

REFERENCES

- [1] A. R. Al-Ali, M. Qasaimeh, M. Al-Mardini, and S. Radder, “ZigBee-based irrigation system for smart agriculture,” *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 10, pp. 1–8, 2015.
- [2] S. R. Nandurkar, V. R. Thool, and R. C. Thool, “Design and development of precision agriculture system using wireless sensor network,” *IEEE International Conference on Automation, Control, Energy and Systems*, pp. 1–6, 2014.
- [3] P. Rawal, “IoT based smart irrigation system,” *International Journal of Computer Applications*, vol. 159, no. 8, pp. 7–11, 2017.
- [4] M. A. Khan, M. S. Islam, and M. S. Rahman, “Automated irrigation system using microcontroller and soil moisture sensor,” *International Journal of Engineering Research and Technology*, vol. 4, no. 5, pp. 1–4, 2015.
- [5] R. Evans and E. Sadler, “Methods and technologies to improve efficiency of water use,” *Water Resources Research*, vol. 44, no. 7, pp. 1–15, 2008.
- [6] Arduino, “Arduino Uno Rev3 – Technical Specifications,” Arduino Documentation. [Online]. Available: <https://www.arduino.cc>. Accessed: 2025.
- [7] Labcenter Electronics, “Proteus Design Suite User Guide,” Labcenter Electronics Ltd., 2024.
- [8] Autodesk, “Tinkercad Circuits: Simulation and Prototyping,” Autodesk Documentation, 2024.

APPENDICES

Appendix A:

MICROCONTROLLER CODE (ARDUINO)

```
// Smart Irrigation System
// Author: [Your Name]
// Date: 2025
// Description: Reads soil moisture sensor and controls water pump in real-time

// Pin Definitions
const int soilMoisturePin = A0; // Analog pin connected to soil moisture sensor
const int relayPin = 7;         // Digital pin connected to relay module
const int ledPin = 13;          // Optional LED to indicate pump status

// Moisture Threshold
const int moistureThreshold = 500; // Adjust based on calibration (0-1023 scale)

void setup() {
    // Initialize serial monitor for debugging
    Serial.begin(9600);

    // Initialize pins
    pinMode(relayPin, OUTPUT);
    pinMode(ledPin, OUTPUT);

    // Ensure pump is off at startup
    digitalWrite(relayPin, LOW);
    digitalWrite(ledPin, LOW);
}

void loop() {
    // Read soil moisture value
    int sensorValue = analogRead(soilMoisturePin);

    // Print sensor value for debugging
    Serial.print("Soil Moisture Reading: ");
    Serial.println(sensorValue);

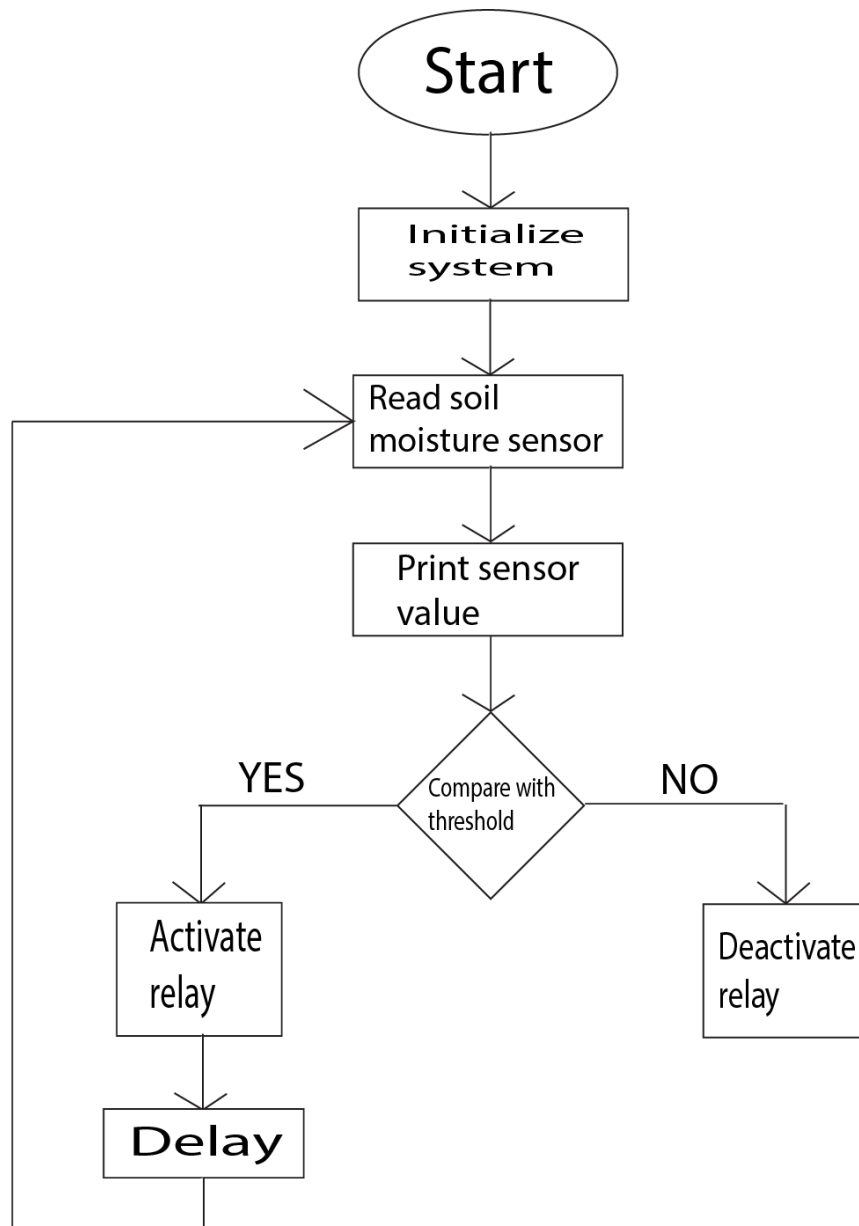
    // Decision making: Activate or deactivate pump
    if(sensorValue < moistureThreshold) {
        // Soil is dry: turn ON pump
        digitalWrite(relayPin, HIGH);
        digitalWrite(ledPin, HIGH); // Optional: LED ON
        Serial.println("Pump ON");
    } else {
        // Soil is wet: turn OFF pump
        digitalWrite(relayPin, LOW);
        digitalWrite(ledPin, LOW); // Optional: LED OFF
        Serial.println("Pump OFF");
    }

    // Wait before next reading
    delay(2000); // 2 seconds, adjust as needed
}
```

