DELHI TECHNOLOGICAL UNIVERSITY
DEPARTMENT OF ELECTRONICS AND COMMUNICATION

Submitted To:

Mr. M. Ganesh

Submitted By:

NIZA GARG(23/EC/143)

ACKNOWLEDGEMENT

First and foremost, we would like to thank Mr. M. Ganesh of The Department of Electronics and Communication, for his invaluable advice and mentorship. He has been a constant source of inspiration and motivation for us. We are deeply indebted to him for his patience, generosity, and expertise.

We wish to express my thanks to all Teaching and Non-teaching staff members of the Department of ELECTRONICS AND COMMUNICATION ENGINEERING who were helpful in many ways for the completion of the project.

Finally, we extend our indebtedness to our parents and all other family members for their patience and help extended in this end over.

TABLE OF CONTENT

CHAPTER NO TITLE

CHAPTER-I

INTRODUCTION

In the realm of modern agriculture and home gardening, automation has become a key factor in efficient and effective plant care. This report presents an innovative solution to plant watering - an Arduino-Based Plant Watering System.

This system leverages the power of Arduino, a widely-used open-source electronics platform, to automate the watering of plants based on the moisture content of the soil. The primary goal of this system is to ensure optimal watering conditions for plants, thereby promoting healthier growth and reducing the manual effort involved in plant care.

The system is equipped with a soil moisture sensor that continuously monitors the moisture level of the soil. When the moisture level falls below a predetermined threshold, the Arduino controller activates a water pump to hydrate the plants. Once the optimal moisture level is restored, the pump is automatically turned off, preventing overwatering.

A unique feature of this system is the integration of an LCD panel. This panel displays real-time information about the temperature and humidity of the surroundings.

This report will delve into the design, implementation, and performance of the Arduino-Based Plant Watering System, providing a comprehensive overview of its benefits and potential applications in various gardening and agricultural contexts.

CHAPTER-2

AIM OF THE PROJECT

Since nowadays, in the age of advanced technology and electronics, the life style of the human should me smart, simpler, easier and much more convenient. So, therefore; there is a need for many automated systems in human's daily life routine to reduce their daily activities and jobs.

Here an idea of one such system named as automatic plant watering system is very useful. As many people are facing a lot of problem watering the plants in the garden, especially when they away from the home.

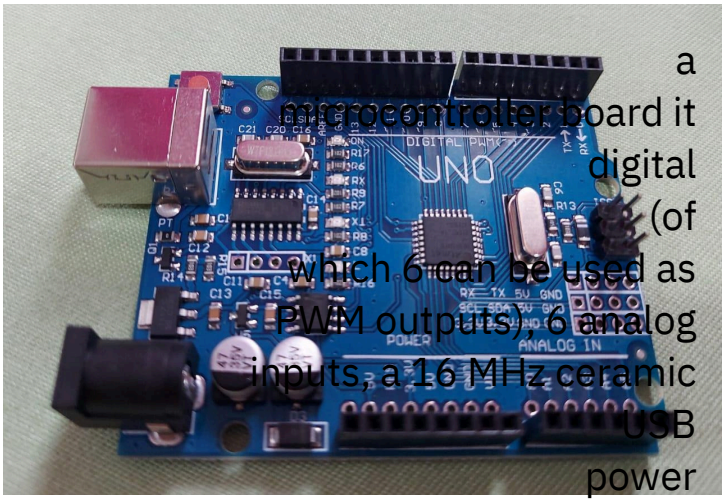
This model uses sensor technologies with microcontroller in order to make a smart switching device to help millions of people. In its most basic form, system is programmed in such a way that soil moisture sensor which senses the moisture level from the plant at particular instance of time, if moisture level of the sensor is less than the specified value of threshold which is predefined according to the particular plant than the desired amount of water is supplied to plant till it's moisture level reaches to the predefined threshold value.

System involves humidity and temperature sensor which keep tracks the current atmosphere of the system and has an influence when watering happens.

