MICROCONTROLLER BASED SMART IRRIGATION SYSTEM

A Project Report of Internship Training

Submitted by

Sk Jalal Uddin

in partial fulfilment for the award of the degree of

B.TECH

In the Department of

ELECTRONICS & TELECOMMUNICATION ENGINEERING

KALYANI GOVERNMENT ENGINEERING COLLEGE

at

Ardent Computech Pvt. Ltd.

August – September 2024

Ardent® Computech Pvt. Ltd.
Module – 132, Ground Floor, SDF Building,
Sector V, Salt Lake, Kolkata – 700091.
☎+91 33 40073507
✉ training@ardentcollaborations.com

BONAFIDE CERTIFICATE

This is to certify that student have successfully completed the project titled **MICROCONTROLLER BASED SMART IRRIGATION SYSTEM** under my supervision during the period from **02/08/2024** to **25/09/2024** which is in partial fulfilment of requirements for the award of the **B.TECH** degree and submitted to the Department of ECE of **KALYANI GOVERNMENT ENGINEERING COLLEGE.**

Name of Student	Signature
Sk Jalal Uddin	Sk. Jalal uddin

Debtanu Mondal

Signature of the Supervisor

Date**:**

Name of the Project Supervisor: **DEBTANU MONDAL**

Acknowledgement

The achievement that is associated with the successful completion of any task would be incomplete without mentioning the names of those people whose endless cooperation made it possible. Their constant guidance and encouragement made all our efforts successful.

We take this opportunity to express our deep gratitude towards our project mentor **DEBTANU MONDAL** for giving such valuable suggestions, guidance and encouragement during the development of this project work.

Last but not the least we are grateful to all the faculty members of Ardent Computech Pvt. Ltd. for their support.

MICROCONTROLLER BASED SMART IRRIGATION SYSTEM

Abstract

Water conservation and efficient agricultural practices are critical in modern agriculture, especially in areas facing water scarcity. This project proposes the development of a Smart Irrigation System using the Arduino Uno R3 to optimize water usage and automate the irrigation process. The system integrates soil moisture sensors, a water pump, and real-time monitoring to deliver water based on soil conditions.

The Arduino Uno R3 serves as the microcontroller, processing data from soil moisture sensors that measure the moisture content in the soil. When the moisture level drops below a predefined threshold, the Arduino triggers the water pump to irrigate the field until optimal moisture levels are restored. Additionally, the system includes features for manual control, remote monitoring, and real-time updates via a connected mobile or web interface. This ensures the farmer can monitor and control the irrigation process from anywhere, reducing water wastage while improving crop health and yield.

The Smart Irrigation System offers an affordable, automated solution for sustainable agriculture, improving water use efficiency and crop management, especially in regions with limited water resources.

Table of Contents

Introduction

Smart irrigation systems use advanced technology to optimize water usage in agricultural and landscaping applications. These systems employ sensors, weather data, and connectivity to automatically adjust watering schedules based on real-time conditions. Soil moisture sensors detect when plants actually need water, while weather forecasts help prevent unnecessary watering before rain. Some systems use satellite imagery or drone data to map plant health across large areas. Mobile apps allow users to monitor and control their irrigation remotely. By precisely delivering water only when and where it's needed, smart irrigation can significantly reduce water waste, lower costs, and improve plant health. This technology is becoming increasingly important as water scarcity and environmental concerns grow worldwide.

Introduction to Arduino UNO R3

The Arduino UNO is a popular microcontroller board based on the ATmega328P chip. It is widely used in electronics projects due to its simplicity, affordability, and flexibility. The board features 14 digital input/output pins, 6 analog inputs, a 16 MHz quartz crystal, and a USB connection for programming. It can be powered via USB or an external power supply. With an open-source platform, the Arduino UNO is supported by an extensive community, offering a wide range of libraries and tutorials, making it ideal for beginners and hobbyists to prototype interactive projects, sensors, and automation systems.

Key Components of Arduino UNO R3

The Arduino Uno is a widely used microcontroller board based on the ATmega328P microchip. It's part of the Arduino family, known for being beginner-friendly and open-source. Here’s a basic introduction to its components, features, and uses:

Key Features

Microcontroller: ATmega328P, an 8-bit AVR microcontroller.

