SMART GARDEN MONITORING SYSTEM

INTERNET OF THINGS PROJECT REPORT

Submitted
In partial fulfillment of the requirements
for the award of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS & COMMUNICATION ENGINEERING

BY

AVULURI SATHVIKA -21761A0404

THIRUPATHI KAVYA-21761A0457

PINDIPROLU VENKATA VISHNU VARDHAN-21761A0440

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING
(AUTONOMOUS)
L. B. Reddy Nagar, Mylavaram – 521 230.

Affiliated to JNTUK, Kakinada & Approved by AICTE, New Delhi

Accredited by NBA and NAAC

Certified by ISO 9001-2015

2024-2025

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(AUTONOMOUS)

L. B. Reddy Nagar, Mylavaram – 521 230.

Affiliated to JNTUK, Kakinada & Approved by AICTE, New Delhi

Accredited by NBA and NAAC, Certified by ISO 9001-2015

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that **INTERNET OF THINGS** report entitled "**SMART GARDEN MONITORING SYSTEM**" is duly presented and submitted by **AVULURI.SATHVIKA (21761A0404) , THIRUPATHI.KAVYA-(21761A0457),PINDIPROLU.VISHNU VARDHAN-(21761A0440)** in partial fulfillment of requirement for the award of Bachelor of Technology in Electronics and Communication Engineering in Lakireddy Bali Reddy College of Engineering (A), Mylavaram, during the academic year 2023-2024.

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

The Satisfaction that accompanies that the successful completion of any task would be incomplete without the mention of people whose ceaseless co-operation made it possible, whose constant guidance and encouragement crown all efforts with success.

I express my deep sense of gratitude with reverence and humility to Government authorities for extending his support and for providing us with an environment to complete the project successfully.

I would like to express my sincerest gratitude to community people, for his support and for providing us with an environment to complete the project successfully.

I humbly express my thanks to our Principal **Dr**. **K. Appa Rao** for extending his support and for providing us with an environment to complete our project successfully.

I would also like to thank our Vice Principal, **Dr. K. Harinadha Reddy** for encouraging us which certainly helped to complete the project in time.

I would like to express our deep gratitude to Dean R & D, **Dr. S. Pichi Reddy** sir for his valuable suggestions.

I deeply indebted to our Head of Department **Dr. Y. Amar Babu,** who modeled us both technically and morally for achieving greater success in life.

I extremely thankful to my guides **Dr. P. Venkata Rao, Mr. K. Sasi Bhushan, Mr. N. Dharmachari, Ms. G. Asha**, Department of Electronics and Communication Engineering, for his excellence guidance, timely and valuable suggestions and encouragement that enabled the success of the project.

I would like to thanks Internet of Things lab technician **Mr. Y. Srinivasa Reddy**, Department of Electronics and Communication, for his support throughout the period of our project work for making it a successful one**.**

I would like to express my heart full thanks to our parents for their unflinching support and constant encouragement throughout the period of our project work for making it a successful one.

I would like to thank all the teaching and non-teaching staff members of Electronics and communication Engineering, who have extended their full co-operation during the course of my project.

I thank all my friends who helped me sharing knowledge and by providing material to complete the project in time.

AVULURI SATHVIKA (21761A0404)
THIRUPATHI.KAVYA-(21761A0457)
PINDIPROLU.VENKATA VISHNU VARDHAN(21761A0440)

CONTENTS

LIST OF FIGURES

ABSTRACT

A state-of-the-art device that makes it easier to monitor garden conditions in real time is the Bluetooth Enabled Garden Monitor. This system combines DHT11, water level, and soil moisture sensors to provide accurate environmental assessments that guarantee the best possible plant care. Through Bluetooth, users may access data remotely, allowing for water saving and prompt irrigation decisions. A buzzer warns users of dangerous water levels, and the mechanism controls irrigation by activating a relay. By encouraging sustainable gardening techniques, this project gives users the tools they need to keep plants healthy and use less water. Combining technology and environmental conscience, the user-friendly interface makes it appropriate for both inexperienced and seasoned gardeners.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Growing interest in sustainable agriculture and gardening in recent years has increased the demand for creative solutions that make effective plant care easier. Effectively controlling soil moisture and environmental factors is a major issue for gardeners of all skill levels. As technology has advanced, smart gardening solutions have surfaced that allow users to remotely monitor and maintain ideal growing conditions.Using Arduino Mega technology, the Bluetooth-enabled garden monitor outlined in this document is a multipurpose tool that integrates a number of crucial sensors to improve gardening productivity. To offer real-time information on the garden's circumstances, it uses a DHT11 temperature and humidity sensor, a soil moisture sensor, and a water level sensor. This monitor's Bluetooth connectivity enables customers to obtain important data from their cellphones making certain that they are able to decide on the requirements of their plants with knowledge.Plant health depends on soil moisture; excessive watering or underwatering can cause serious stress to plants, which can impair development and output. Continuously monitoring moisture levels, the soil moisture sensor sends out notifications when conditions depart from ideal ranges. Furthermore, by giving information about the amount of water in storage, the water level sensor helps to avoid unforeseen shortages during crucial watering times. Additionally important factors in gardening are temperature and humidity, which affect plant development rates and disease susceptibility. By monitoring these environmental variables, the DHT11 sensor helps users modify their care regimens in response to shifting circumstances.The garden monitor has a relay system to automate tasks and an audible buzzer for notifications to ensure user safety and convenience. irrigation according to measurements of soil moisture. A comprehensive monitoring solution that not only encourages healthy plant growth but also minimizes labor and water waste is produced by the combination of multiple sensors and technologies. In conclusion, this Bluetooth-enabled garden monitor is made to give gardeners the resources they require to successfully raise their plants, guaranteeing a flourishing garden while embracing the advantages of contemporary technology.

1.2 OBJECTIVES:

1. **Enhance Plant Care:** The primary goal of the Bluetooth-enabled garden monitor is to provide real-time monitoring of critical environmental factors such as soil moisture, temperature, and humidity. By offering accurate data, gardeners can make informed decisions to optimize their plant care, ensuring healthy growth and reducing the risk of stress or disease.
2. **Automate Watering:** The system aims to automate the watering process based on soil moisture readings. By utilizing a relay mechanism, the monitor can activate a watering system only when the soil is dry, preventing overwatering and underwatering. This automation not only saves time for gardeners but also helps maintain optimal soil conditions for plant health.
3. **Facilitate Remote Access**: The Bluetooth connectivity feature allows users to monitor their garden conditions remotely via a smartphone app. This objective emphasizes convenience for busy gardeners who may not always be on-site. They can receive real-time alerts and updates, enabling timely interventions when necessary.
4. **Educate Users:** The garden monitor serves an educational purpose by increasing awareness of the importance of soil moisture and environmental conditions. By providing clear feedback and alerts, it helps users understand how these factors impact plant health, fostering better gardening practices and enhancing overall knowledge of plant care.
5. **Promote Sustainability:** A key objective is to promote sustainable gardening practices by reducing water wastage. The monitor’s efficient detection of soil moisture and automated watering capabilities contribute to responsible water usage. By ensuring that water is applied only when needed, the system supports environmentally friendly gardening while also encouraging users to be more mindful of their resource consumption.