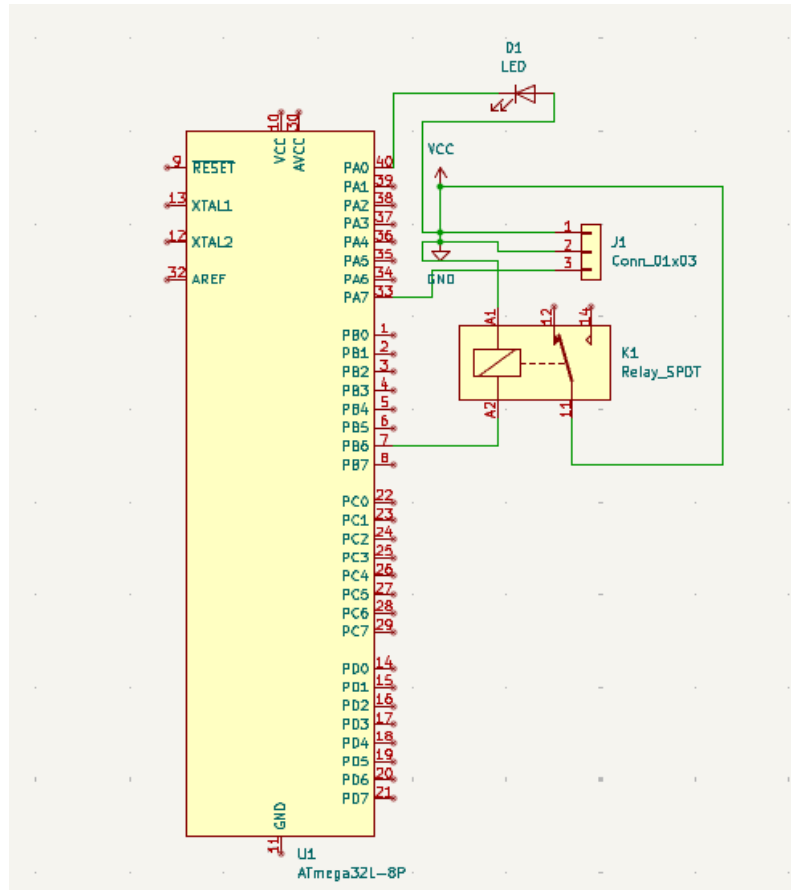
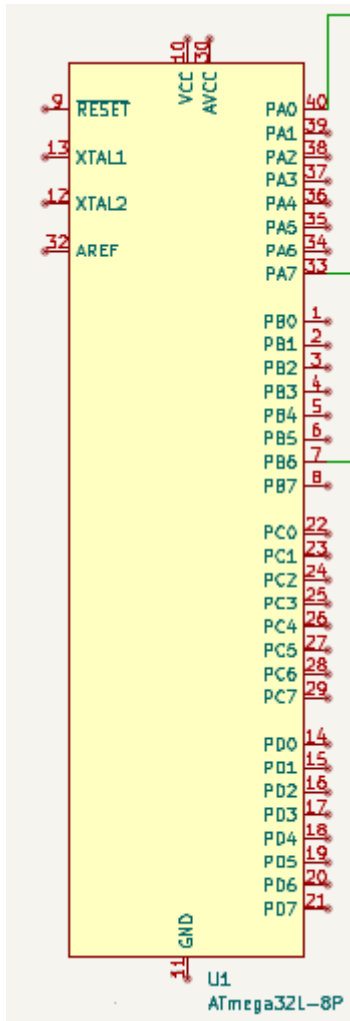
FLOWCHART CODE (RELAY)

```
/******  
* Smart Irrigation System  
* Author: [Your Name]  
* Date: 2025  
* Description: This program reads a soil moisture sensor  
* and controls a water pump in real-time using a relay.  
* The system automatically irrigates when soil moisture  
* falls below a threshold and stops when adequate moisture  
* is detected.  
*****/  
  
// Pin Definitions  
const int soilMoisturePin = A0; // Analog input from soil moisture sensor  
const int relayPin = 7;        // Digital output to relay module  
const int ledPin = 13;         // Optional LED to indicate pump status  
  
// Moisture Threshold (0-1023)  
const int moistureThreshold = 500; // Adjust after calibration  
  
void setup() {  
    // Initialize serial communication for debugging  
    Serial.begin(9600);  
  
    // Initialize pins  
    pinMode(relayPin, OUTPUT);  
    pinMode(ledPin, OUTPUT);  
  
    // Ensure pump is OFF at startup  
    digitalWrite(relayPin, LOW);  
    digitalWrite(ledPin, LOW);  
  
    Serial.println("Smart Irrigation System Initialized");  
}  
  
void loop() {  
    // Read the soil moisture sensor value  
    int sensorValue = analogRead(soilMoisturePin);  
  
    // Print sensor reading for monitoring  
    Serial.print("Soil Moisture Reading: ");  
    Serial.println(sensorValue);  
  
    // Compare sensor value to threshold  
    if(sensorValue < moistureThreshold) {  
        // Soil is dry: activate pump  
        digitalWrite(relayPin, HIGH);  
        digitalWrite(ledPin, HIGH); // Optional LED ON  
        Serial.println("Pump ON - Soil is dry");  
    } else {  
        // Soil is wet: deactivate pump  
        digitalWrite(relayPin, LOW);  
        digitalWrite(ledPin, LOW); // Optional LED OFF  
        Serial.println("Pump OFF - Soil moisture adequate");  
    }  
  
    // Wait for 2 seconds before next reading  
    delay(2000);  
}
```