Operating Voltage: 5V.

Input Voltage: 7-12V (via barrel jack or USB).

Digital I/O Pins: 14 (6 of which can be used for PWM output).

Analog Input Pins: 6, for reading sensor data.

Flash Memory: 32 KB for storing programs.

SRAM: 2 KB for storing temporary data.

EEPROM: 1 KB for storing permanent data.

Clock Speed: 16 MHz.

Communication Interfaces: UART (serial), SPI, and I²C.

Main Components

USB Port: Connects to a computer to upload code or power the board.

Power Jack: Used to power the board with an external adapter (7-12V).

Reset Button: Resets the board and restarts the program.

Digital and Analog Pins: Used to interface with external components like sensors, motors, LEDs, etc.

PWM (Pulse Width Modulation): 6 digital pins (marked with ~) can output PWM signals, useful for controlling devices like motors and LED brightness.

Common Uses

Prototyping Electronics: Arduino Uno is ideal for beginners and hobbyists building prototypes of electronic projects.

Sensors and Actuators: It can interface with sensors (e.g., temperature, distance) and actuators (e.g., motors, relays).

IoT Projects: With additional modules (Wi-Fi, Bluetooth), the Uno can be used in Internet of Things (IoT) applications.

Automation: It can automate tasks by controlling devices based on sensor input.

Programming

Arduino IDE: The Arduino Uno is programmed using the Arduino IDE (Integrated Development Environment), where you write code in a simplified version of C++.

Sketch: The program written in the IDE is called a "sketch." It is uploaded to the board via a USB connection.

Arduino Uno is known for its simplicity and ease of use, making it a great starting point for anyone interested in electronics and embedded systems development.

Applications: The Arduino UNO is a versatile microcontroller board used in various applications, such as:

1. Home Automation: Control lights, fans, and other household appliances using sensors or mobile devices via Wi-Fi or Bluetooth modules.

2.Robotics: Used to control motors, servos, and sensors in robots for tasks like obstacle avoidance, line following, and remote control.

3. Weather Monitoring Stations: Measure environmental parameters like temperature, humidity, and atmospheric pressure using sensors (e.g., DHT11 or BMP180).

4. IoT Projects: Collect data from sensors and transmit it to the cloud for monitoring and analysis using Wi-Fi modules (e.g., ESP8266).

5. Wearable Technology: Create smart wearable devices for health monitoring, fitness tracking, or gesture control.

6. Automated Irrigation Systems: Control water flow based on soil moisture levels using soil moisture sensors and relays.

7. Security Systems: Build alarm systems with motion detectors (PIR sensors), cameras, and door lock systems.

8. Interactive Art Installations: Use Arduino to create responsive artworks that react to sound, light, or movement.

Arduino UNO's simplicity and flexibility make it ideal for hobbyists, engineers, and educators alike.

Components used in the Project

Here is a list of the total components used in the microcontroller based smart street light project:

1.Arduino UNO R3 Board: Acts as the main controller for processing the input signals from the sensors and triggering outputs (like alarms).

2. Soil Moisture Sensor : Soil moisture sensors measure the volumetric water content in soil.

3. Relay Module : Relay modules are simply circuit boards that house one or more relays.

4. LEDs: Provide illumination for the streetlights.

5. Resistors and Jumper Wires: Used for connecting the components to the Arduino.

6. Breadboard: For prototyping the circuit. You can build your circuit on a breadboard before finalizing it.

These components together allow the Arduino to detect both ambient light and vehicles or pedestrians to trigger whole system to provide lighting for the street.

