TITLE

# SMART FARMING INNOVATION

## Automatic Crop Protection from Rain!

ajay pathipaka
ajaypathipaka456@gmail.com

## ABSTRACT

This project presents an **automatic crop protection system** designed to safeguard crops from unexpected rainfall using **Arduino-based automation**. A **rain sensor module** detects the presence of rain and activates a **DC motor with a horizontal sliding mechanism** to deploy a protective cover over the crops. When the rain stops, the system retracts the cover automatically, ensuring minimal human intervention. The system aims to reduce **crop damage due to excessive water exposure**, enhance **farm efficiency**, and support **smart agriculture practices**. The setup is cost-effective, easy to implement, and scalable for small farms, greenhouses, and home gardens.

## COMPONENTS:

### 1. Arduino UNO (Microcontroller)

- **Role:** Acts as the brain of the system.
- **Function:** Reads data from the rain sensor and controls the motor driver to move the cover.
- **usage:** It is easy to program, cost-effective, and widely used for automation and IoT projects.

---

### 2. Rain Sensor Module (e.g., FC-37)

- **Role:** Detects rainfall.
- **Function:** When raindrops fall on the sensor plate, its resistance changes, producing an analog signal. Arduino reads this value to decide whether to cover the crops.
- **Usage:** Simple and reliable way to detect the presence of rain.

---

### 3. DC Motor (or Gear Motor)

- **Role:** Moves the horizontal cover over the crops.
- **Function:** Rotates in **forward direction** to extend the cover and **reverse direction** to retract it.
- **Usage:** DC motors can move covers of larger sizes, making them suitable for horizontal sliding mechanisms.

---

### 4. L298N Motor Driver

- **Role:** Controls the DC motor.
- **Function:** Acts as an interface between the Arduino (low power) and the motor (high power). It allows **bidirectional control** (forward and reverse) of the motor.
- **Usage:** Arduino alone cannot supply enough current to drive a DC motor directly.

---

### 5. Power Supply

- **Role:** Provides necessary power for the motor and Arduino.
- **Function:** Typically, **12V** powers the DC motor through the L298N, and **5V** powers the Arduino and rain sensor.
- **Usage:** To ensure reliable operation of both logic circuits and motors.

---

### 6. Horizontal Cover Mechanism (Custom Frame)

- **Role:** The physical structure that slides to cover the crops.
- **Function:** Converts the motor's rotation into horizontal movement (using a pulley, gear, or rack-and-pinion setup).
- **Usage:** To provide physical protection to crops from rain.

### 7. (Optional) Limit Switches

- **Role:** Stop the motor when the cover is fully opened or closed.
- **Function:** When the cover reaches an endpoint, the switch is pressed, and the motor stops to avoid damage.
- **Usage:** To ensure safe and precise cover movement.

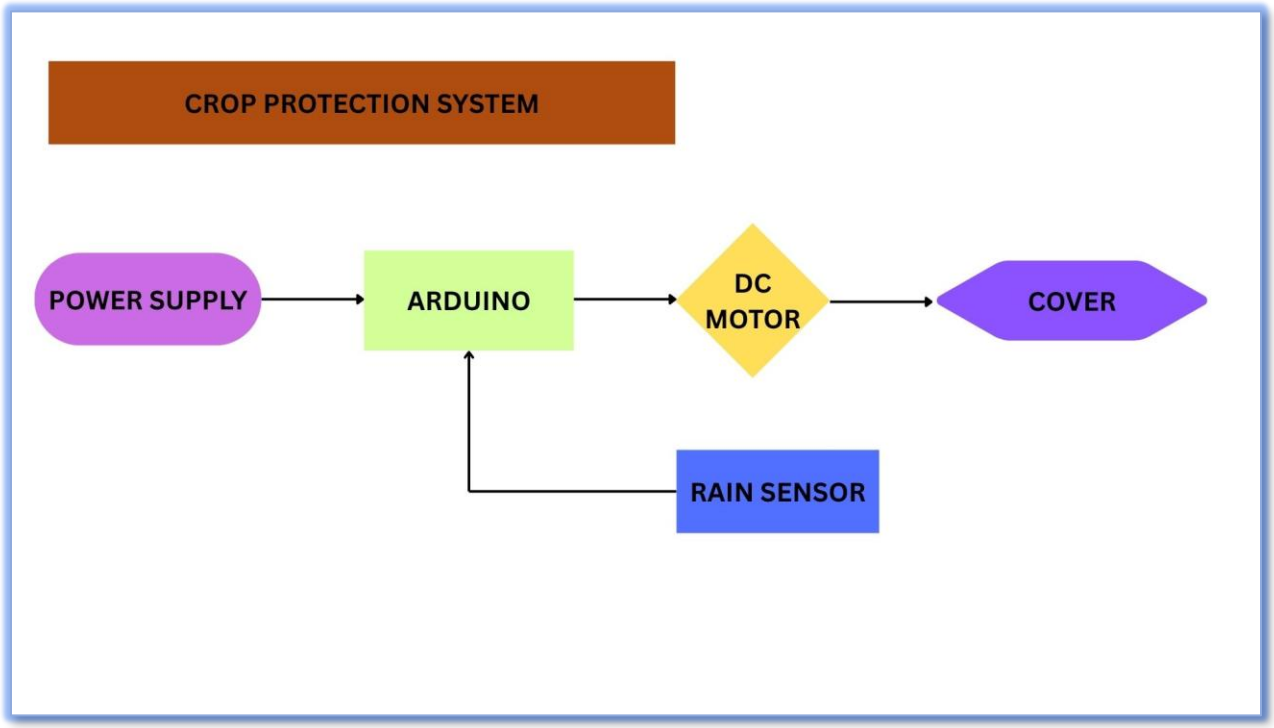
**CODE:**

```
// Pin definitions
int ENA = 9;
int IN1 = 8;
int IN2 = 7;
int rainSensorPin = A0;
int rainThreshold = 400;
int rainValue = 0;

void setup() {
  pinMode(ENA, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(rainSensorPin, INPUT);
  Serial.begin(9600);
  Serial.println("Horizontal Crop Cover System Started");
}
```

```
void loop()
 {
  rainValue = analogRead(rainSensorPin);
  Serial.print("Rain Sensor Value: ");
  Serial.println(rainValue);

  if (rainValue < rainThreshold)
{
    Serial.println("Rain detected! Closing cover...");
    moveCoverForward();
    delay(3000);
    stopMotor();
  }
 else
 {
    Serial.println("No rain. Opening cover...");
    moveCoverBackward();
    delay(3000);
    stopMotor();
  }
  delay(1000);
}
```

```
void moveCoverForward()
{
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  analogWrite(ENA, 200); // Speed (0-255)
}

void moveCoverBackward()
{
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  analogWrite(ENA, 200);
}

void stopMotor()
 {
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
}
```

## BLOCK DIAGRAM:




## CONCLUSION :

The **Automatic Crop Protection System** successfully demonstrates how **Arduino-based automation** can safeguard crops from unexpected rain using a **rain sensor and a horizontal cover mechanism**. The system minimizes manual intervention, reduces crop damage, and supports farmers by providing a cost-effective and reliable solution. This project showcases the potential of **IoT and embedded systems** in modern agriculture, enabling smarter and more efficient farming practices.