CHAPTER 3

COMPONENTS USED

3.1 ARDUINO UNO



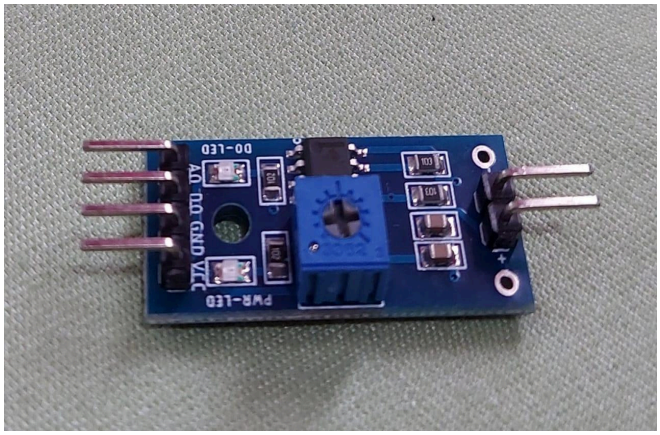
Arduino UNO is a microcontroller board it has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connec-tion, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simp-ly connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

SOFTWARE USED : ARDUINO IDE

Before you can start doing anything with the Arduino, you need to download and install the Arduino IDE (integrated development environment). From this point on we will be referring to the Arduino IDE as the Arduino Programmer.

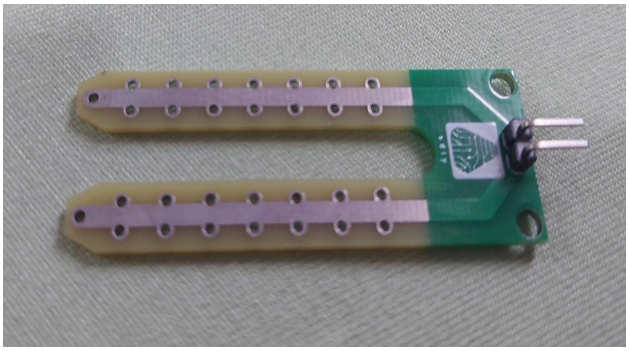
The Arduino Programmer is based on the Processing IDE and uses a variation of the C and C++ programming languages.

3.2 SOIL MOISTURE SENSOR



The Arduino Soil Moisture Sensor is a simple but powerful tool used in various gardening and agricultural applications. It plays a crucial role in monitoring the moisture level of the soil, which is vital for the healthy growth of plants.

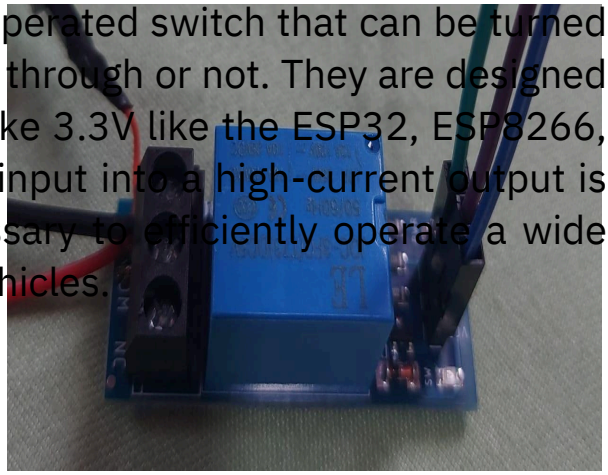
The sensor works on the principle of capacitive sensing It consists of two conductive plates that measure the dielectric constant of the soil. The dielectric constant is an indicator of the water content in the soil. When the soil is dry, the dielectric constant is low, and when the soil is wet, the dielectric constant is high.



The sensor is easy to interface with an Arduino board. It has an analog output that can be connected to one of the analog input pins on the Arduino. The Arduino can then read the analog voltage to determine the moisture level of the soil.

The Soil Moisture Sensor is a valuable tool in the realm of modern agriculture and home gardening, enabling efficient and effective plant care through automation.