Components Descriptions

Arduino UNO R3 Board

Features

1. Microcontroller: ATmega328P (8-bit AVR microcontroller)

2. Operating Voltage: 5V

3. Input Voltage (recommended): 7-12

4. Input Voltage (limits): 6-20V

5. Digital I/O Pins: 14 pins, of which 6 can be used as PWM outputs

6. Analog Input Pins: 6 pins (A0 to A5)

7.DC Current per I/O Pin*:

8. Flash Memory: 32 KB (of which 0.5 KB is used by the bootloader

9. SRAM: 2 KB

10. EEPROM: 1 KB

11. Clock Speed:

12. USB Connection: Used for programming and power (Type B USB port)

13. Power Jack: For external power input (7-12V)

14.ICSP Header: For in-circuit programming of the microcontroller

15. Reset Button: Resets the microcontroller

16. LED Indicator: On-board LED connected to pin 13

17. USB to Serial Converter: ATmega16U2 for easier programming via USB

18.Dimensions: 68.6 mm × 53.4 mm

19. Compatible with Shields: The R3 version has improved pin alignment and compatibility with more shields and modules.

Applications

1. Home Automation: Control lights, fans, and appliances.

2.Robotics: Drive motors, sensors, and actuators for autonomous robots.

3. IoT Projects: Collect and transmit sensor data via Wi-Fi/Bluetooth.

4. Environmental Monitoring: Measure temperature, humidity, and air quality.

5. Security Systems: Motion detection, alarms, and surveillance.

6. Wearable Tech: Health monitors and gesture-controlled devices.

7. 3D Printing: Manage motor movements and temperature control.

8. Educational Tools: Teach electronics and programming basics.

Figure 1: ARDUINO UNO R3 Pin diagram

Soil Moisture Sensor

Features

1. **Accuracy:** High-quality sensors can measure soil moisture content with precision, typically within 1-3% accuracy.

2. **Durability:** Many sensors are designed to withstand harsh outdoor conditions, including temperature extremes and exposure to water and chemicals.

3. **Wireless connectivity:** Modern sensors often have wireless capabilities, allowing for remote monitoring and data collection without the need for physical access.

4. **Low power consumption:** Advanced sensors are engineered to operate on minimal power, enabling long-term deployment with battery or solar power sources.

Application

1. **Agriculture:** Farmers use these sensors to optimize irrigation schedules, conserve water, and improve crop yields by maintaining ideal soil moisture levels.

2. **Landscaping and turf management:** Golf courses and parks utilize sensors to maintain healthy grass and plants while minimizing water usage.

3. **Environmental monitoring:** Scientists employ soil moisture sensors to study ecosystem health, track climate change effects, and predict natural disasters like landslides.

4. **Civil engineering**: Construction projects use these sensors to monitor soil stability, assess foundation conditions, and prevent structural damage due to changes in soil moisture.

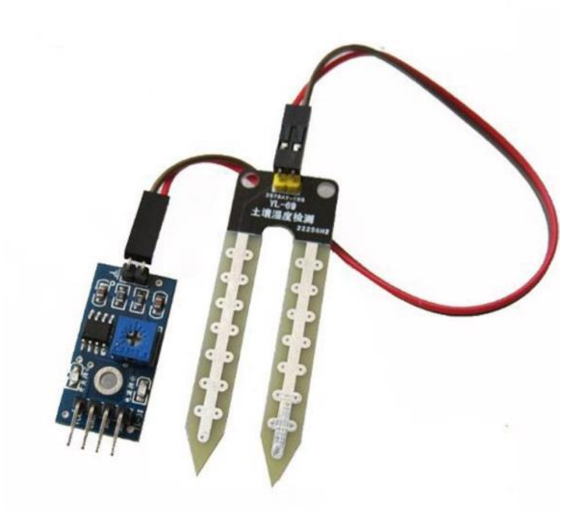


Figure 2: Soil Moisture Sensor

Relay Module

Features

1. **Electrical isolation:** Relay modules provide separation between the control circuit and the switched circuit, protecting sensitive components.
2. **Electrical isolation:** Relay modules provide separation between the control circuit and the switched circuit, protecting sensitive components.
3. **Multiple channels:** Many relay modules come with multiple independent relay channels that can be controlled separately.
4. **Status indicators:** Most modules include LED indicators to show the on/off status of each relay channel.

Applications

1. **Home automation**: Controlling household appliances, lighting systems, and HVAC equipment remotely or via automation systems.

2. **Industrial control:** Switching motors, pumps, valves, and other high-power equipment in manufacturing and process control environments.

