

# Automated Plant Watering System Documentation

## Overview:

The **Automated Plant Watering System** is a smart solution designed to automatically water plants when the soil becomes dry. It integrates various sensors, including a **DHT11** for measuring temperature and humidity, a **Soil Moisture Sensor** for determining soil dryness, and a **Rainwater Sensor** to manage rainwater collection. The system uses a **Relay** to control a water pump, which waters the plant, and an **LCD Display** to show real-time data. The system also includes a **Servo Motor** for rainwater collection when rain is detected.

---

## Hardware Components:

1. **DHT11 Sensor:**
    - Measures temperature and humidity.
    - Outputs data to the Arduino.
  2. **Soil Moisture Sensor:**
    - Detects the moisture level in the soil.
    - Sends an analog signal to the Arduino indicating moisture level.
  3. **Relay Module:**
    - Acts as a switch to control the water pump.
    - Turns the water pump on or off depending on soil moisture readings.
  4. **Water Pump:**
    - Water pump activated by the relay to water the plant.
  5. **LCD Display (16x2 with I2C):**
    - Displays real-time data such as temperature, humidity, soil moisture, and system status.
  6. **Servo Motor:**
    - Controls a rainwater collection system when rain is detected.
  7. **Buzzer:**
    - Alerts the user when the soil is dry and the pump is activated.
  8. **Jumper Wires:**
    - Connects the components to the Arduino.
  9. **Breadboard (optional):**
    - For easier connections and prototyping.
  10. **Power Supply:**
    - Provides power to the Arduino and other components.
- 

## Hardware Connections:

1. **Arduino Uno R3:**
  - Pin 2: Connect to DHT11 Data Pin.
  - Pin A0: Connect to Soil Moisture Sensor analog output.

- Pin 8: Connect to Relay Module IN Pin (for controlling the pump).
  - 2. **DHT11 Sensor:**
    - VCC Pin: Connect to 5V on the Arduino.
    - GND Pin: Connect to GND on the Arduino.
    - Data Pin: Connect to Pin 2 on the Arduino.
  - 3. **Soil Moisture Sensor:**
    - VCC Pin: Connect to 5V on the Arduino.
    - GND Pin: Connect to GND on the Arduino.
    - Analog Pin (A0): Connect to Pin A0 on the Arduino for soil moisture readings.
  - 4. **Relay Module:**
    - VCC Pin: Connect to 5V on the Arduino.
    - GND Pin: Connect to GND on the Arduino.
    - IN Pin: Connect to Pin 8 on the Arduino (for controlling relay).
    - Common Pin (COM): Connect to the positive terminal of the water pump.
    - Normally Open (NO) Pin: Connect to the positive terminal of the power supply for the water pump.
  - 5. **Water Pump:**
    - Positive terminal: Connect to the NO Pin of the relay.
    - Negative terminal: Connect to GND of the power supply.
  - 6. **LCD Display (16x2 with I2C):**
    - VCC Pin: Connect to 5V on the Arduino.
    - GND Pin: Connect to GND on the Arduino.
    - SDA Pin: Connect to A4 on the Arduino.
    - SCL Pin: Connect to A5 on the Arduino.
  - 7. **Servo Motor:**
    - Orange (Signal) to Pin 6 on the Arduino.
    - Red (Power) to 5V.
    - Black/Brown (Ground) to GND.
  - 8. **Buzzer:**
    - Positive terminal to Pin 9 on the Arduino.
    - Negative terminal to GND.
  - 9. **Power Supply:**
    - Use a 5V or 9V adapter or USB cable to power the Arduino.
    - The water pump can be powered using an external power supply (e.g., 12V or 9V) depending on the pump specifications.
- 

## Software Requirements:

To run the Automated Plant Watering System, you will need the following libraries installed in the Arduino IDE:

1. **DHT Sensor Library:** To read data from the DHT11 sensor.
2. **LiquidCrystal\_I2C Library:** For controlling the LCD display using the I2C protocol.

You can install these libraries via the Arduino IDE by navigating to **Sketch > Include Library > Manage Libraries**, and then searching for and installing **DHT sensor library** and **LiquidCrystal\_I2C**.

