INTRODUCTION

The implementation of smart farming systems has emerged as a promising solution to address the challenges faced by modern agriculture, such as resource management, environmental sustainability, and increasing productivity. In this project, a smart farming system is designed and developed to monitor and control key environmental parameters crucial for crop growth, including temperature, rainfall, soil moisture, and humidity. By leveraging sensor technologies, Arduino microcontroller, and an LCD display, the system offers real-time monitoring and automated control of essential farming operations. This report presents the design, implementation, and evaluation of the smart farming system, highlighting its potential to revolutionize agricultural practices and promote sustainable farming methods.

SYSTEM ARCHITECTURE

1. Components Used:

CELL - GSM Cell Phone Module
LM35 - LM35 Temperature Sensor
LM044L - Character LCD (LM044L)
LOGICSTATE - Logic State Probe
MOTOR - DC Motor
POT-HG - Potentiometer
RELAY - Relay
SIMULINO UNO - Simulino Uno (Arduino Uno Simulator)

2. Work Flow:

1. *Initialization:*

Set up the LCD display to show system status.
Define pins for fan and pump control.

2. *Main Loop:*

- Temperature Sensing:

Read temperature from the LM35 sensor.
Convert the sensor reading to Celsius.
Display temperature on the LCD.

- Rain Sensing:

Read the state of the rain sensor.
Display rain status on the LCD.

- Soil Moisture Sensing:

Read soil moisture level from the sensor.
Display soil moisture percentage on the LCD.

- Air Humidity Sensing:

Read humidity level from the sensor.
Display humidity percentage on the LCD.

- Fan Control:

If temperature exceeds 30°C:
Turn on the fan to cool down, Display fan status on the LCD as "Fan: ON".
Otherwise:
Turn off the fan, Display fan status on the LCD as "Fan: OFF".

- Pump Control:

If soil moisture falls below 40% and it's not raining:

Turn on the pump for watering.
Display pump status on the LCD as "Motor: ON".
Otherwise:
Turn off the pump, Display pump status on the LCD as "Motor: OFF".

3. Repeat:

Loop back to the beginning and continue monitoring and controlling the system based on sensor readings.

3. Elevator Architecture:

1) **Sensors**:

Temperature Sensor (LM35): Measures temperature and sends analog readings to the Arduino.
Rain Sensor: Detects rain and sends digital signals to the Arduino.
Soil Moisture Sensor: Measures soil moisture level and sends analog readings to the Arduino.
Humidity Sensor: Measures air humidity and sends analog readings to the Arduino.

2) Arduino (Simulino Uno):

Receives sensor data from all connected sensors.
Processes the data to determine appropriate control actions.
Controls the LCD display to show system status.
Controls the fan and pump based on predefined conditions.

3) Output Devices:

LCD (LM044L): Displays temperature, rain status, soil moisture, humidity, and status of fan and pump.
Fan: Controlled by the Arduino to turn on/off based on temperature.
Pump: Controlled by the Arduino to turn on/off based on soil moisture and rain status.

4) Relay:

Acts as a switch controlled by the Arduino to turn the fan and pump on/off.

5) Flow of Operation:

Sensors continuously monitor environmental conditions (temperature, rain, soil moisture, humidity, wind speed).
Sensor data is sent to the Arduino for processing.
Arduino analyzes the data and determines control actions.
Control signals are sent to the relay to activate/deactivate the fan and pump accordingly.
System status and control actions are displayed on the LCD for monitoring.

6) Interaction:

Sensors provide environmental data to the Arduino.
The Arduino processes sensor data and controls the relay based on predefined logic.
The relay switches the fan and pump on/off as instructed by the Arduino.
The LCDs real-time system status, including sensor readings and the operational state of the fan and pump.