3. **Automotive systems:** Managing various electrical subsystems in vehicles, such as lights, power windows, and central locking.

Figure 3: Relay Moodule

5V DC Pump

Features:

1. **Low voltage operation:** Runs on 5V DC power, making it safe and compatible with many electronic systems and batteries.
2. **Compact size:** Typically small and lightweight, allowing for easy integration into space-constrained projects or portable devices.
3. **Low power consumption:** Generally draws minimal current, making it energy-efficient and suitable for battery-powered applications.
4. **Quiet operation:** Often designed to run with minimal noise, important for use in environments where silence is valued.

Applications:

1. **Aquarium and hydroponics:** Used to circulate water in small fish tanks or provide water flow in hydroponic gardening setups.
2. **DIY cooling systems:** Employed in custom PC cooling loops or small electronic device cooling projects.
3. **Robotics and automation:** Incorporated into robotic systems for fluid transfer or as part of automated processes in small-scale manufacturing.
4. **Medical devices:** Utilized in portable medical equipment for precise fluid delivery, such as in wearable drug delivery systems or diagnostic tools.

Figure 4: 5V DC Pump

Power Supply

Features:

1. **Regulated Voltage Output:** Power supplies for Arduino typically provide a stable and regulated output, ensuring that components receive a consistent voltage, usually between 5V and 12V. This prevents fluctuations that could damage sensitive sensors and modules.
2. **Overcurrent and Overvoltage Protection:** Many power supplies come with built-in protection circuits that prevent excessive current or voltage, which safeguards the Arduino board and connected components from potential damage due to surges or shorts.
3. **Multiple Output Modes:** Power supplies often offer multiple output modes, such as DC output for direct use with Arduino or USB output for powering the Arduino through its USB port, providing flexibility in different setups.
4. **High Efficiency:** Modern power supplies are designed to be energy-efficient, converting AC to DC with minimal power loss. This efficiency is crucial for long-running Arduino projects, especially those deployed in remote locations.

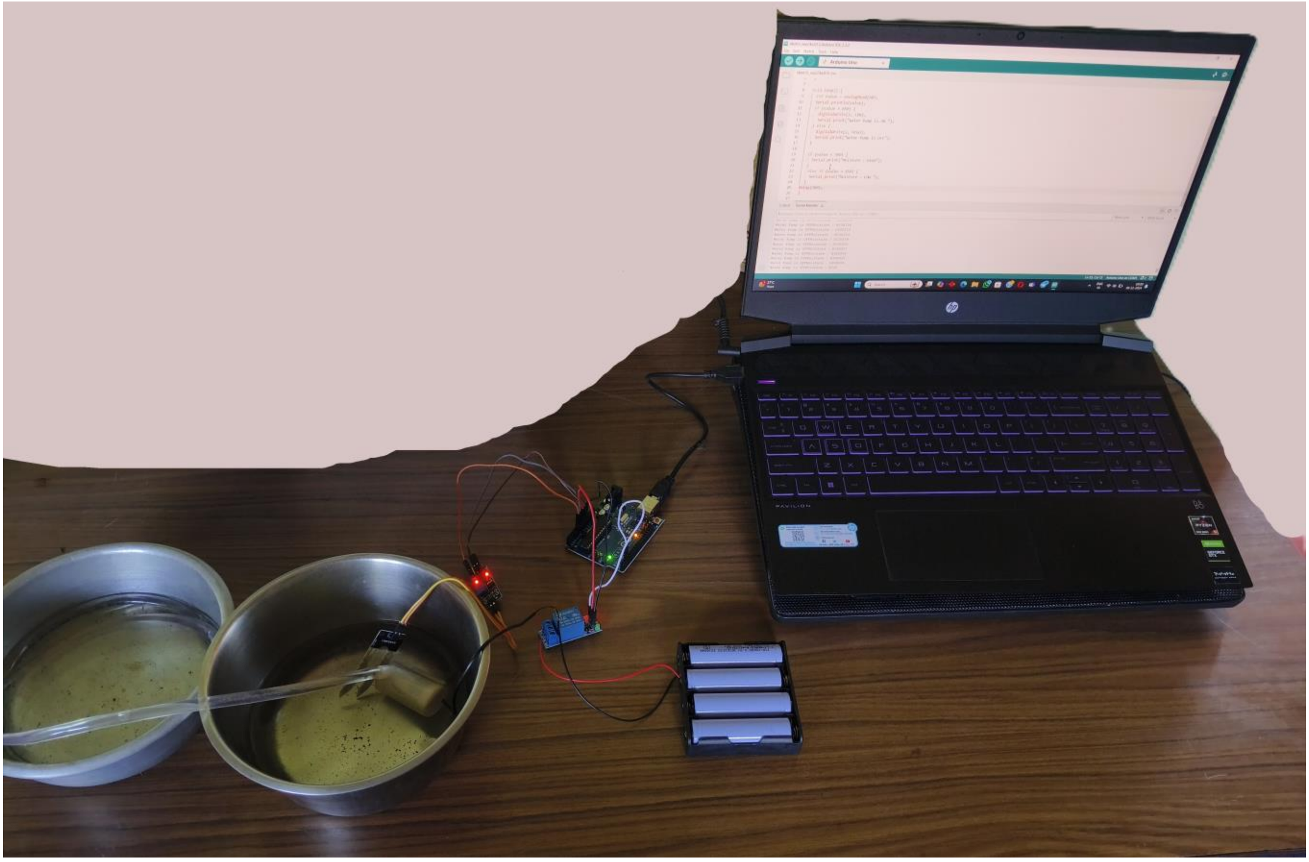
Applications:

1. **Home Automation Systems:** Power supplies ensure continuous operation of Arduino-controlled devices such as smart lights, fans, and security systems in home automation setups.
2. **Robotics:** In robotic systems, power supplies provide stable energy to control motors, sensors, and communication modules, allowing for smooth functioning of autonomous robots.
3. **IoT Projects:** Arduino-based IoT systems require reliable power sources to manage data transmission and sensor networks, enabling devices to stay connected and operational.
4. **Agricultural Monitoring Systems:** In smart agriculture projects, a power supply ensures continuous monitoring of environmental conditions, such as temperature and humidity, using Arduino-controlled sensors.

Figure 5: Power Supply

Project Overview

The Smart Irrigation System project aims to optimize water usage for agriculture and landscaping by leveraging real-time data and automation. It integrates sensors to monitor soil moisture, weather conditions, and plant needs, adjusting watering schedules accordingly. The system reduces water waste, ensures efficient irrigation, and improves plant health by delivering precise amounts of water only when needed. Users can control and monitor the system remotely via a mobile app or web interface. By conserving water resources and enhancing crop yields, this technology supports sustainable agricultural practices and eco-friendly landscape management.



How the System

Works Step-by-step breakdown of the microcontroller based of smart irrigation system

Sensor Data Collection:

- Soil moisture sensors measure water content in the soil.
- Temperature and humidity sensors monitor environmental conditions.
- Rain sensors detect precipitation.

Data Processing:

- The microcontroller receives data from all sensors.
- It processes this information using pre-programmed algorithms.

Decision Making:

- The system compares sensor data with preset thresholds.
- It determines if irrigation is needed based on soil moisture levels.
- Weather forecasts may be factored in if the system is connected to the internet.

Irrigation Control:

- If irrigation is required, the microcontroller activates water pumps or valves.
- It controls the duration and intensity of watering based on calculated needs.

Water Flow Monitoring:

- Flow sensors measure the amount of water being dispensed.
- This data is used to track water usage and detect any leaks in the system.

Real-time Adjustments:

- The system continuously monitors conditions and adjusts irrigation in real-time.
- For example, it may stop watering if it starts raining.

Data Logging:

- The microcontroller records all sensor readings, irrigation events, and water usage.
- This data is stored for later analysis and system optimization.

User Interface:

- Many systems include a mobile app or web interface.
- Users can view system status, override automatic controls, and adjust settings.

Communication:

- In advanced systems, the microcontroller may communicate with a central server.
- This allows for remote monitoring and control.