---

## Arduino Code:

The following Arduino code integrates all the hardware components into one working system. The system reads data from the DHT11 sensor, checks soil moisture levels, and controls the water pump via the relay based on the soil's dryness. It also detects rain using the rain sensor and activates the rainwater collection system.

cpp

Copy code

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#include <Servo.h>

// LCD Settings
LiquidCrystal_I2C lcd(0x27, 16, 2); // I2C address may need to be
adjusted

// DHT11 Sensor Settings
#define DHTPIN 7 // Pin for DHT11 sensor
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE); // Initialize DHT sensor

// Soil Moisture Sensor Settings
#define SOIL_PIN A0 // Analog pin for Soil Moisture
sensor
#define MOISTURE_THRESHOLD 600 // Threshold for soil moisture
(Adjust if needed)

// Rain Sensor Settings
#define RAIN_PIN A1 // Analog pin for Rain Sensor
#define RAIN_THRESHOLD 500 // Threshold for rain detection

// Buzzer Settings
#define BUZZER_PIN 9 // Digital pin for Buzzer

// Relay and Pump Settings
```

```

#define RELAY_PIN 8                // Relay to control the water
pump

// Servo Motor Settings
Servo rainServo;                  // Initialize servo motor
#define SERVO_PIN 6                // Digital pin for Servo Motor

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

    // LCD Initialization
    lcd.begin(16, 2);
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("System Starting");
    delay(2000);
    lcd.clear();

    // DHT Initialization
    dht.begin();

    // Pin Initialization
    pinMode(SOIL_PIN, INPUT);
    pinMode(RAIN_PIN, INPUT);
    pinMode(BUZZER_PIN, OUTPUT);
    pinMode(RELAY_PIN, OUTPUT);
    digitalWrite(RELAY_PIN, LOW);    // Initially turn off the pump
    rainServo.attach(SERVO_PIN);
    rainServo.write(90);             // Start with servo in closed
position
}

void loop() {
    // Read Temperature and Humidity
    float temperature = dht.readTemperature();
    float humidity = dht.readHumidity();

    // Display Temperature and Humidity on LCD
    lcd.setCursor(0, 0);
    lcd.print("Temp: ");
    lcd.print(temperature);
}

```

```
lcd.print("C");

lcd.setCursor(0, 1);
lcd.print("Humidity: ");
lcd.print(humidity);
lcd.print("%");

delay(2000); // Update every 2 seconds

// Soil Moisture Reading
int soilMoistureValue = analogRead(SOIL_PIN);
Serial.print("Soil Moisture: ");
Serial.println(soilMoistureValue);

// Rain Sensor Reading
int rainValue = analogRead(RAIN_PIN);
Serial.print("Rain Sensor: ");
Serial.println(rainValue);

// Display Soil Moisture Value
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Soil Moisture:");
lcd.setCursor(0, 1);
lcd.print(soilMoistureValue);
delay(2000);

// Check Rain Sensor for Collection
if (rainValue < RAIN_THRESHOLD) {
    // Open servo to collect rainwater
    rainServo.write(0); // Adjust angle as per your setup
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Rain Detected");
    lcd.setCursor(0, 1);
    lcd.print("Collecting Water");
    delay(2000);
} else {
    // Close the collection point
    rainServo.write(90); // Adjust angle as per your setup
    lcd.clear();
}
```

```

        lcd.setCursor(0, 0);
        lcd.print("No Rain");
        lcd.setCursor(0, 1);
        lcd.print("Closed");
        delay(2000);
    }

    // Soil Moisture Check for Pump and Buzzer
    if (soilMoistureValue > MOISTURE_THRESHOLD) {
        digitalWrite(RELAY_PIN, HIGH); // Turn on the pump
        digitalWrite(BUZZER_PIN, HIGH); // Sound the buzzer
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Dry Soil");
        lcd.setCursor(0, 1);
        lcd.print("Pump ON");
        delay(2000);
    } else {
        digitalWrite(RELAY_PIN, LOW); // Turn off the pump
        digitalWrite(BUZZER_PIN, LOW); // Turn off the buzzer
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Soil Moisture OK");
        delay(2000);
    }
}

```

---

## System Functionality:

1. **Soil Moisture Detection:**
  - The soil moisture sensor checks if the soil is dry. If the moisture level is below the threshold (600), the water pump is turned on, and the buzzer sounds. If the soil is moist, the pump stays off.
2. **Temperature and Humidity:**
  - The DHT11 sensor reads the temperature and humidity of the surrounding environment and displays it on the LCD screen.
3. **Rainwater Collection:**
  - The rain sensor checks whether rain is detected. If rain is detected, the servo motor opens to collect rainwater. If no rain is detected, the servo closes the collection point.
4. **Alerts:**

- The buzzer alerts when the soil is dry, indicating that the system is watering the plant.
- 

### **Future Enhancements:**

1. **Water Level Sensor:**
    - Add a water level sensor to ensure the water pump does not run dry.
  2. **Automatic Adjustments:**
    - Implement automatic adjustments for the watering frequency based on environmental conditions, such as temperature and humidity.
  3. **Cloud Connectivity:**
    - Add cloud connectivity to monitor the system remotely via a web app.
- 

This is a working system for automating the watering of plants based on environmental conditions and soil moisture, with additional capabilities for rainwater collection and monitoring.