FLOWCHART

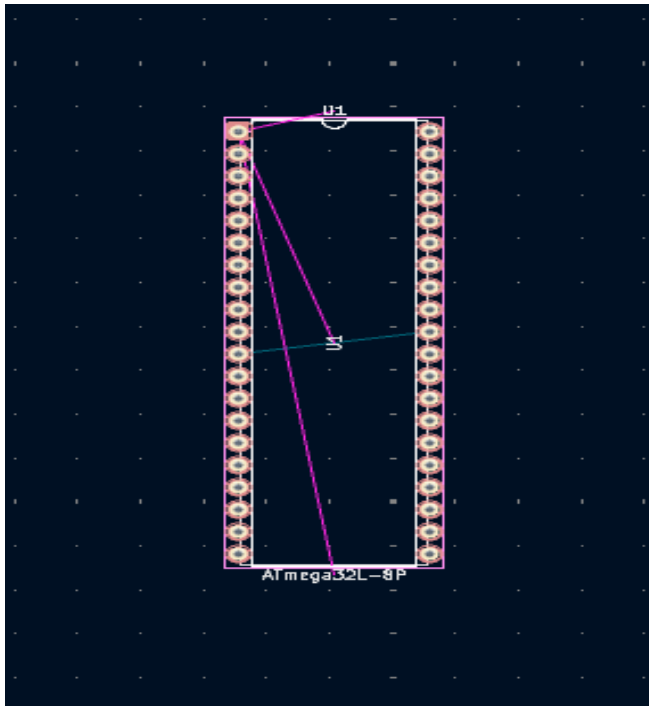


Appendix B:



The image on the left shows the Microcontroller, ATmega32L-8P which AV 8 pin form.

Appendix C:



The screenshot above shows the PCB layout of the smart irrigation system.