CHAPTER 2

TECHNICIAL SPECIFICATIONS:

2.1 HARDWARE TOOLS

2.1.1 Arduino MEGA 2560:

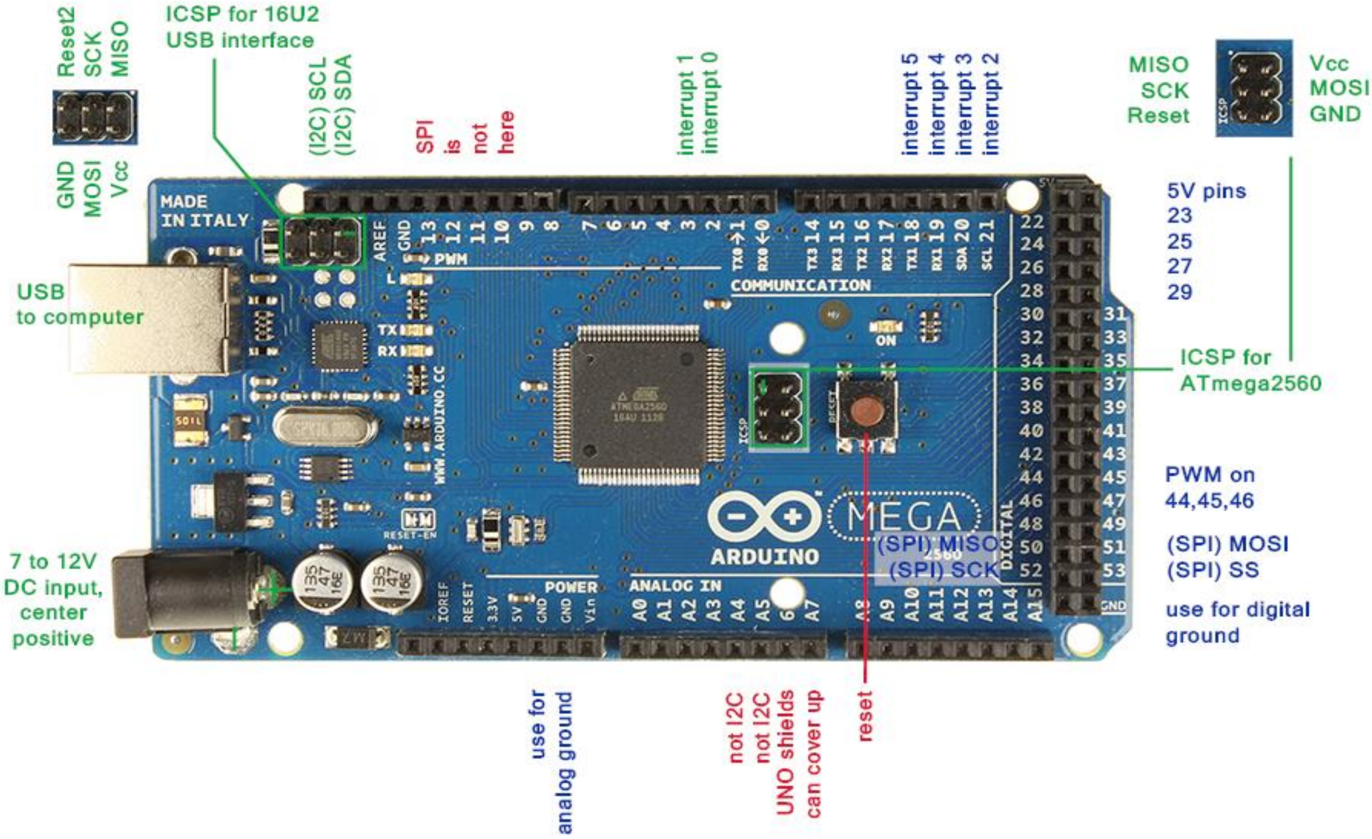

FIGURE 2.1. Arduino MEGA 2560

The **Arduino Mega 2560** is an open-source microcontroller board based on the ATmega2560 microcontroller, developed by Arduino.cc and first released in 2010. It is designed for projects that require a large number of input/output (I/O) pins, making it ideal for more complex applications compared to its smaller counterparts. The board is equipped with 54 digital I/O pins (15 of which can be used for PWM output), 16 analog input pins, and 4 UARTs (hardware serial ports), allowing for extensive interfacing capabilities with various sensors, actuators, and other devices.

Powering the Mega 2560 can be accomplished through a USB connection or an external power supply, with an acceptable voltage range of 7 to 12 volts. The board is programmable using the Arduino IDE, which simplifies the coding process and allows for easy uploads of new programs via the USB connection. The Mega 2560 comes with a pre-installed bootloader that facilitates uploading code without the need for an external programmer, streamlining the development process.

Unlike the earlier Arduino boards that relied on the FTDI USB-to-serial driver chip, the Mega 2560 utilizes an ATmega16U2 chip, which is programmed to act as a USB-to-serial converter, providing improved communication speed and flexibility. The Mega 2560 is particularly favored by hobbyists and educators for its robust capabilities, making it an excellent choice for beginners looking to tackle more ambitious projects.

The "Mega" in its name reflects its increased capacity for I/O operations, and the board's architecture has become a reference design for the Arduino community, contributing to a rich ecosystem of compatible shields and libraries. With a wide array of applications, from robotics to IoT projects, the Arduino Mega 2560 remains a pivotal tool in the maker and engineering communities.

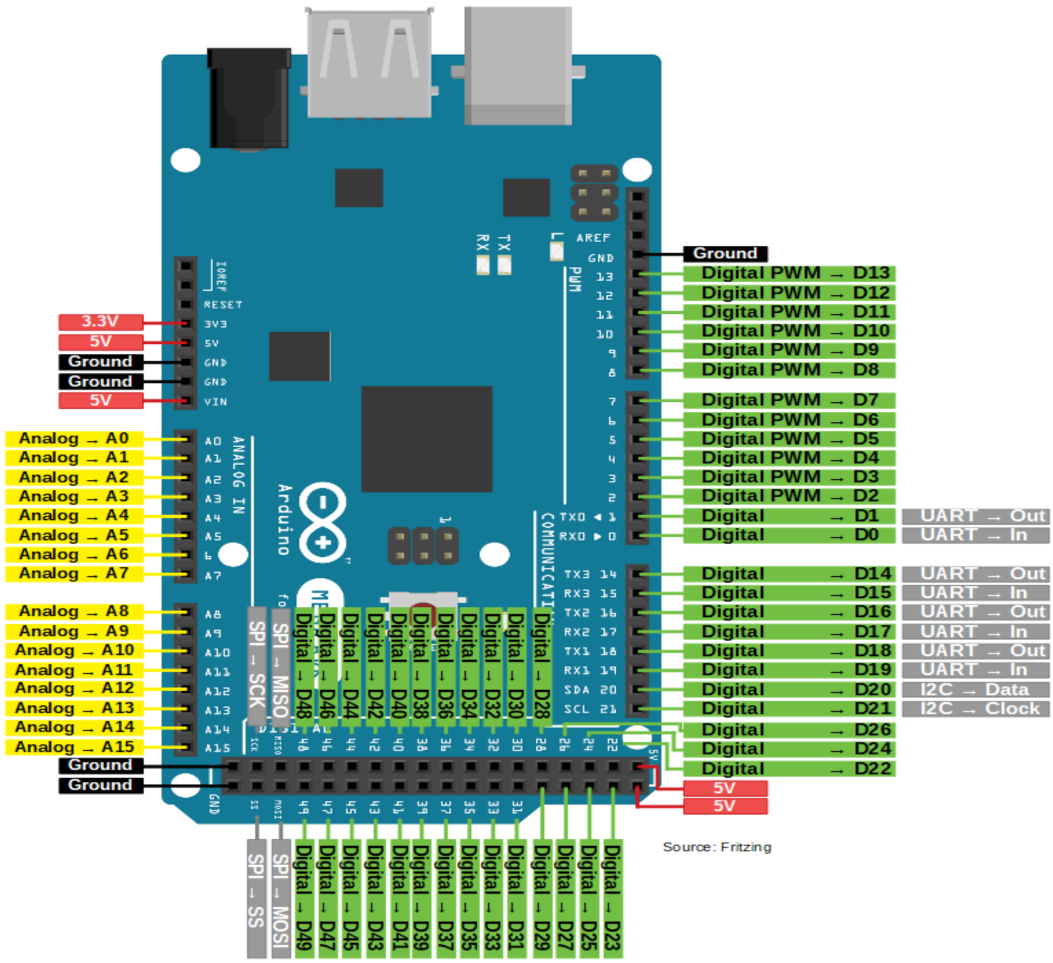

FIGURE 2.2. Arduino MEGA 2560 pin diagram

Arduino Mega 2560 Pin Overview:

1. **Digital I/O Pins (54 total):**
    - The Mega 2560 has 54 digital I/O pins, labeled from 0 to 53. These pins can be configured as either input or output.
    - PWM (Pulse Width Modulation): 15 of these pins (marked with a tilde, e.g., ~3, ~5, ~6, ~9, ~10, ~11, ~13, ~44, ~45, ~46) can generate PWM signals, allowing for analog-like control over devices such as motors and LEDs.
2. **Analog Input Pins (16 total):**
    - The board features 16 analog input pins, labeled A0 to A15. These pins are capable of reading analog signals, such as those from sensors, and converting them to a digital value (0-1023) through the built-in 10-bit ADC (Analog-to-Digital Converter).
    - These pins can also be used as digital pins if needed.
3. **Power Pins:**
    - Vin: This pin accepts an external power supply, with a recommended voltage range of 7 to 12 volts. It powers the board and can also be used to power connected peripherals.
    - 5V: This pin outputs a regulated 5V supply derived from the onboard voltage regulator, allowing you to power external components that require 5V.
    - 3.3V: This pin provides a regulated 3.3V supply, useful for powering low-voltage components that operate at this voltage level.
    - GND: There are multiple GND (ground) pins available, providing a common ground reference for connected components, ensuring stable operation of the entire system.
4. **Serial Communication Ports:**
    - The Mega 2560 includes 4 hardware serial ports (UARTs): Serial (pins 0 and 1), Serial1 (pins 18 and 19), Serial2 (pins 16 and 17), and Serial3 (pins 14 and 15). These allow for communication with multiple devices simultaneously, making it suitable for complex projects.
5. **ICSP Header:**
    - The ICSP (In-Circuit Serial Programming) header allows for direct programming of the microcontroller and provides an alternative method for uploading code, useful for debugging and advanced applications.
6. **Reset Pin**:
    - The reset pin can be used to reset the microcontroller. By connecting it to GND, the board will reset, which can be useful for debugging or restarting the program without cycling.

2.1.2 SOIL MOISTURE SENSOR:

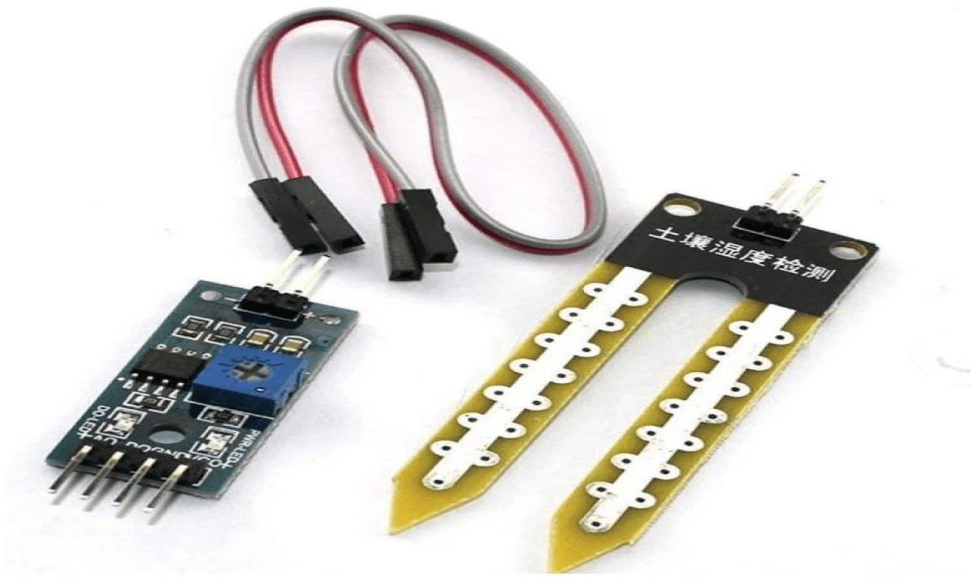

FIGURE 2.3. SOIL MOISURE SENSOR

A soil moisture sensor is an essential tool for monitoring the moisture level in the soil, which is critical for plant health. These sensors help automate irrigation systems, ensuring plants receive the appropriate amount of water.

Types of Soil Moisture Sensors:

1. **Resistive Soil Moisture Sensors:**
   - These sensors measure soil moisture by assessing the electrical resistance between two probes inserted into the soil. The resistance decreases as moisture increases since water conducts electricity better than dry soil.
2. **Capacitive Soil Moisture Sensors:**
   - These sensors use capacitive sensing to determine moisture levels. They measure the dielectric constant of the soil, which changes with moisture content. Capacitive sensors are generally more durable and less susceptible to corrosion compared to resistive sensors.

Pin Configuration and Working of a Typical Soil Moisture Sensors:

Here’s a detailed breakdown of a common capacitive soil moisture sensor's pins and their functions:

1. **VCC (Power Supply Pin):**
    - **Description:** This pin provides power to the sensor, typically operating at 3.3V to 5V.
    - **Function:** Connect this pin to the positive voltage supply (VCC) of the Arduino or another microcontroller.
2. **GND (Ground Pin):**
    - **Description:** This pin serves as the ground reference for the sensor.
    - **Function:** Connect this pin to the ground (GND) of the microcontroller to complete the electrical circuit.
3. **A0 (Analog Output Pin):**
    - **Description:** This pin outputs an analog voltage that corresponds to the moisture level in the soil.
    - **Function:** Connect this pin to an analog input on the microcontroller (e.g., A0 on an Arduino). The output voltage varies based on the soil moisture content, which can be read and processed to determine if watering is necessary.

4. **D0 (Digital Output Pin) (optional):**

- **Description:** Some soil moisture sensors have a digital output pin that provides a binary signal (HIGH/LOW) based on a set moisture threshold.
- **Function:** This pin can be connected to a digital input on the microcontroller. It outputs a HIGH signal when the soil is dry and a LOW signal when the soil is sufficiently moist. This feature is useful for simple threshold-based irrigation systems.

2.1.3 BUZZER

FIGURE 2.4. buzzer

Also known as a sounder, audio alarm or audio indicator, a buzzer is a basic audio devicethat **generates a sound from an incoming electrical signal**. Buzzers come in two primary forms — piezo buzzers and magnetic buzzers.

2.1.4 LEDs

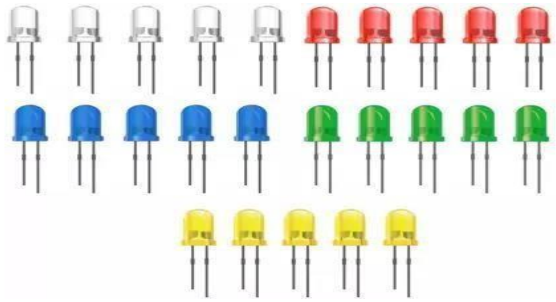

FIGURE 2.5. LEDs

A light-emitting diode (LED) is **a semiconductor device that emits light when an electric current flows through it**. When current passes through an LED, the electrons recombine with holes emitting light in the process.

