SMART IRRIGATION SYSTEM USING AVR MICROCONTROLLER

Sujal Malapure
Entc
Vishwakarma Institute of Information Technology
pune
sujal.22310176@viit.ac.in

Shrihari Ganbavale
Entc
Vishwakarma Institute of Information Technology
Pune
shrihari.22310150@viit.ac.in

Atharva Bhavsar
Entc
Vishwakarma Institute of Information Technology
Pune
atharva.22310666@viit.ac.in

Purva Sapkal
Entc
Vishwakarma Institute of Information Technology
Pune
purva .22310197@viit.ac.in

Dr. Pravin Gawande
Entc
Vishwakarma Institute of Information Technology
Pune
pravin.gawande@viit.ac.in

***Abstract*— This project demonstrates the design and implementation of a Smart Irrigation System that is built around an AVR microcontroller. This system automates the irrigation process by sensing the dry and wet conditions of the soil and the environmental conditions as well. The soil moisture sensor is used to detect how dry the soil is, and the DHT11 sensor senses the temperature and humidity. The microcontroller will turn the water pump ON or OFF based on the sensors' readings through a relay module to automate the watering of plants and allows for more water-efficient irrigation. All real-time data from the system, including the sensor values and state of irrigation is displayed on a 16x2 I2C LCD. Every component of the system is connected using a single-layer PCB to make it compact and reliable. The intended goal of the Smart Irrigation System is to reduce the amount of wasted water through irrigation by minimizing labor efforts and improve sustainable agricultural practices across the world.**

Keywords—Smart Irrigation, AVR microcontroller, DHT11, I2C LCD Display, Soil Moisture Sensor, Water management, Sustainable farming.

I. INTRODUCTION

Agriculture is essential for life by providing food and supporting global economies. However, traditional farming practices are usually reliant on hand irrigation processes which result in overuse of water, improper crop handling, and higher labour costs, to name a few. Often, irrigation is performed even though the irrigator is not accurately taking into account the moisture content in the soil and or the environmental conditions. Consequently, excessive irrigation occurs which results in crop death as well as in reducing crop productivity on poor or unproductive soils.

To tackle these problems, we now require smart and automated solutions. The project Smart Irrigation System Using AVR Microcontroller, presents a way to utilize water in a better manner while automating irrigation depending on real-time environmental feedback. The system uses an AVR micro-controller along with a soil moisture sensor and DHT11 temperature and humidity sensor to monitor the field environment continuously. While the soil moisture is under a threshold the AVR will activate a relay that activates the water pump, thus, watering the fields and crops only when necessary. The temperature and humidity readings help understand the field condition as well, which are the last few factors that will influence needs for irrigation.

The real-time display of sensor readings and system state is provided by means of a 16x2 I2C LCD display. This display makes it simple for the operator to see how the sensors are performing. The system is arranged on a single-layer PCB, with all components neatly organized. This helps improve the reliability of the system and simplifies maintenance of the board. The system presented aims to reduce human interaction, reduce water usage, and implement a more sustainable farming method through automation. In a world where smart farming is appearing to be more important than ever, this project demonstrates an affordable and practical solution for small-scale farmers, garden owners, and agricultural researchers. The conciseness of the design, with simple operator intervention and smart operation makes a good step towards modernizing irrigation systems for agriculture.

II. LITERATURE SURVEY

Ashok et al. [1] introduced an automated irrigation system for microcontroller, where real time soil parameters were monitored such as moisture, temperature and pH. A PIC microcontroller was used to control a motor or sprinkler, and the system automated irrigation to save water, and only operated when needed. The system allowed the farmer to communicate with the irrigation system via GSM to enable SMS control remotely. The authors also examined using WSNs and android applications, and observed matters of energy consumption and user friendliness for rural farmers.

Pasha et al. [2] developed an AVR based irrigation control system using an ATmega32 microcontroller. The control system operates using resistance based soil moisture sensors and ultrasonic water level sensors in order to automate drip irrigation. The AVR microcontroller is used to control a relay driven motor which is triggered based on thresholds set by the sensor input and provides feedback through an LCD. This system facilitates water use efficiency with minimal human intervention. The sensor data is communicated to a central controller which has the potential to alert the farmer - increasing yield while conserving resources.

