MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

DEPARTMENT OF MECHATRONICS

AUTOMATED HOME IRRIGATION SYSTEM

MTE 3262-IIoT LAB

Submitted by
ISHA HARISH (200929098)

DEPARTMENT OF MECHATRONICS

MANIPAL INSTITUTE OF TECHNOLOGY, MANIPAL ACADEMY OF HIGHER EDUCATION (MAHE), MANIPAL

TABLE OF CONTENTS

1. ABSTRACT

Automation and control have advanced significantly as a result of the Internet of Things (IoT). An automated home irrigation system that can be managed and seen remotely over the Internet has been developed thanks to the integration of IoT and the Arduino microcontroller.
This system, which is built on Internet of Things concepts, employs a variety of sensors to track temperature, humidity, and soil moisture content. These sensors are linked to an Arduino microcontroller, which uses a motorized valve to regulate plant watering.
Additionally, the system has an Industrial Internet of Things (IIoT) module that enables remote computer or mobile device control and monitoring of the system. This implies that even while they are not at home, users can still monitor the machine and change its settings.
In addition, the IIoT module offers a platform for data collecting and analysis, giving customers the ability to understand environmental conditions and make well-informed decisions on plant watering. Additionally, by optimizing the irrigation schedule with this data, water usage can be decreased and water conservation can be encouraged.
In conclusion, the management of our farms and gardens could be completely transformed by the integration of IoT and IIoT in an automated home irrigation system.
By providing remote control and monitoring capabilities, users can save time and effort while promoting water conservation and ensuring optimal plant growth.

2. INTRODUCTION

One of the most important aspects of gardening is watering your plants, but doing so by hand may be labor- and time-intensive. To get around these problems, plants can be automatically watered according to their moisture requirements by installing an Arduino-powered automated home irrigation system.

An Arduino microcontroller is attached to a soil moisture sensor, which measures the moisture content of the soil.

Because this kind of system works in both indoor and outdoor environments, it's perfect for greenhouses, home gardens, and even larger-scale farming. By making sure that plants receive the proper amount of water at the right time, it not only saves time and labor but also encourages water conservation.

Additionally, the system may be tailored to meet the unique requirements of various plants thanks to the usage of an Arduino microcontroller. The system can be configured to modify the watering schedule in accordance with the needs of individual plants, such as those that need more water than others.

The goal of this project is to use Arduino to measure soil moisture in order to develop an automated home irrigation system. We will go over the necessary parts, the logic for programming, and how to put the system together and test it. By the time the project is finished, you will have a working system that can water your plants automatically according to their moisture requirements, improving the convenience and enjoyment of gardening.

3. PROBLEM STATEMENT

The Automated Home Irrigation System project aims to solve the problem of manual plant watering and promote efficient and effective plant growth through automation and customization.

4. OBJECTIVE.

The following are the project's goals for the automated home irrigation system:

- to create a home irrigation system that is automated, capable of measuring soil moisture content and automatically watering plants according to their moisture requirements.
- can adapt the system to the unique requirements of various plants, making sure that every plant gets the ideal quantity of water.
- to encourage water conservation by using water resources more effectively and minimizing water waste.
- To make gardening more pleasurable by offering a practical and efficient replacement for hand plant watering.
- to design a dependable and efficient system that can be applied in home gardens, greenhouses, and commercial farming, among other contexts.
- To demonstrate the potential of the Internet of Things (IoT) in promoting efficient and effective plant growth through automation and customization.

By achieving these objectives, the Automated Home Irrigation System project aims to provide a practical solution to the challenges of manual plant watering and promote sustainable and efficient plant growth through the use of technology.

