

SMART AGRICULTURE MONITORING

SYSTEM

V.R.S COLLEGE OF ENGINEERING AND

TECHNOLOGY

A.AARTHI(422722106001)

R.ARCHANA(422722106009)

M.MADHU MITHA(42722106034)

A.PRIYA DHARSHINI(422722106044)

A.SUJITHA(422722106052)

**Abstract**

A smart agriculture monitoring system leverages Internet of Things (IoT) technology to enhance farming practices. By integrating various sensors and devices, it provides real-time data on environmental conditions, crop health, and irrigation needs. This enables farmers to make informed decisions and optimize yield.

**Introduction**

Smart farming aims to improve agricultural efficiency, reduce resource wastage, and enhance crop quality. Key components of such a system include IoT sensors, cloud-based data storage, and automated control mechanisms.

**Methodology**

1. **Sensors**: The system employs different types of sensors:
   * **Soil Moisture Sensor**: Measures soil moisture levels to determine irrigation requirements.
   * **DHT11 Sensor**: Monitors temperature and humidity.
   * **DS18B20 Waterproof Temperature Sensor Probe**: Measures soil temperature.
   * **LDR (Light Dependent Resistor)**: Detects ambient light conditions.
2. **Automation**:
   * When soil moisture falls below a certain threshold, the system activates a submersible water pump to irrigate the crops.
   * The 12V LED strip provides additional lighting as needed.
3. **Data Transmission**:
   * The system sends sensor data (temperature, humidity, soil moisture, and soil temperature) to an IoT cloud platform for live monitoring.
   * Farmers can access this data remotely via web or mobile applications.

**Existing Work**

* Various smart agriculture systems have been developed worldwide, each tailored to specific crops, climates, and farm sizes

**Proposed Work**

* Our system extends existing approaches by integrating multiple sensors and providing real-time alerts.
* We prioritize efficient water usage and crop health, aiming to increase overall yield and product quality.

**System Requirements**

**Hardware Components**

1. NodeMCU ESP8266 (Microcontroller)
2. Soil Moisture Sensor
3. DHT11 Sensor (Temperature and Humidity)
4. DS18B20 Waterproof Temperature Sensor Probe
5. LDR (Light Dependent Resistor)
6. Submersible Mini Water Pump
7. 12V LED Strip
8. 7805 Voltage Regulator
9. Transistors (TIP122)
10. Resistors and Capacitors

**Software Components**

* Adafruit IO (Cloud-based data platform for aggregation and visualization)

**Implementation Details**

* Connect the sensors to the NodeMCU ESP8266 as per the circuit diagram.
* Set up an Adafruit IO account and configure it to receive sensor data.
* Deploy the system in the field, ensuring proper placement of sensors.
* Monitor data remotely via the Adafruit IO dashboard.

**Data Collection and Processing**

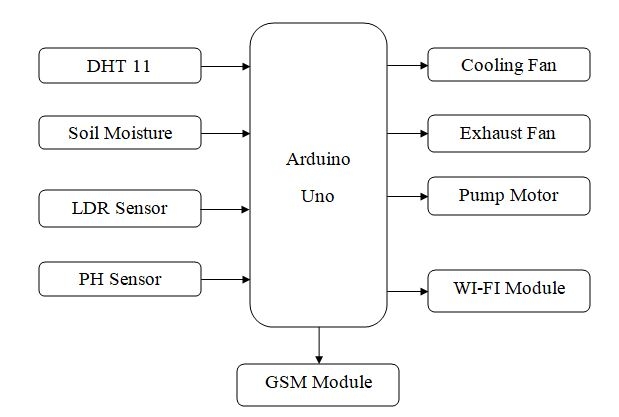
* The system continuously collects data from sensors.
* Algorithms process this data to trigger actions (e.g., activating the water pump).
* Historical data helps analyze trends and optimize farming practices.

**Communication Protocols**

* MQTT (Message Queuing Telemetry Transport) for data transmission between sensors and the cloud.
* HTTP/HTTPS for web and mobile app communication.

Remember that smart agriculture systems can be customized based on specific crop types, local conditions, and farmer preference

CIRCUIT DIAGRAM: [1](https://www.ibm.com/topics/smart-farming).



PROGRAM:

# Project objectives:

#   Read temperature and humidity values from the DHT22 sensor

#   Display the sensor readings in the console

#

# Hardware connections used:

#   DHT22 VCC Pin to 3.3V

#   DHT22 SDA Pin to GPIO Pin 15

#   10k ohm pull-up resistor from DHT22 SDA Pin to 3.3V

#   DHT22 GND Pin to GND

#

# Programmer: Adrian Josele G. Quional

# modules

from machine import Pin

from time import sleep

from dht import DHT22   # if the sensor is DHT11, import DHT11 instead of DHT22

# creating a DHT object

# change DHT22 to DHT11 if DHT11 is used

dht = DHT22(Pin(15))

# continuously get sensor readings while the board has power

while True:

    # getting sensor readings

    dht.measure()

    temp = dht.temperature()

    hum = dht.humidity()

    # displaying values to the console

    print(f"Temperature: {temp}°C   Humidity: {hum}% ")

    # format method or string concatenation may also be used

    #print("Temperature: {}°C   Humidity: {:.0f}% ".format(temp, hum))

    #print("Temperature: " + str(temp) + "°C" + "   Humidity: " + str(hum) + "%")

    # delay of 2 secs because DHT22 takes a reading once every 2 secs

    sleep(2)

CONCLUSION:

Certainly! Here’s a concise conclusion for your smart agriculture monitoring system:

The implementation of a smart agriculture monitoring system offers several benefits, including improved crop yield, resource efficiency, and reduced environmental impact. By integrating sensors, data analytics, and automation, farmers can make informed decisions about irrigation, fertilization, and pest control. Additionally, real-time monitoring allows for early detection of issues, minimizing crop loss.

Remember that successful deployment depends on factors such as sensor placement, data accuracy, and connectivity. Regular maintenance and updates are crucial to ensure optimal performance. Overall, smart agriculture systems contribute to sustainable farming practices and enhance food security.