Kanumalli et al. [3] proposed an Internet of Things (IoT)-based smart irrigation system using NodeMCU (ESP8266), soil moisture and DHT11 sensors, and a DC pump which can be controlled using a relay, users can monitor the real time data from the soil moisture and DHT11 sensors, so the irrigation process fully went automatically based on the sensor data. Using MQTT (Message Queue Telemetry Transport) protocol and Blynk mobile application to ensure that the datasets from the sensors are delivered securely via SSL, notifications of irrigation duration is sent after a certain period of time elapsed. They also leveraged cloud connectivity, which makes the system very flexible and reduces risks by allowing further expansion and upgrade on the hardware of the IoT smart irrigation system with minimal to no additional costs. The authors also acknowledged that the system can potentially integrate with machine learning for additional features.

M. Manoj Venkata Sai et al. [4] proposed an IoT Smart Agriculture Project that used temperature sensors, moisture sensors, humidity sensors, and pH sensors for monitoring crops. They used a feedback control mechanism having a centralized control unit that regulates water irrigation onto the field in real-time by using the instantaneous moisture, temperature, humidity, and pH values provided by the sensor unit as per the need of the crop, the controller decides to turn on or off the water irrigation using Arduino NodeMCU. Their project focused on providing the right amount of water which eventually increased the efficiency of the farm.

Miss. Bhagyashree A. Tapakire et al. [5] proposed a project in their paper of a smart irrigation system using various sensors, control valves, and pump motors and combined it with IoT, cloud computing, and optimization tools. This system uses various low-cost sensors to sense soil moisture and temperature. The project consists of a Raspberry Pi, soil moisture sensor, temperature sensor, camera, control valve, and a motor driver. The camera is used to keep watch on plant growth. The sensed data is stored in the Thing Speak cloud service for monitoring the data and its storage. This system regulates irrigation work without any manual intervention using IoT technology. The paper concluded at the point which said that this agriculture monitoring system served as a reliable and efficient solution for farmers' corrective action can be taken and we can achieve uniform irrigation with this proposed system.

III. METHODOLOGY

The Smart Irrigation System development methodology is divided into several sections, moving from hardware setup to operation of the system. Each section is intended to ensure effective, dependability, and real-time irrigation based on environmental factors.

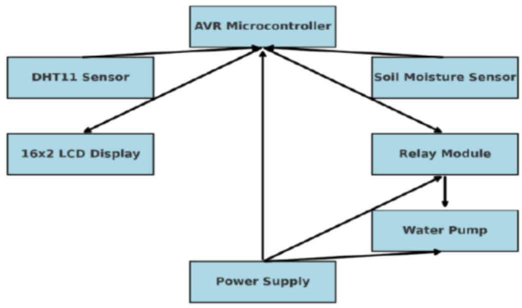

Fig [1] General Flow Diagram of Implementation

1. Research Design

The design of the study was experimental. The study's objective was to create an irrigation hardware prototype that took environmental factors into account. All of the essential parts (AVR microcontroller, sensors, relay module, and LCD display) were included in a single-layer PCB design. Without the need for human intervention, the system was built to gather environmental data in real time and process it to turn on (or off) outputs, such as a water pump. To find out if the system could react to different environmental inputs, the experimental setup was tested in controlled soil conditions (dry and wet).

2. Data Collection Methods

2.1. Sensor Data Acquisition

- The water content of the soil was measured using a soil moisture sensor. In order to provide continuous soil moisture readings, it was interfaced with the microcontroller's analog input (PC0).
- A **DHT11 sensor** was used to measure **temperature** and **humidity**, connected to digital input (PD2).

2.2. Real Time Monitoring

- The measured data were visible on a 16x2 I2C LCD, which was connected through the I2C protocol (PC4 for SDA and PC5 for SCL).
- During testing, observations were noted manually to assess the reliability of the sensors and the performance of the system.

2.3. Operational Trigger

- Soil moisture readings informed our decisions to turn on/off the relay-controlled water pump.

3. Data Analysis Strategies

3.1. Threshold-Based Analysis

- A soil moisture threshold had been established (co-efficient value through calibration).
- Data from the soil moisture sensor was compared against this threshold data to see what the proper irrigation action at the time would be (pump ON or OFF).