5. COMPONENTS USED

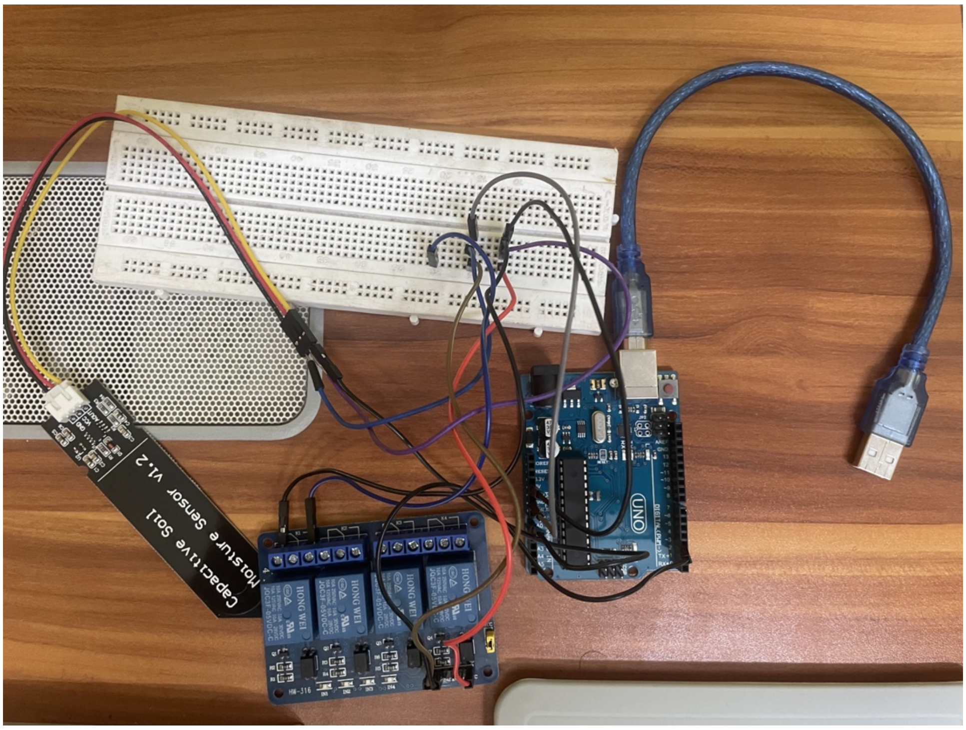
- Arduino Uno – 1
- Arduino Cable
- Capacitive Soil Moisture Sensor – 1
- Relay – 1 (Represent irrigation system) (One relay used out of a 4 relay module)
- Jumper Wires M-M
- Jumper Wires M-F
- Bread Board
- Arduino IDE

6. METHODOLOGY

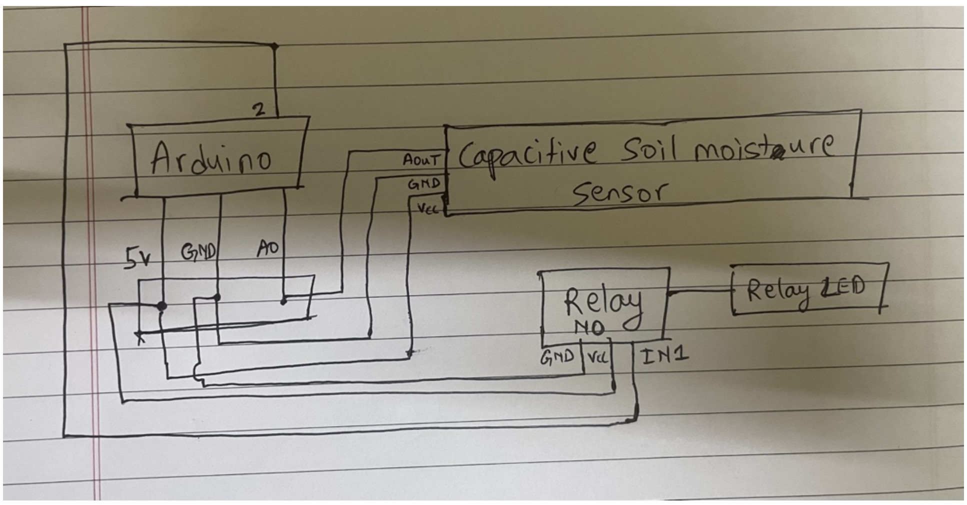
The Arduino is the Main Brains of the system. The Power output (5V) and Ground pins of the Arduino Are connected to a bread board so that it can be distributed. The soil Moisture Sensor is connected to the power and ground pins on the bread board and the analog output of the sensor is connected to the A0 analog in pin of the arduino. The power and ground pins of the relay are connected to the bread board, and the digital input pin of the relay is connectd to the pin 2 of the arduino (digital out mode).