Power Management:

- The system manages power consumption, often using solar panels for energy.
- It may enter low-power modes when inactive to conserve energy.

System Diagnostics:

- The microcontroller performs regular self-checks.
- It can alert users to malfunctions or required maintenance.

Adaptive Learning:

- Some advanced systems use machine learning algorithms.
- These systems improve irrigation efficiency over time by learning from past data and outcomes.

Types of Information Provided by the Smart Irrigation System

The microcontroller based Smart Irrigation System provides the following types of information:

1. **Soil Moisture Levels:** Monitors and reports real-time soil moisture conditions.
2. **Plant Health Status:** Gives updates on plant conditions based on irrigation and environmental data.
3. **System Performance:** Alerts on any issues or irregularities within the system, such as leaks.
4. **Irrigation History:** Logs past watering events and their duration.
5. **Zone-specific Data:** Provides detailed insights for different garden or field zones.

These types of information help to operate whole system efficiently.

Software and Programming Language Used

The software used to develop the program for the MICROCONTROLLER BASED SMART IRRIGATION SYSTEM is the **Arduino IDE** (Integrated Development Environment). The programming language used to write the program is the **Arduino Programming Language**, which is a simplified version of C/C++ tailored for programming Arduino-compatible microcontrollers, such as the ATmega328p used in this project.

Arduino IDE

Features

- **User-Friendly Interface**: The Arduino IDE offers a simple and intuitive user interface that makes it easy for beginners and professionals alike to write, compile, and upload code to microcontrollers.
- **Built-in Libraries**: The IDE includes numerous built-in libraries that simplify the integration of various sensors, actuators, and modules, such as LDR sensor, and IR sensor used in this project.
- **Serial Monitor**: The IDE provides a serial monitor that allows users to communicate with the microcontroller in real-time, which is useful for debugging and monitoring data output directly from the device.
- **Cross-Platform Support**: The Arduino IDE is available for Windows, macOS, and Linux, making it accessible to users across different operating systems.
- **Open Source**: The Arduino IDE is open-source software, which means it is free to use and continuously improved by a large community of developers.

Applications in the Project

- **Code Development**: The Arduino IDE is used to write the code that controls the ATmega328p microcontroller. The code is written to interface with LDR sensor for ambiance light measurement, and the IR sensor for vehicle or pedestrian detection.
- **Code Compilation and Uploading**: The IDE compiles the written code and uploads it to the ATmega328P microcontroller via a USB connection, enabling the microcontroller to execute the program.
- **Debugging**: The serial monitor feature is used for debugging the program and checking the output of sensors in real-time, allowing for adjustments and optimization of the code.

Arduino Programming Language

Features

- **Simplified Syntax**: The Arduino programming language uses a simplified syntax based on C/C++, making it easier for beginners to learn and write code for microcontrollers.
- **High-Level Abstractions**: It provides high-level abstractions for hardware programming, such as digitalWrite(), digitalRead(), analogRead(), and analogWrite(), which simplify the process of interfacing with hardware components.
- **Wide Community Support**: The Arduino programming language has a large community of users and developers, providing extensive resources, tutorials, and example codes to help with development.
- **Flexible and Scalable**: The language is flexible enough to handle simple tasks, such as blinking an LED, and complex tasks,etc.

Project Code

To create the microcontroller based smart street light using Arduino UNO R3 write a complete Arduino code that integrates the Soil Moisture Sensor for Soil Moisture detection, and the Relay Module to run the 5V dc water pump using the ATmega328P microcontroller.

Prerequisites

1. **Arduino IDE** installed on your computer.
2. **ATmega328P Board Package** installed in the Arduino IDE.

Full Arduino Code

```
void setup() {
  Serial.begin(9600);
  pinMode(2, OUTPUT);
  digitalWrite(2, HIGH);
  Serial.print("SMART IRRIGATION SYSTEM IS ON ");
}

void loop() {
  int value = analogRead(A0);
  Serial.println(value);
  if (value > 850) {
    digitalWrite(2, LOW);
    Serial.print("Water Pump is ON ");
  } else {
    digitalWrite(2, HIGH);
    Serial.print("Water Pump is OFF");
  }

  if (value < 500) {
    Serial.print("Moisture : HIGH");
  }
   else if (value > 850) {
    Serial.print("Moisture : LOW ");
  }

}
```

Future Scope of The Microcontroller Based Smart Irrigation System

1. **AI-powered optimization:** Machine learning algorithms to predict optimal watering schedules based on plant types, soil conditions, and weather forecasts.