2.1.5 WATER LEVEL SENSOR:

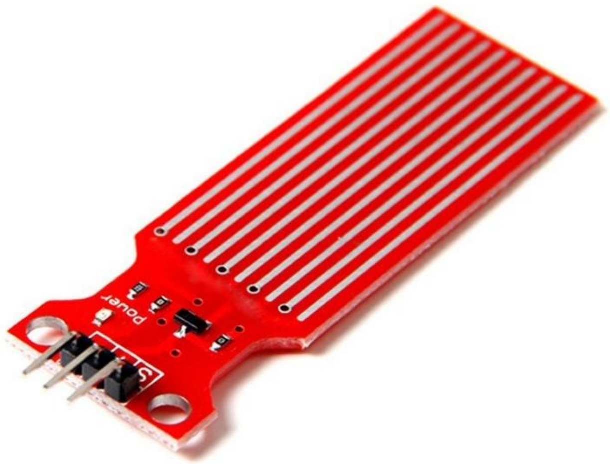

FIGURE 2.6 . Water level sensor

A water level sensor is a tool used to measure and identify the water level in reservoirs, tanks, and other containers. It is essential for maintaining appropriate water levels and avoiding problems like overflow or dry pump operation in a variety of applications, including industrial processes, water management, and autonomous irrigation systems.

The sensor uses mechanical floats or electrical signals to detect changes in the water level. A microprocessor receives a signal from the sensor when the water level hits a predetermined threshold. The microcontroller interprets the data and can initiate actions like turning on or off a water pump. This automation guarantees the health of plants or systems that depend on steady water levels while also improving water usage efficiency.

2.1.6 Breadboard

2.1.7 Bus Strips

A bus strip lets you connect the breadboard to a power supply so that the other electronic components on the breadboard can be powered. To give your breadboard power, you'll use the bus strips to connect to a power supply.

Bus strips are usually found at the outer edges of a breadboard or in between the terminal strips, and are almost always narrower than the terminal strips.

A typical breadboard will have two bus strips: a column for ground, which is marked in blue or

black coloring, and a column for power, also called voltage, which is marked in red.
Bus strips are also sometimes called rails, power rails, power buses, or just buses.

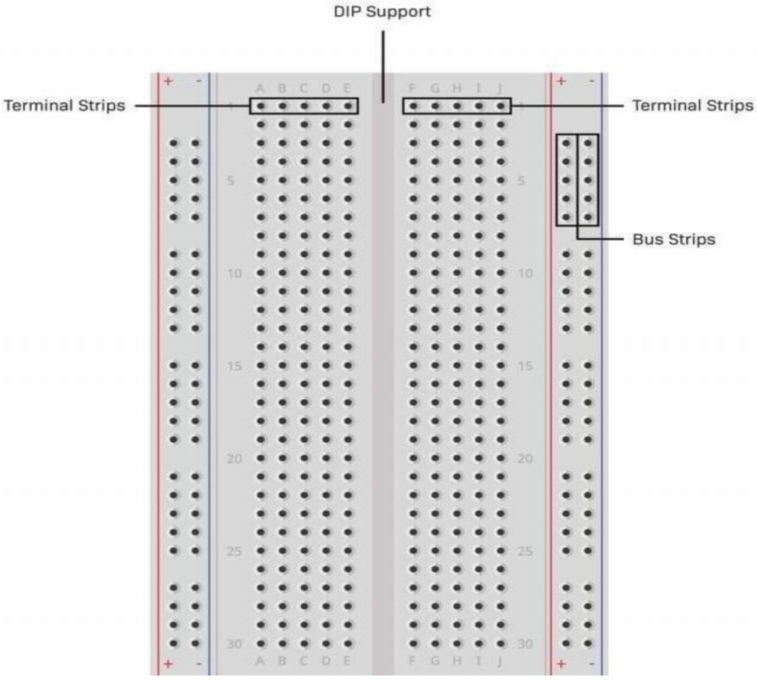

FIGURE 2.7. breadboard

Terminal Strips

Most of the area in a breadboard is taken up by terminal strips. Terminal strips are made up of small holes, or perforations, where you can plug in your electronic components.

Terminal strips are connected in specific ways based on their rows and columns. It's important to understand the layout of the terminal strip on the breadboard you're working with. Make sure you check the labeling of your breadboard before plugging things in!

Centergroove (DIP Support)

Most breadboards have a notch or a groove that runs down the center, through the middle of the terminal strips. This line down the middle serves a number of functions.

The center groove on a breadboard allows certain types of integrated circuits called dual in- line packages to be connected in a way that straddles that line. It also shows where the terminalstrips have been divided and which columns are connected, and it also allows breadboards to be easily stacked on top of each other for storage or larger projects.

A breadboard (sometimes called a plug block) is used for **building temporary circuits**. It is useful to designers because it allows components to be removed and replaced easily. It is useful to the person who wants to build a circuit to demonstrate its action, then to reuse the components in another circuit.

The holes in a breadboard are connected **by metal clips that span five holes, horizontally**. These metal clips allow each row of five holes to be connected. There are no vertical connections on a terminal strip. Horizontal rows on either side of the center groove are also not connected to each other.

2.1.8 Jumper wires

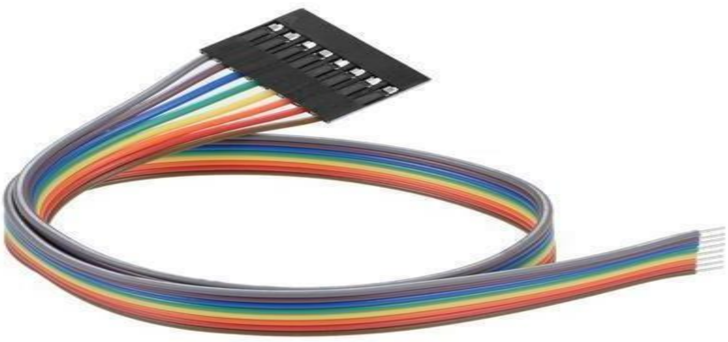

FIGURE 2.8 . Jumper wires

Jumper wires are not technically part of the breadboard itself, but they're an important part of any circuit or other electronics project that uses a breadboard.

A jumper wire is a short piece of wire with hard metal points on the end which plug into the holes on a breadboard. These allow you to make connections on the breadboard and start building a circuit. Jumper wires come in different colors and lengths, and you can buy them pre-made or make them yourself.

2.2 SOFTWARE TOOLS

2.2.1 TINKERCAD

Tinkercad is an online platform for 3D design, electronics, and coding. It is primarily aimed at beginners, students, and hobbyists who want to learn and experiment with these technologies without the need for advanced technical skills. Here are some key aspects of Tinkercad:

1. 3D Design: Tinkercad provides a user-friendly interface for creating 3D models. Users can start with basic shapes and manipulate them to create more complex objects. It's widely used for 3D printing projects and allows users to export their designs for use with 3D printers.

2. Electronics: Tinkercad also offers a circuit design and simulation tool. Users can build virtual electronic circuits by connecting components like resistors, LEDs, and microcontrollers.

The platform supports Arduino simulation, allowing users to program and test their circuits before physically building them.

3. Code Blocks: Tinkercad provides a visual coding environment that is based on block-based programming. Users can drag and drop code blocks to create programs for their 3D designs or electronic circuits. This is a great way for beginners to learn programming concepts.

4. Community and Sharing: Tinkercad has a community aspect where users can share their designs and projects with others. This can be a valuable resource for learning and collaborating with like-minded individuals.

5. Educational Use: Tinkercad is commonly used in educational settings to teach 3D design, electronics, and programming. Teachers can create accounts for their students, assign projects, and track progress.

6. Free and Cloud-Based: Tinkercad is a web-based platform, so there's no need to install software. It offers a free version with basic features, making it accessible to a wide range of users. Autodesk, the company behind Tinkercad, also offers premium features for those who require more advanced capabilities.

Tinkercad can be a valuable tool for anyone interested in learning about 3D design, electronics, and programming. Its user-friendly interface and educational resources make it a popular choice for beginners and educators in these fields.

2.2.2 ARDUINO IDE

The Arduino Integrated Development Environment (IDE) is an open-source software application that is used to write and upload code to Arduino microcontroller boards. Arduino boards are popular among hobbyists, makers, and engineers for creating a wide range of electronic projects, from simple LED blinkers to complex robots and IoT devices.