3.3 RELAY MODULE



The relay module is an electrically operated switch that can be turned on or off deciding to let current flow through or not. They are designed to be controlled with low voltages like 3.3V like the ESP32, ESP8266, etc, or 5V like your small electrical input into a high-current output is no easy feat, but this task is necessary to efficiently operate a wide range of standard appliances and vehicles.

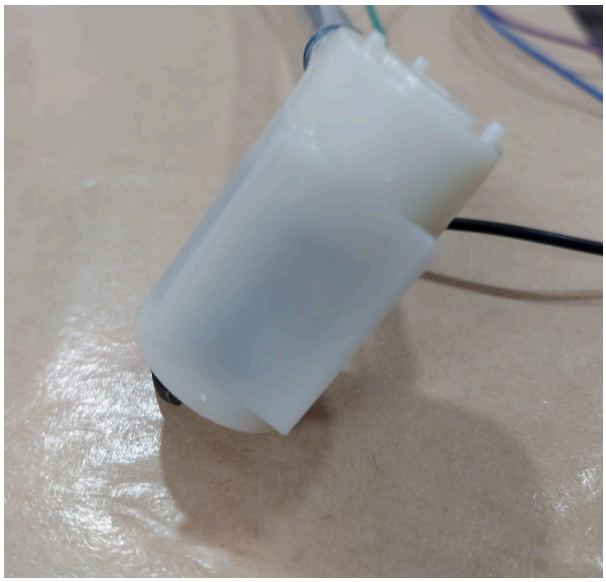
Arduino. Converting a

The relay module is essential because the Arduino cannot directly control high-power devices like a water pump. The relay acts as an electrical isolator, protecting the Arduino from potential damage due to the high power and voltage associated with the pump.

Relays are normally used in the control panels, manufacturing and building automation to control the power along with switching the smaller current values in a control circuits.

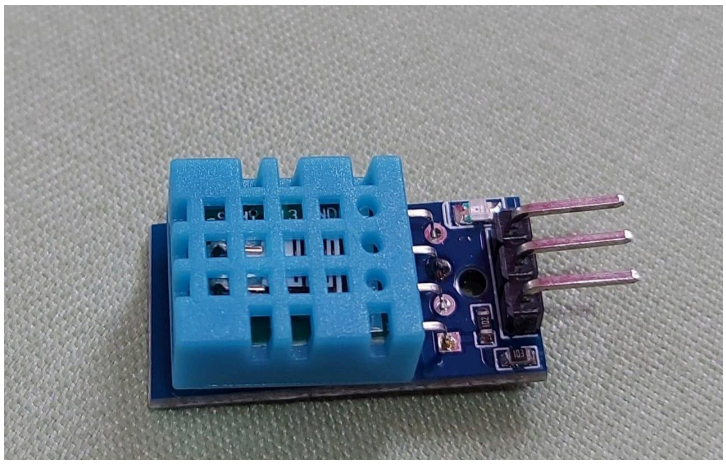
3.4 DCMOTOR

DC motors are configured in many types and sizes, including brush less, servo, and gear motor types. A motor consists of a rotor and a permanent magnetic field stator. The magnetic field is maintained using either permanent magnets or electromagnetic windings. DC motors are most commonly used in variable speed and torque.



DC motors are often used to **drive water pumps** in automatic plant watering systems. The speed and direction of the DC motor can be controlled electronically, allowing for **precise regulation of water flow**. This control is essential for ensuring that plants receive the appropriate amount of water. DC motors are known for their **efficiency**, which is important in systems that may need to operate continuously or intermittently over long periods.

3.5 DHT-11 SENSOR



The DHT sensor, particularly the DHT11 is used in automatic plant watering systems to measure environmental parameters such as temperature and humidity.

While the DHT sensor primarily measures temperature and humidity, it indirectly helps in assessing soil moisture levels.

In addition to triggering watering events, the DHT sensor helps prevent overwatering by ensuring that plants are watered only when necessary.