2. **IoT integration:** Expanded use of interconnected sensors and devices to create more comprehensive and responsive irrigation networks.

3. **Precision agriculture:** Drone and satellite imagery integration for highly targeted irrigation based on real-time crop health data.

4. **Water source diversification:** Systems that can automatically switch between multiple water sources (e.g., rainwater, greywater, municipal) based on availability and quality.

5. **Predictive maintenance:** Advanced diagnostics to anticipate system failures and schedule preventive maintenance.

6. **Climate change adaptation:** Systems that adjust long-term strategies based on changing climate patterns and water availability.

7. **Blockchain for water management:** Transparent and efficient water allocation and trading systems, especially in water-scarce regions.

8. **Biofeedback systems:** Integration of plant-based sensors to directly measure plant water needs and stress levels.

9. **Urban farming integration:** Tailored systems for vertical farms, rooftop gardens, and other urban agriculture applications.

10. **Gamification and user engagement:** Mobile apps with game-like features to encourage water conservation among users.

11. **Renewable energy integration:** Self-powered systems using solar or wind energy for pumps and controllers.

Conclusion

The smart irrigation system designed using Arduino Uno R3 has proven to be an efficient and practical solution for optimizing water usage in agricultural and garden environments. By integrating soil moisture sensors, temperature sensors, and real-time control mechanisms, the system ensures that plants receive adequate water without over-irrigating, thus conserving water resources. The automation provided by the Arduino platform reduces the need for manual intervention and increases the precision of irrigation, leading to healthier plant growth and a more sustainable agricultural practice. Furthermore, the scalability of the system makes it adaptable for various applications, from small home gardens to larger agricultural fields. The project demonstrates the potential for IoT-based solutions in agriculture, paving the way for further advancements in smart farming techniques.

In summary, the smart irrigation system is a cost-effective, reliable, and environmentally friendly solution that contributes to more sustainable water management practices. This project not only enhances the efficiency of irrigation but also promotes the integration of technology in everyday farming operations.

References

ArduinoGetStarted (n.d.) Arduino - Automatic Irrigation System, ArduinoGetStarted. Available at: https://arduinogetstarted.com/project/automatic-irrigation-system (Accessed: 8 October 2024).

Cirkit Designer (n.d.) Arduino Nano-Controlled Automated Irrigation System with Moisture Sensing and Water Level Detection, Cirkit Designer. Available at: https://docs.cirkitdesigner.com/automated-irrigation-system (Accessed: 8 October 2024).

Instructables (2024) Smart Irrigation System Using Arduino Uno, Soil Moisture Sensor, and Humidity Sensor, Instructables. Available at: https://www.instructables.com/Smart-Irrigation-System-Using-Arduino-Uno-Soil-Moi/ (Accessed: 8 October 2024).

Praveen, M., Kumar, R. and Kumar, A. (2022) Automatic irrigation system using Arduino, ResearchGate. Available at: https://www.researchgate.net/publication/358024876_Automatic_irrigation_system_using_Arduino (Accessed: 8 October 2024).

Patil, P., Pawar, M. and Kadam, P. (2020) Smart irrigation system using Arduino and soil moisture sensor, International Journal of Engineering Research and Technology, 9(12), pp. 25-30. Available at: https://www.ijert.org/smart-irrigation-system-using-arduino-and-soil-moisture-sensor (Accessed: 8 October 2024).

Lakshmanan, K. (2021) IOT Based Smart Irrigation System Using Arduino, International Journal of Innovative Technology and Exploring Engineering, 11(4), pp. 45-50. Available at: https://www.ijitee.org/download/volume-11-issue-4/IOT-Based-Smart-Irrigation-System.pdf (Accessed: 8 October 2024).