Here are some key aspects of the Arduino IDE:

1. Code Editor: The Arduino IDE provides a simple yet effective code editor with features like syntax highlighting, autocompletion, and error checking. It uses the C/C++ programming language, which is the standard for Arduino development.

2. Board Manager: Arduino supports a variety of microcontroller boards, each with its own specifications and capabilities. The Arduino IDE includes a Board Manager that allows you to select the specific Arduino board you are using, which ensures that the code is compiled correctly for that board.

3. Libraries: Arduino IDE includes a collection of libraries that provide pre-written code for various functions, such as interfacing with sensors, displays, and communication protocols (e.g., I2C, SPI, UART). Users can also create their own libraries or download additional libraries from the Arduino Library Manager.

4. Serial Monitor: Arduino boards often communicate with a computer through a serial connection. The Serial Monitor in the Arduino IDE allows you to send and receive data between your computer and the Arduino board, which is useful for debugging and monitoring your projects.

5. Uploading Code: Once you've written your code, you can upload it to the Arduino board via a USB connection. The IDE handles the compilation and uploading process seamlessly.

6. Examples: The Arduino IDE provides a wide range of example sketches that demonstrate various functionalities of Arduino boards and components. These examples serve as a great starting point for beginners and help users learn how to use different sensors and modules.

7. Community and Ecosystem: Arduino has a large and active community of users and developers. You can find a wealth of tutorials, forums, and resources online to help you with your projects and troubleshoot any issues you encounter.

8. Cross-Platform: Arduino IDE is available for multiple operating systems, including Windows, macOS, and Linux, making it accessible to a broad audience.

In addition to the standard Arduino IDE, there are alternative development environments and platforms that offer enhanced features and capabilities, such as PlatformIO, which supports a wider range of microcontrollers and has a more advanced build system.

Overall, the Arduino IDE is a beginner-friendly tool that makes it easy to get started with microcontroller programming and electronics projects. It has played a significant role in promoting the maker and DIY culture.

2.23 CODE:

```
#include <DHT.h>
#define moisturePin A0      // Soil moisture sensor connected to analog pin A0
#define relayPin 7          // Relay connected to digital pin 7
#define waterLevelPin A1    // Water level sensor connected to analog pin A1
#define buzzerPin 6         // Buzzer connected to digital pin 6
#define DHTPIN 8            // DHT11 connected to digital pin 8
#define DHTTYPE DHT11       // DHT11 type

DHT dht(DHTPIN, DHTTYPE);    // Initialize DHT11
int moistureValue = 0;       // Variable to store moisture level
int waterLevelValue = 0;     // Variable to store water level
int moistureThreshold = 500; // Threshold for dry/wet soil
int validReadThreshold = 50; // Minimum valid reading to filter out invalid readings

void setup() {
  Serial.begin(9600);            // Initialize serial communication with the computer
  Serial1.begin(9600);           // Initialize Bluetooth serial communication
  pinMode(relayPin, OUTPUT);      // Set relay pin as output
  pinMode(buzzerPin, OUTPUT);     // Set buzzer pin as output
  digitalWrite(relayPin, HIGH);   // Turn off relay initially (active low)
  digitalWrite(buzzerPin, LOW);   // Turn off buzzer initially
  dht.begin();                    // Initialize DHT11 sensor
}

void loop() {
  // Read soil moisture value
  moistureValue = analogRead(moisturePin);
  Serial.print("Soil Moisture Level: ");
  Serial.println(moistureValue);
  Serial1.print("Soil Moisture Level: ");
  Serial1.println(moistureValue);

  // Check moisture sensor is valid
```

```
if (moistureValue > validReadThreshold && moistureValue <= 1023) {
  if (moistureValue < moistureThreshold) {
    Serial.println("Soil is wet. No watering needed.");
    Serial1.println("Soil is wet. No watering needed.");
    digitalWrite(relayPin, HIGH);  // Turn off relay
  } else {
    Serial.println("Soil is dry. Watering needed.");
    Serial1.println("Soil is dry. Watering needed.");
    digitalWrite(relayPin, LOW);   // Turn on relay
  }
} else {
  Serial.println("Invalid moisture reading.");
  Serial1.println("Invalid moisture reading.");
}

// Read water level sensor
waterLevelValue = analogRead(waterLevelPin);
Serial.print("Water Level: ");
Serial.println(waterLevelValue);
Serial1.print("Water Level: ");
Serial1.println(waterLevelValue);

if (waterLevelValue > 600) {  // Adjust threshold based on your sensor's testing
  Serial.println("Water tank is full, stop pouring.");
  Serial1.println("Water tank is full, stop pouring.");
  digitalWrite(relayPin, HIGH);  // Turn off relay
  digitalWrite(buzzerPin, LOW);  // Turn off buzzer
} else {
  Serial.println("Water level is low, activate watering alert.");
  Serial1.println("Water level is low, activate watering alert.");
  digitalWrite(relayPin, LOW);   // Turn on relay
  digitalWrite(buzzerPin, HIGH); // Turn on buzzer
}

// Read DHT11 sensor for humidity and temperature
float humidity = dht.readHumidity();
float temperature = dht.readTemperature();
```

```
  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.print(" °C, Humidity: ");
  Serial.print(humidity);
  Serial.println(" %");

  Serial1.print("Temperature: ");
  Serial1.print(temperature);
  Serial1.print(" °C, Humidity: ");
  Serial1.print(humidity);
  Serial1.println(" %");

  delay(2000);  // Wait 2 seconds before next loop
}
```

CHAPTER 3

METHODOLOGY AND DESIGN FRAMEWORK

3.1 METHODOLOGY:

The process for the Bluetooth-enabled garden monitor project starts with meticulous design and planning. To provide clarity on what the system should do, the first phase entails defining the project scope, objectives, and requirements. It is essential to choose the right parts, such as the Arduino Mega 2560, the DHT11 temperature and humidity sensor, the soil moisture sensor, the water level sensor, the relay module, the buzzer, and the Bluetooth module. Creating the schematic diagram and circuit layout also aids in visualizing the relationships and interactions between the various parts.

Purchasing the required parts and equipment comes next when the planning stage is finished. This include assembling the sensors, Arduino board, and additional peripherals as well as making sure there is a dependable power source. The circuit assembly happens after procurement. The circuit determines how components are connected.