The Soil Moisture sensor is caliberated for open air reading (corresponding to least moisture level), and water reading (maximum moisture level). These values are mapped between 0 and 100 to get a percentage value when the sensor is in soil. The logic of the code is such that above a set moisture percentage, the irrigation system (represented by the NO relay) remains off. If the moisture readings drop below the set percentage, the irrigation system turns on.

The moisture readings are taken every second (adjustable), and the readings are also displayed on the serial monitor.



Physical Layout



Schematic Diagram

```
CalliberationCode_Capacitive_SoilMoisture_Sensor
int MoistureLevel = 0;

void setup() {
  Serial.begin(9600);

}

void loop() {
  MoistureLevel = analogRead(A0);
  Serial.println(MoistureLevel);

  delay(1000);

}
```

Simple Code for Caliberation

```
IIoT_Lab_MiniProject_MainCode_SoilMoisture
const int OpenAirReading = 472;   //calibration data 1
const int WaterReading = 195;     //calibration data 2
int MoistureLevel = 0;
int SoilMoisturePercentage = 0;
int RelayPin = 2;

void setup() {
  Serial.begin(9600); // open serial port, set the baud rate to 9600 bps
  pinMode(RelayPin, OUTPUT);

}

void loop() {
  MoistureLevel = analogRead(A0);  //update based on the analog Pin selected
  SoilMoisturePercentage = map(MoistureLevel, OpenAirReading, WaterReading, 0, 100);

  if (SoilMoisturePercentage >= 30)
  {
    Serial.print("Irrigation System Off, Moisture Percentage = ");
    Serial.println(SoilMoisturePercentage);
    digitalWrite(RelayPin, HIGH);
  }
  else
  {
    Serial.print("Irrigation System On, Moisture Percentage = " );
    Serial.println(SoilMoisturePercentage);
    digitalWrite(RelayPin, LOW);
  }

  delay(1000);

}
```

Main Code

```
Irrigation System On, Moisture Percentage = -13
Irrigation System On, Moisture Percentage = -11
Irrigation System On, Moisture Percentage = -18
Irrigation System On, Moisture Percentage = -17
Irrigation System On, Moisture Percentage = -16
Irrigation System On, Moisture Percentage = -15
Irrigation System On, Moisture Percentage = -23
Irrigation System On, Moisture Percentage = -25
Irrigation System On, Moisture Percentage = -27
Irrigation System On, Moisture Percentage = -26
Irrigation System On, Moisture Percentage = -26
Irrigation System On, Moisture Percentage = -24
Irrigation System On, Moisture Percentage = -26
Irrigation System On, Moisture Percentage = -25
Irrigation System On, Moisture Percentage = -28
Irrigation System On, Moisture Percentage = 9
Irrigation System On, Moisture Percentage = 8
Irrigation System Off, Moisture Percentage = 46
Irrigation System Off, Moisture Percentage = 51
Irrigation System On, Moisture Percentage = 10
Irrigation System On, Moisture Percentage = 3
Irrigation System Off, Moisture Percentage = 32
Irrigation System Off, Moisture Percentage = 42
Irrigation System Off, Moisture Percentage = 40
Irrigation System Off, Moisture Percentage = 40
Irrigation System Off, Moisture Percentage = 36
Irrigation System Off, Moisture Percentage = 37
Irrigation System Off, Moisture Percentage = 41
Irrigation System Off, Moisture Percentage = 40
Irrigation System Off, Moisture Percentage = 43
Irrigation System Off, Moisture Percentage = 42
Irrigation System Off, Moisture Percentage = 42
Irrigation System Off, Moisture Percentage = 42
Irrigation System Off, Moisture Percentage = 41
Irrigation System Off, Moisture Percentage = 45
```

Data on Serial Monitor

This project can be furthered by connecting an ESP-8266 wifi transceiver to the arduino. Using this, we can wirelessly communicate and send signals to the arduino and manually control the functioning of the irrigation system.

7. RESULTS AND CONCLUSION

The project shows us the benefits of an automated system and how it can be more sophisticated with the use of multiple sensors. Such a system can be viewed and controlled in real time, with the comfort of not being physically present or constantly keeping a check on it.

8. REFERENCES

- Guide to use Capacitive Soil Moisture Sensor
- Guide to use 4 relay module