DHT sensor plays a crucial role in automatic plant watering systems by providing essential environmental data that guides watering decisions, helps prevent overwatering, and optimizes plant health and growth. Its integration enhances the efficiency and effectiveness of the system, ensuring that plants receive the right amount of water at the right time.

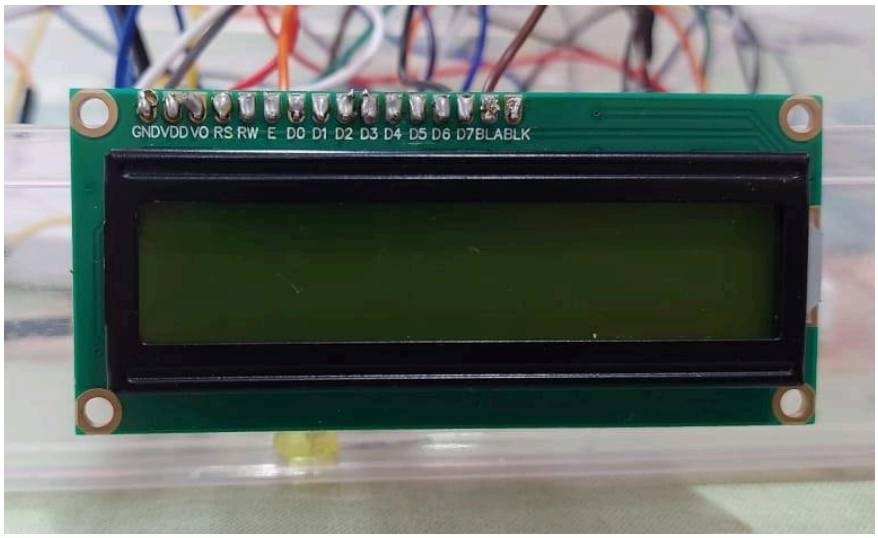
3.6 BREADBOARD

A breadboard is a rectangular plastic board with a grid of tiny holes. It's a fundamental tool used in prototyping electronic circuits without the need for any soldering.

Here's how it works:

Holes and Connections: The tiny holes on the breadboard are connected internally in rows and columns. The middle section usually contains rows of connected holes, while the outer sections (often marked with red and blue or positive (+) and negative (-) symbols) are typically used for power supply lines. Component Insertion: Electronic components like resistors, capacitors, and integrated circuits (ICs) can be inserted into these holes. The leads of the components go into the holes and make contact with the internal connectors. Circuit Building: By inserting components and connecting them with jumper wires, you can create a complete electronic circuit on a breadboard. The layout can mimic the schematic diagram of the circuit, making it easier to build and troubleshoot. Flexibility: One of the key advantages of a breadboard is its flexibility. Components can be easily added, removed, or repositioned, allowing for rapid prototyping and testing.

3.7 LCD DISPLAY



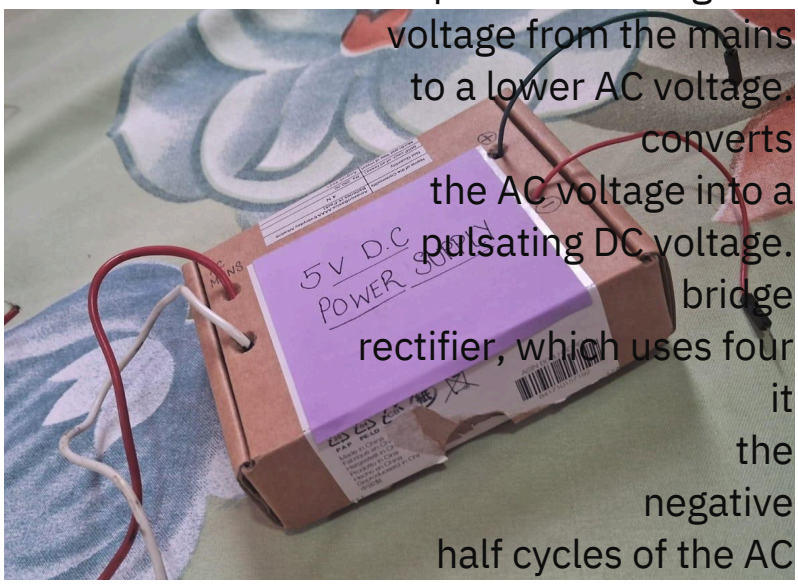
The 16x2 LCD display is a very versatile display module commonly used in various Arduino projects. The ‘16x2’ refers to the display’s ability to support 16 characters in 2 rows, providing ample space to display useful information. It uses the HD44780 controller, making it easy to interface with Arduino using the LiquidCrystal library.