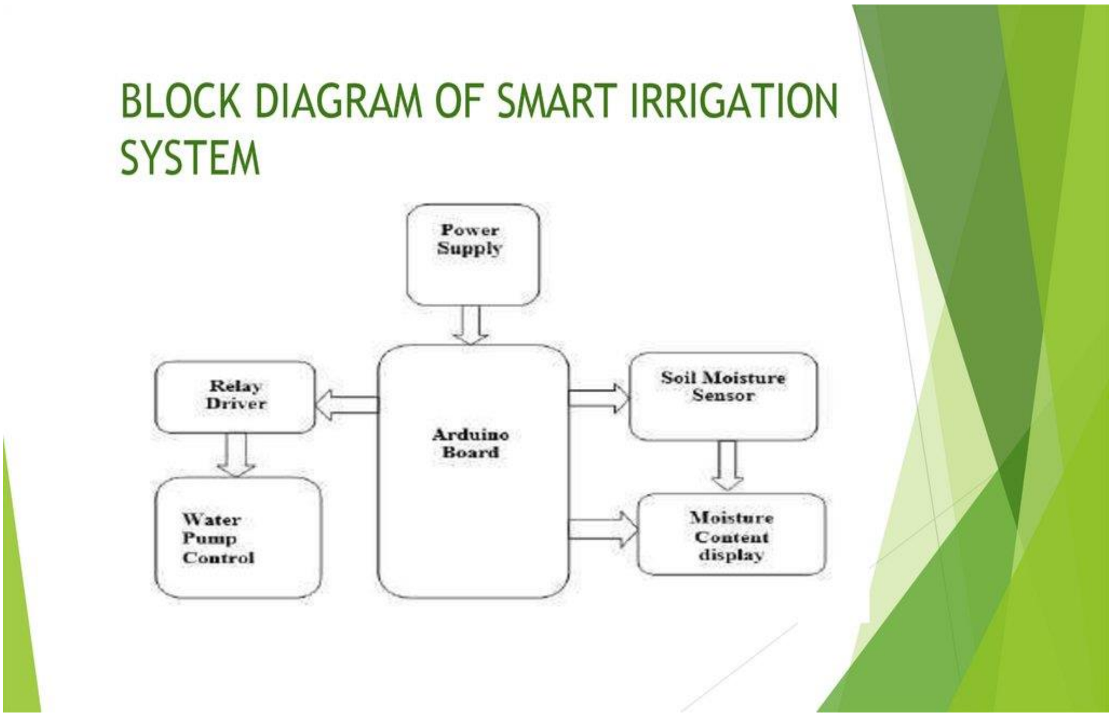

FIGURE 3.1. Block diagram of Bluetooth enabled garden monitor.

3.2 HARDWARE ASSEMBLY

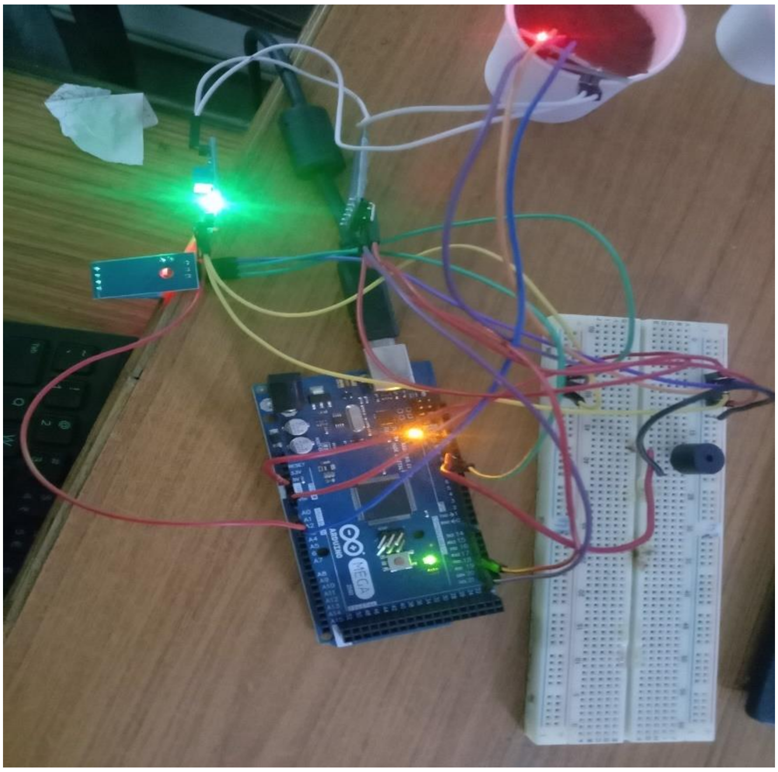

FIGURE 3.2 Hardware Circuit

Connections for the Bluetooth-Enabled Garden Monitor Circuit:

Based on the provided code, here’s a detailed description of the connections for each component in the circuit.

1. Arduino Mega 2560

- The central microcontroller that will read sensor data and control the relay.

2. Soil Moisture Sensor

- **VCC:** Connect to the 5V pin on the Arduino.
- **GND:** Connect to a GND pin on the Arduino.
- **A0:** Connect to Analog pin A0 on the Arduino.

3. Water Level Sensor

- **VCC:** Connect to the 5V pin on the Arduino.
- **GND:** Connect to a GND pin on the Arduino.
- **A1:** Connect to Analog pin A1 on the Arduino (or use Digital pin D0 if configured for digital output).

4. DHT11 Temperature and Humidity Sensor

- **VCC:** Connect to the 5V pin on the Arduino.
- **GND:** Connect to a GND pin on the Arduino.

- **Data Pin:** Connect to Digital pin 8 on the Arduino.

5. Relay Module (for Water Pump Control)

- **VCC:** Connect to the 5V pin on the Arduino.
- **GND:** Connect to a GND pin on the Arduino.
- **IN (Control Pin):** Connect to Digital pin 7 on the Arduino. This pin will control the relay to activate or deactivate the connected water pump.

6. Buzzer

- **Positive Lead:** Connect to Digital pin 6 on the Arduino.
- **Negative Lead:** Connect to a GND pin on the Arduino.

7. Bluetooth Module (e.g., HC-05)

- **VCC:** Connect to the 5V pin on the Arduino.
- **GND:** Connect to a GND pin on the Arduino.
- **TX (Transmit):** Connect to Digital pin 10 on the Arduino (or any other available pin for software serial).
- **RX (Receive):** Connect to Digital pin 11 on the Arduino (with a voltage divider if necessary, as the Arduino operates at 5V and HC-05 at 3.3V).

3.3 CONTRIBUTION TO THE PROJECT:

1. **Testing and Validation:**
   - Assist in testing the Bluetooth-enabled garden monitor under various environmental conditions to ensure accurate readings from soil moisture, temperature, humidity, and water level sensors. Contribute to the development of test cases and procedures that evaluate the system's response to different watering needs, sensor performance, and Bluetooth connectivity, ensuring reliable operation in real-world scenarios.
2. **Documentation and Reporting:**
   - Help create comprehensive documentation, including user manuals that guide users on setup, operation, and troubleshooting of the garden monitor. Develop technical specifications for each component used in the project and compile project reports that detail progress, findings, and challenges encountered during development, ensuring effective communication within the team.
3. **Project Management and Coordination:**
   - Take on project management responsibilities such as scheduling tasks, allocating resources, and coordinating activities among team members. Monitor progress towards project milestones and objectives, ensuring timely completion of tasks and effective collaboration to meet project goals.
4. **Continuous Improvement:**
   - Identify opportunities for improvement throughout the project, including enhancing sensor calibration processes, optimizing the software for better performance, and refining the user interface for Bluetooth connectivity. Suggest actionable enhancements for future versions of the garden monitor, ensuring that the system remains user-friendly and adaptable to evolving gardening n

Contributions in the project:

A. SATHVIKA(21761A0404):

As a key member of the Smart Garden Monitor project, my contributions spanned multiple stages, from project selection to practical implementation and documentation. Here is a detailed overview of my involvement:

- **Project Selection:** After thorough analysis of several options, I selected this project for its potential to improve gardening efficiency.
- **Sensor Integration:** I designed and integrated soil moisture, temperature, and humidity sensors with the Arduino Mega.
- **Firmware Development**: I developed firmware to process sensor data and control automated watering.

T.KAVYA(21761A0457):

- **Hardware Assembly:** Assembled the system hardware, including the water level sensor and servo motor.
- **IoT Integration:** Programmed the microcontroller to send real-time notifications to a mobile app/web platform.
- **Testing and Debugging:** Conducted rigorous testing under different conditions to ensure system reliability and accuracy.
- **Documentation:** Assisted in preparing detailed reports covering the system's design, functionality, testing, and performance.

P.VENKATA VISHNU VARDHAN(21761A0440):

- **System Design:** Designed the overall system architecture, ensuring seamless integration of hardware and software components.
- **Automated Watering System:** Developed and implemented an automated watering system using servo motors.
- **Data Analytics:** studied the core working of various sensors and their individual operation (temperature, humidity, soil moisture).
- **Team Collaboration:** Worked collaboratively with teammates to integrate hardware and software components effectively.