3.2. Performance Evaluation

- System response times of the system (dry soil into pump ON situation) were judged and measured manually.
- The relative stability and accuracy of the values displayed on the LED were judged by comparing

sensor readings in various climatic conditions (dry soil, moist soil, temperature and humidity variation).

3.3. Reliability Testing

- The continuous use of the system was reviewed over an extended period (few hours) for evidence of reliability and to look for failures in relay action, variability of sensors, or power stability.

4. Ethical Considerations

4.1. Environmental Consideration

- Testing Phase: The System was used in a controlled environment without interfering with any live plants of natural environments during any of the experimental work.

4.2. Safety

- Utilization of proper Electrical insulation, including proper relay circuit design, to reduce risks. The System was working with water and electrical current so safety measures needed to be prioritized.

4.3. Integrity of data

- ABC performed an accurate account of all sensor readings and behaviors of the System without altering any results for truthful recordings of data.

4.4. Sustainability

- Overall the project supports sustainable agriculture by improving irrigation efficiency, and therefore providing more accurate and timely management on the use of natural resources in the environment.

IV. IMPLEMENTATION

1. Hardware Implementation

1.1. Microcontroller Setup

- An AVR microcontroller (ATmega32A is an example) was used because it is robust and has a wide variety of I/O (input/output) interfaces.
- It was set up to read in signals (sensor inputs) and control relays (outputs) as needed.

1.2. Sensor Connections

- The soil moisture sensor was connected to an analog input for example PC0.
- The DHT11 temperature and humidity sensor was connected to digital input for example PD2.
- I used pull-up resistors where required.

1.3. LCD Display

- I ran a 16x2 I2C LCD with:
  PC4 (SDA - data line)
  PC5 (SCL - clock line)
- I used I2C communication to reduce wiring, which gives us just two data lines to wire.

1.4. Relay and Water Pump

- The relay control input was connected to digital pin PC1.
- The relay output controlled the +ve terminals of the pump.
- The pumps -ve terminal was connected directly to the GND rail.
- The relay module was powered from the 5V rail ensuring a sufficient current supply.

1.5. Power Distribution

- A **5V rail** and a **GND rail** were established on the PCB to supply all components:
  5V to LCD, DHT11, soil sensor, and relay module
  Common ground for all returns

2. PCB Design and Fabrication

2.1. A PCB with a 1-layer PCB was designed using available online PCB design software (EasyEDA or KiCad as examples).

2.2. In addition care was used to minimize signal noise by ensuring sensor lines were away from the relay control lines.

2.3. Standard components (resistors, connectors) and appropriate PCB headers were used to make assembly and maintenance easier.

2.4. The PCB was fabricated, soldered, and tests for continuity were conducted before mounting the components.

3. Software Development

3.1. The firmware was all written in CPP.

3.2. We wrote a program with many core modules, including:

- **Initialization functions:** for the ADC, I2C communications, LCD initialization, and GPIO pins.
- **Sensor reading functions:** for reading the Soil Moisture and DHT11 sensors.
- **Decision making functions**: which were made on the levels of Soil Moisture, threshold, using prescribed levels.
- **Peripheral control functions:** relay turning the water pump ON or OFF.
- **Display update functions:** dynamically displaying temperature, humidity, soil moisture, and pump status on the LCD.

3.3. Threshold Calibration

- A soil moisture threshold value was experimentally determined to decide when the soil is considered dry.

4. Working Flow

4.1. The microcontroller continuously reads sensor values

4.2. If the soil moisture value drops below the threshold

- The relay is activated.
- The water pump starts irrigating the field

4.3. If the soil moisture value exceeds the threshold:

- The relay is deactivated
- The water pump turns off

4.4. Throughout the process, the LCD displays:

- Current temperature (°C)
- Humidity (%)
- Soil moisture level
- Water pump status (ON/OFF)

5. Testing and Validation

5.1. The system was tested for real soil conditions.

5.2. The conditions of dry and wet soil were simulated in the real environments to validate sensor performance, and operation of the pump.

5.3. Dry and wet soil conditions were simulated in the real environments to validate sensor performance, and operation of the pump.