This display can show alphanumeric characters and some special characters, making it suitable for a wide range of applications. It operates on 5V and uses a 16-pin interface for connections.

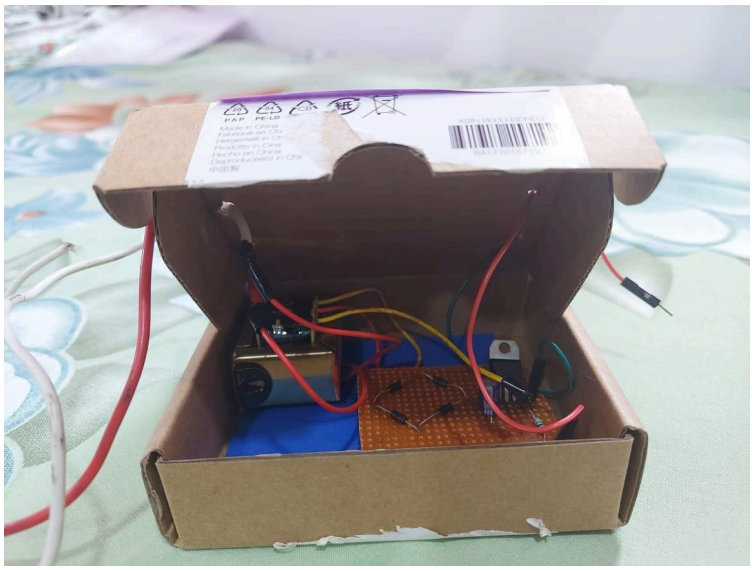
In an Arduino-based plant watering system, the 16x2 LCD could display real-time data such as soil moisture levels, system status, and other relevant information, enhancing the user interaction with the system.

3.8 5V DC POWER SUPPLY

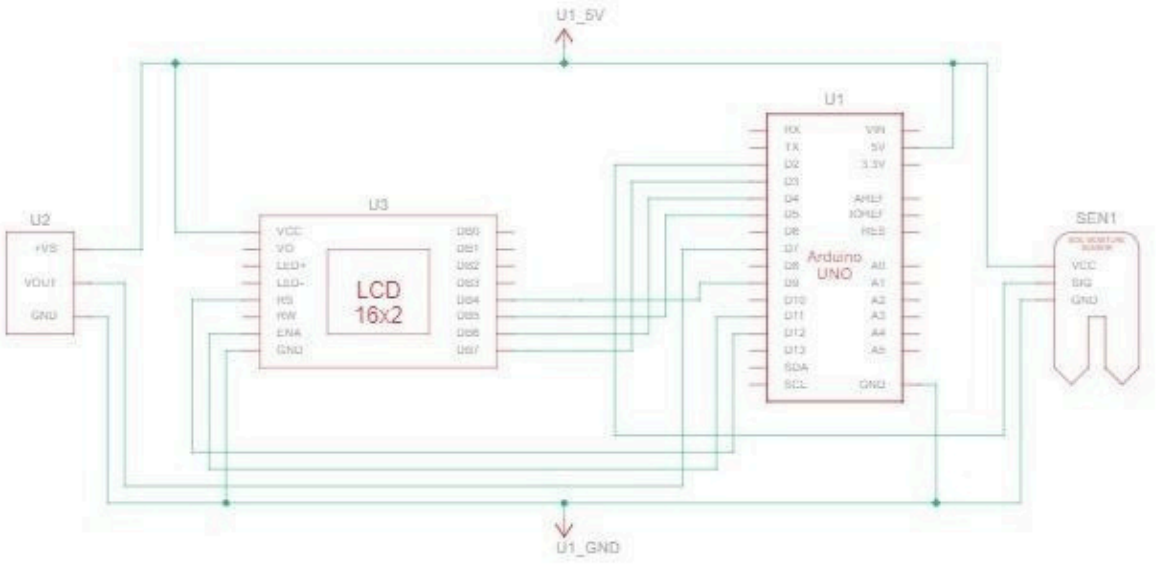
A DC power supply is a device that supplies electric energy of fixed polarity, either positive or negative. It can be powered from an AC or DC source. We made a DC power supply consisting of a transformer, a rectifier, a filter capacitor, and a voltage regulator. The transformer steps down the high AC voltage from the mains to a lower AC voltage.