TEAM ACHEIVEMENTS:

- Developed a Smart Garden Monitor system to optimize gardening practices
- Successfully integrated sensors for real-time monitoring.
- Implemented automated watering and IoT-based notifications.
- Ensured accurate detection and response through thorough testing and calibration.

CHAPTER 4
RESULTS AND DISCUSSION

The Bluetooth-enabled garden monitor system utilizes an Arduino Mega 2560 microcontroller, integrated with multiple sensors to monitor soil moisture, temperature, humidity, and water levels. The system is designed to enhance gardening efficiency by providing real-time data and alerts to users. When the soil moisture sensor detects low moisture levels, it sends a signal to the Arduino, which processes this input and activates a relay to control the watering system. Additionally, the DHT11 sensor measures temperature and humidity, providing valuable environmental data for plant health.

The system features a Bluetooth module (such as HC-05) that allows users to connect their smartphones or computers for remote monitoring. When moisture levels are insufficient, the relay is activated, turning on the water pump, while a buzzer sounds an alert to notify users of the need for watering. The LCD display provides real-time information on soil moisture, temperature, and humidity, enhancing user awareness.

This garden monitor incorporates a user-friendly interface that displays messages such as “Soil is dry. Watering needed” or “Water tank is full, stop pouring,” depending on sensor readings. The Bluetooth connectivity enables users to receive updates directly on their devices, facilitating proactive garden management. Overall, this system combines effective monitoring and automation to support healthy plant growth and efficient resource.

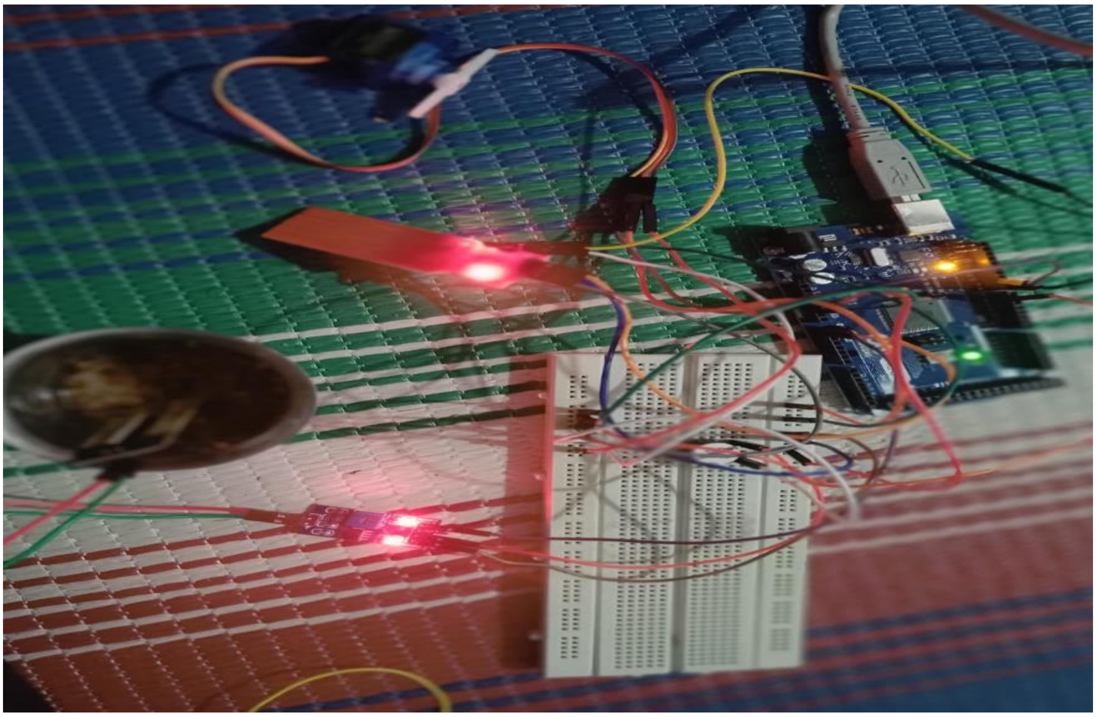

FIGURE 4.1. soil is dry

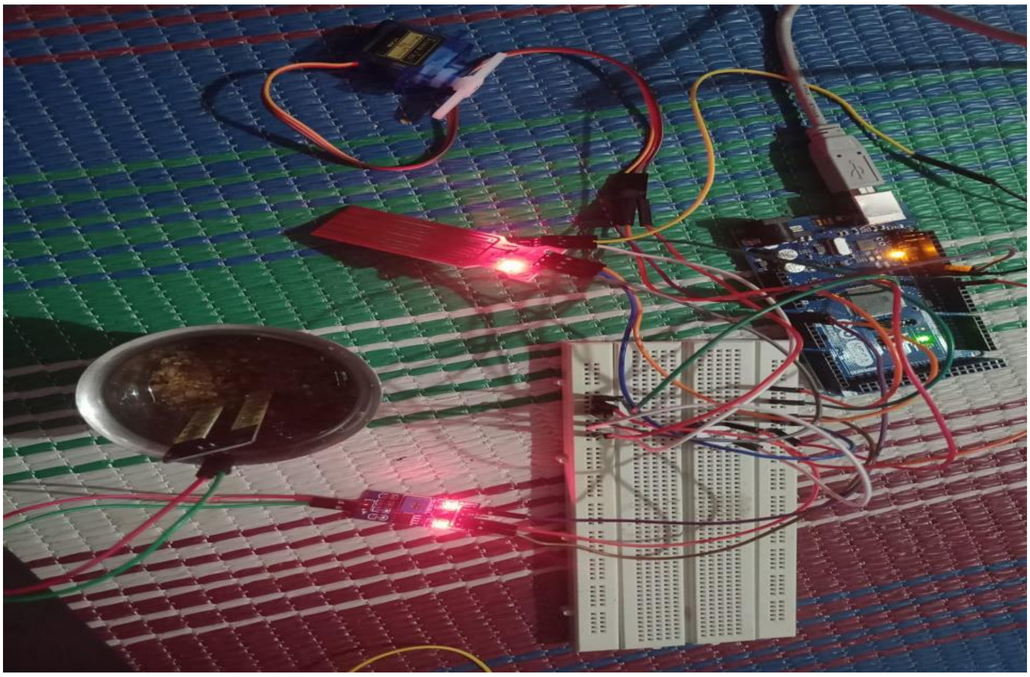

FIGURE 4.2. Soil is wet

The successful implementation of this Bluetooth-enabled garden monitor system has yielded truly remarkable results, unequivocally showcasing its vast potential to revolutionize and transform gardening practices for the better. By leveraging its automated watering system and real-time monitoring capabilities, gardeners can now optimize water usage with unparalleled precision, significantly reduce waste, and create an ideal environment conducive to vibrant plant growth. Furthermore, the system's ability to provide timely alerts and notifications enables gardeners to take prompt, informed action, effectively preventing potential issues and ensuring a healthy and thriving garden. The impact on plant health is evident, with noticeable improvements in growth rates, yield, and overall resilience. Moreover, the system's efficiency and effectiveness have also led to substantial water conservation, reducing the environmental footprint of gardening practices. Overall, this innovative system has proven to be an indispensable valuable tool for gardeners, offering a unique and compelling combination of efficiency, effectiveness, sustainability, and ease of use.