5.4. Relays were monitored in terms of switching and activation delay and minimize those delays.

5.5. Continuous operation of the system was observed to evaluate stability and reliability in the long term.

V. Results and Discussion

The Smart Irrigation System was able to automate irrigation control and monitor soil moisture, temperature, and humidity using a relay driven water pump. The results indicate the system autonomously interpreted the environmental conditions precisely and turned the water pump on or off when the environmental conditions required in order to conserve water with no human observation. The real-time updates of the sensor data on the 16x2 I2C LCD enabled human observation; the system could recognize when the soil was dry and switched on the pump, fulfilling the project objective of effective water management. The hardware design was validated by the successful operation of the system over a single-layer PCB with no hardware failures after extended duration.

ACKNOWLEDGMENT *(Heading 5)*

We would like to take this chance to thank everyone that supported and guided us through the course of this research project called Smart Irrigation System using AVR Microcontroller.

First of all, we would like to thank our honourable mentor and guide, Dr. Pravin Gawande, for always providing us with essential advice, continuous support, and motivation through this research.

For the commitment, preparation, cooperation, and teamwork which led to this project, we would also like to thank our teammates, Sujal Malapure, Purva Sapkal, Shrihari Ganbavale, and Atharva Bhavsar.

We would also like to recognize the Department of Electronics and Telecommunication Engineering at the Vishwakarma Institute of Information Technology in Pune, for providing us with the equipment and materials needed to complete our research..

VI. Conclusion and Future Scope

This study clearly demonstrated the process of using an AVR microcontroller for designing and implementing a smart irrigation system. Based on environmental monitoring in real time, we were able to automate the irrigation process appropriately through the use of a relay module, soil moisture sensor, DHT11 temperature and humidity sensor, and a 16x2 I2C LCD. In summary, a single-layer PCB enabled a compact, cheap and reliable hardware solution for small agricultural and gardening applications. The system was fully capable of achieving the primary aims of enhancing water use, decreasing manual intervention, and easily accessing status updates through a simple, powerful and effective LCD interface. The experiments showed that the system was able to successfully and accurately read the soil moisture content, act to control the water pump, and feed it constant updates without the need for an external network. In resource-limited, or rural environments, this design represents a cost effective, low tech and independent alternative to more complicated IoT based agriculture systems. Overall, this project improves the sustainability of agriculture by provide efficient water management, and by providing alternate forms of agricultural practice with high efficiency relative to low tech and complexity.

While the implemented system achieved its goals, there are a couple of opportunities to enhance its versatility and usefulness:

- Multi-Zone Irrigation:

  By using multiple soil moisture sensors and relays to independently manage different agricultural zones, allow for tailored irrigation with a variety of crop types.

- Advanced Environmental Monitoring:

  You might want to add additional sensors to the suite if you'd like to increase the accuracy of the decision-making, additional sensors like light intensity, pH and rainfall sensors could be added.

- Weather Prediction Systems:

  Dynamically alter irrigation scheduling based on weather forecasts and expected temperature (weather forecasting APIs).

- Energy-Efficient Designs:

  Consider utilizing solar or energy harvesting for remote or off-the-grid agronomic applications.

REFERENCES

[1] J. Ashok et al., "Microcontroller Based Automatic Real Time Water Irrigation Management System," M. Kumarasamy College of Engineering, Karur, 2017.

[2] I. A. Pasha et al., "Automated AVR Based Irrigation System," Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET), vol. 6, no. 4, pp. 1035–1041, Apr. 2018, doi: 10.22214/ijraset.2018.4174.

[3] S. S. Kanumalli et al., "Automated Irrigation Management System using IoT," in Proc. 6th Int. Conf. Intell. Comput. Control Syst. (ICICCS), 2022, pp. 476–482, doi: 10.1109/ICICCS53718.2022.9788313.

[4] M. Manoj Venkata Sai, K. Subba Rao, N. Vamsi Krishna, and M. Vasantha Lakshmi, "IoT Based Smart Agriculture," IRE Journals, Vol. 1, Issue 8, pp. 1-5, Feb 2018.

[5] Miss. Bhagyashree A. Tapakire, Dr. Manasi M. Patil, "IoT Based Smart Agriculture Using Thingspeak," International Journal of Engineering Research & Technology (IJERT), Vol. 8, Issue 12, pp. 1-5, Dec 2019.