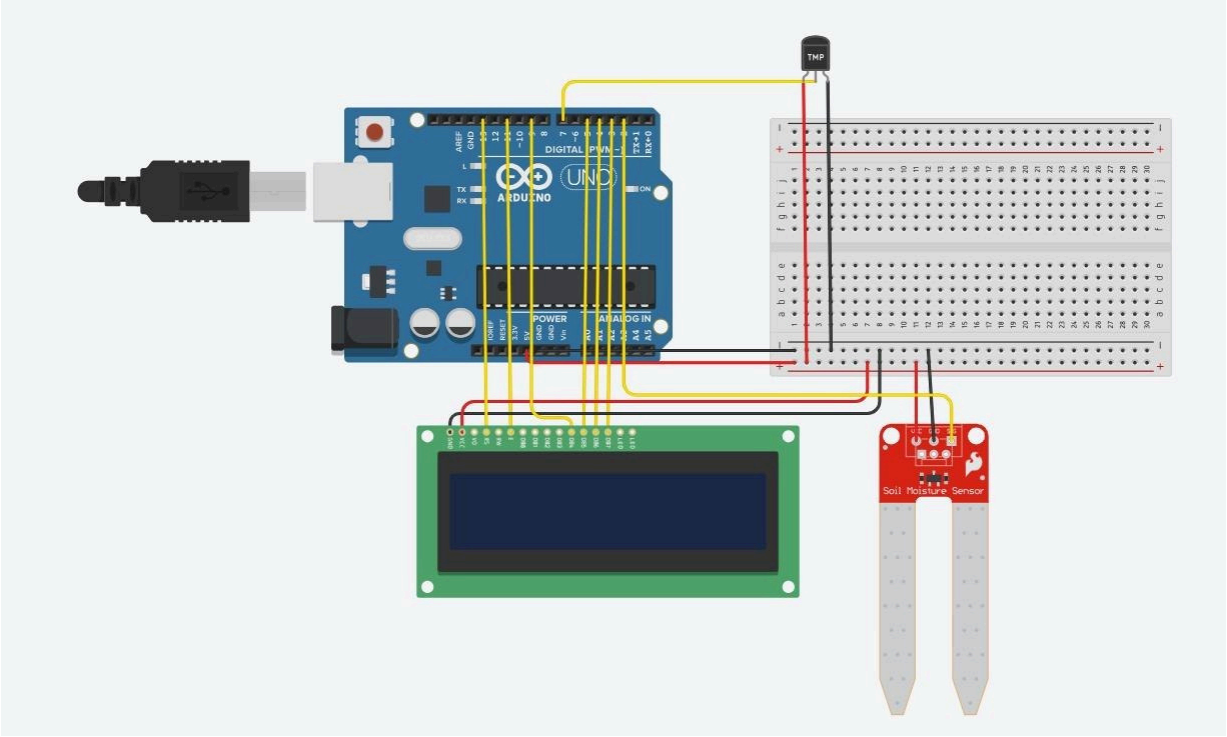


The rectifier converts the AC voltage into a pulsating DC voltage. We used a bridge rectifier, which uses four diodes because it converts both the positive and negative half cycles of the AC signal into positive voltage. The filter capacitor is used to smooth out the pulsating DC voltage by charging up when the voltage increases and discharging when the voltage decreases. This results in a smoother, steady DC voltage. The voltage regulator IC maintains a constant output voltage even when the input voltage or load current changes. It removes any remaining fluctuations in the voltage and provides a stable DC voltage.



CHAPTER 4
CIRCUIT DIAGRAM

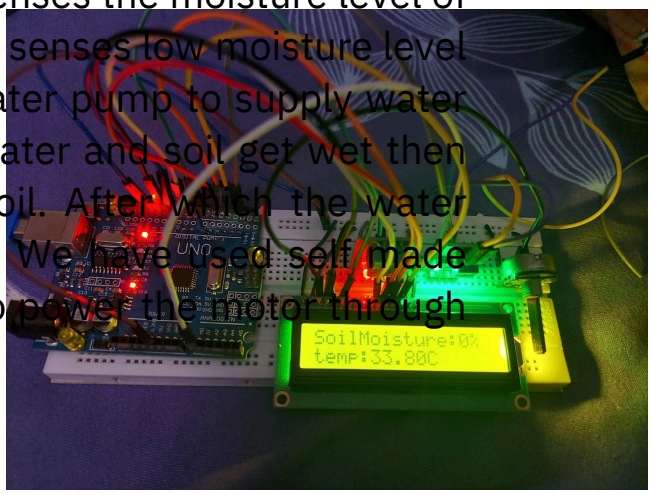




CHAPTER 5

WORKING OF AUTOMATIC PLANT WATERING SYSTEM

In this system, soil moisture sensor senses the moisture level of the soil. If soil will get dry then sensor senses low moisture level and automatically switches on the water pump to supply water to the plant. As plant get sufficient water and soil get wet then sensor senses enough moisture in soil. After which the water pump will automatically get stopped. We have used self made 5v power supply, which we will use to power the motor through the relay.



- **Power on**: The system is powered by a **9V** power supply connected to the Arduino board.
- **Sensor reading**: The system uses a **capacitive soil moisture sensor** connected to an **analog input** pin of the **Arduino**.

  ● **Moisture calculation**: The digital value from the sensor is used to determine whether the water supply should be on or off.
- **LCD display:** The temperature and humidity is displayed on a **16x2 LCD** module connected to the Arduino. **Relay control**:
- Based on the **reading from the moisture sensor**, the Arduino makes a decision whether to **turn on** or the moisture reading is LOW, **off** the irrigation system. If indicating a low moisture level, the Arduino sets the relay pin to

- **LOW**, turning on the relay connected to the pump. This activates the pump and starts the irrigation process.

● **Monitoring and loop**: The Arduino **continuously reads** the **moisture level**, updates the LCD display, and checks the **moisture reading** to determine the pump status.

CHAPTER 6
PROGRAM

```
#include <LiquidCrystal.h> // Library to operate LCD imported.

#include "DHT.h"// Library to operate imported DHT sensors.

#define DHTPIN 7    // Digital pin connected to the DHT sensor

#define DHTTYPE DHT11

#define relay 8 //relay connected to digital pin 8

// Sensor pins

#define sensorPower 2 // Soil moisture sensor connected to digital pin 2.

const int rs = 12, en = 11, d4 = 9, d5 = 5, d6 = 4, d7 = 3;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

DHT dht(DHTPIN, DHTTYPE);

void setup() {

   lcd.begin(16, 2);


pinMode(sensorPower, INPUT);

pinMode(relay, OUTPUT);

dht.begin();

 // Initially keep the sensor OFF

 digitalWrite(sensorPower, LOW);

 digitalWrite(relay, LOW);

}
```

```
void loop() {
 //get the reading from the function below and print it
 int readSensor= digitalRead(sensorPower);
 if(readSensor==LOW){
     digitalWrite(relay, HIGH);
  }
  else{
     digitalWrite(relay , LOW);
  }
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  // Check if any reads failed and exit early
  if (isnan(h) || isnan(t) ) {
     lcd.print("error in reading temp");
     return;}
  //printing values on lcd screen
  lcd.setCursor(0,0);
  lcd. print("temp:");
  lcd.print(t);
  lcd.print("C");
  lcd.setCursor(0,1);
  lcd.print("Humidity:");
  lcd.print(h);
  lcd.print("%");
  delay(1000);}
```