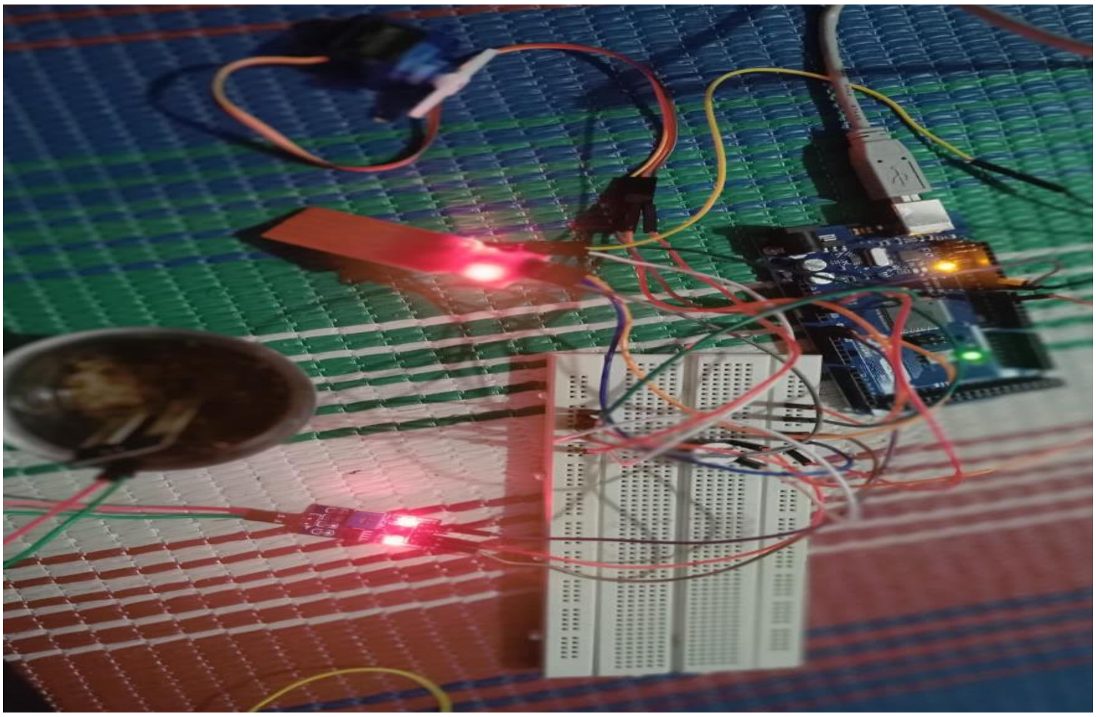

FIGURE 4.3 Soil is dry

Operation: My aurdino code helps to ring buzzer whenever soil is dry and led will glow as well as my servo motor will be on.It detects that the soil is dry using the soil moisture sensor,water level sensor which measures soil moisture level and water level in the soil.And,my project also prints the temperature and humidity present in the soil.

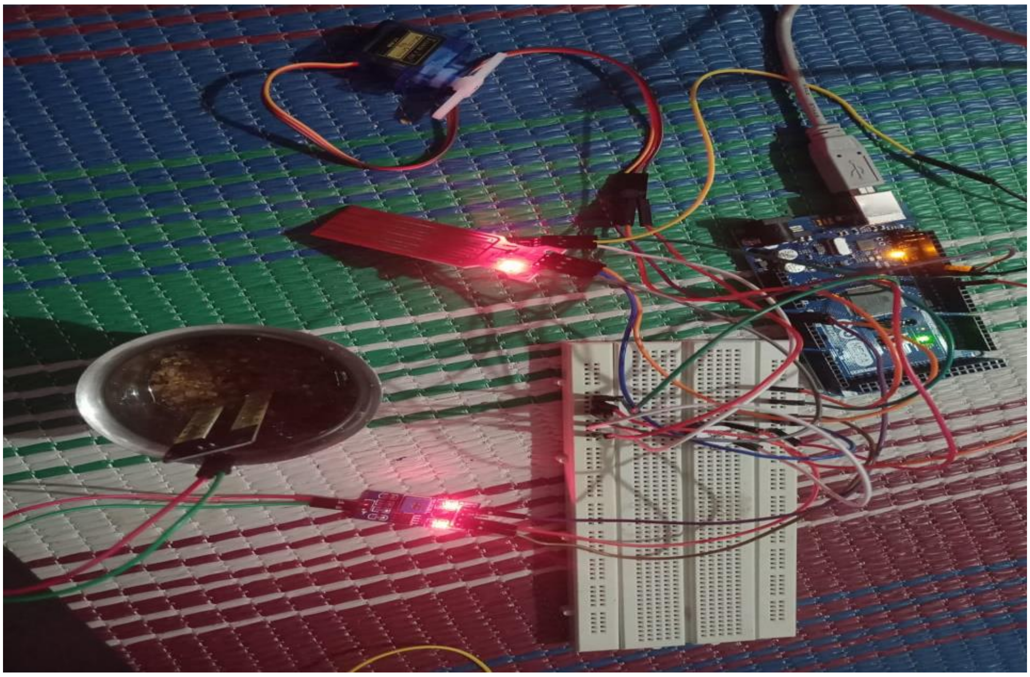

FIGURE 4.4 Soil is wet

CHAPTER 5

CONCLUSIONS AND FUTURE SCOPE

5.1 CONCLUSION

The Smart Garden Monitor system successfully integrates cutting-edge technology to optimize gardening practices. Utilizing Arduino Mega, soil moisture, water level, and DHT11 sensors, this system ensures efficient water management and automated watering via servo motor, minimizing waste and promoting healthy plant growth. Real-time monitoring and alerts enable proactive gardening, reducing potential issues. The system demonstrates significant potential for water conservation and environmental sustainability, with results showing improved plant health, yield, and growth rates. This project showcases the effectiveness of IoT applications in modern gardening, offering ease of use, efficiency, and reliability. Its scalability and adaptability make it suitable for various gardening applications, enhancing gardening precision through sensor integration and automation. Overall, the Smart Garden Monitor ensures optimal soil moisture and water levels, while DHT11 temperature and humidity monitoring further optimize plant care. This innovative system revolutionizes traditional gardening methods, demonstrating the potential for technology-driven gardening solutions that conserve resources and promote sustainable practices.

5.2 FUTURE SCOPE

The Smart Garden Monitor system has tremendous potential for future enhancements and expansions. Hardware upgrades can include integrating additional sensors for comprehensive soil analysis, weather forecasting modules for predictive watering, and advanced actuators for optimized water distribution. Software advancements can involve implementing machine learning algorithms for predictive maintenance and anomaly detection, developing mobile apps for remote monitoring and control, and integrating with popular smart home platforms. Furthermore, enhancing networking and connectivity through wireless communication protocols, cloud-based data storage, and analytics will enable scalable monitoring. Integrating artificial intelligence and automation will allow for automated decision-making, AI-powered plant disease detection, and recommendation systems. The system can also expand to other applications such as greenhouse monitoring, urban agriculture, vertical farming, and environmental monitoring. Additionally, research and development opportunities exist in exploring alternative energy sources, nanotechnology in soil sensing, and novel sensor technologies for plant health monitoring. By addressing these areas, the Smart Garden Monitor system will continue to evolve, providing innovative solutions for modern gardening and contributing to a more sustainable and efficient agricultural future.

REFERENCES

1. DFRobot. "DHT11 Temperature and Humidity Sensor Module" (2018)

Technical documentation for DHT11 sensor specifications and application.

2. SparkFun Electronics. "Soil Moisture Sensor Module" (2019)

Documentation for soil moisture sensor specifications, wiring, and code.

3. Adafruit Industries. "Water Level Sensor Module" (2020)

Tutorial and documentation for water level sensor usage and programming.

4. ServoCity. "Servo Motor Control for Automation" (2017)

Guide to servo motor selection, wiring, and programming.

5. A. Kumar et al. "Smart Gardening Systems Using IoT" (IoT Times, 2020)

Research on IoT-based smart gardening systems and architecture.

6. S. S. Iyengar et al. "IoT-Based Smart Gardening System" (ResearchGate, 2020)

Research on IoT-based smart gardening with soil moisture monitoring.

7. R. A. Khan et al. "Smart Gardening Using IoT and Machine Learning" (ScienceDirect, 2020)