO2	Practical implementation