CHAPTER 7

EXPLANATIONOFPROGRAM

1. Header Inclusions: The code includes the “ LiquidCrystal.h”and "DHT.h" library, which is used to operate LCD display and DHT sensors.

1. MacroDefinitions:
   - DHTPIN is defined as the digital pin connected to the DHT sensor (pin 7 in this case).
   - DHTTYPE is defined as DHT11, indicating the type of DHT sensor being used.
   - relayisdefinedasthedigitalpinconnectedtoarelay(pin8in this case).
   - sensorPower is defined as the digital pin connected to a soil moisture sensor (pin 2 in this case).
   - PinassignmentsfortheLiquidCrystallibrary(rs,en,d4,d5,d6, d7) are also defined.
2. Object Instantiation:
   - An instance of the LiquidCrystal class is created with the pins specified earlier.
   - An instance of the DHT class is created, specifying the pin connected to the DHT sensor and its type.
3. Setup Function:
   - setup() function initialises pin modes for sensorPower (input) and relay (output).
   - DHTsensorisinitialised.
   - Initially, both sensorPower and relay pins are set to LOW (off state).
4. Loop Function:
   - loop()functioncontinuouslyexecutes.
   - ItreadsthestateofsensorPowerpinandifit'sLOW,itturnson the relay (presumably to activate some external device).
   - It then reads temperature and humidity data from the DHT sensor.
   - Ifthereadingsarevalid(notNaN),itdisplaysthemontheLCD screen.

- Thelooprepeatswithadelayof1second(delay(1000)).

5. LCDDisplay:
   - Temperature and humidity readings are displayed on a 16x2 character LCD.
   - The first line shows temperature in Celsius and the second line shows humidity percentage.

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

An automatic plant watering system using Arduino is designed in this project. The prototype of the model worked properly when tested on different soils. The components that we use in the system are readily available and easy to operate. Thus, this system acts as an effective method of irrigation. It is far better than the manual watering process which requires a lot manpower and time. By using the app, the farmer can operate the system from distant places. The farmer can utilise this time in other significant activities.

Also, the major issue of water scarcity is dealt with. No amount of water is wasted in the process of irrigation. Thus, this system can be very useful in areas where water is in short supply. As the required amount of water is provided to the crop, the crop growth is better. Farmers can thus benefit from the enhanced crop yields. The project is tested for different types of soils and it works properly. The future work of the system can include the addition of temperature sensors and a more powerful motor to pump water to the fields. Thus, the large-scale implementation of the project can also be done.

CHAPTER 9
REFERENCES

- https://projecthub.arduino.cc/Saikan45/automated-watering-plant-fe5d5c
- https://www.instructables.com/Arduino-Plant-Watering-System/
- https://www.eleccircuit.com/lm317-power-supply/
- https://docs.arduino.cc/learn/electronics/lcd-displays
- https://projecthub.arduino.cc/arcaegecengiz/using-dht11-12f621
- https://www.arduino.cc/reference/en/libraries/dht11/
- https://www.electronicwings.com/arduino/soil-moisture-sensor-interfacing-with-arduino-uno
- https://projecthub.arduino.cc/nikolaiapalis/simple-soil-moisture-sensor-ec23c7