SOIL MOISTURE SENSOR- HARDWARE.

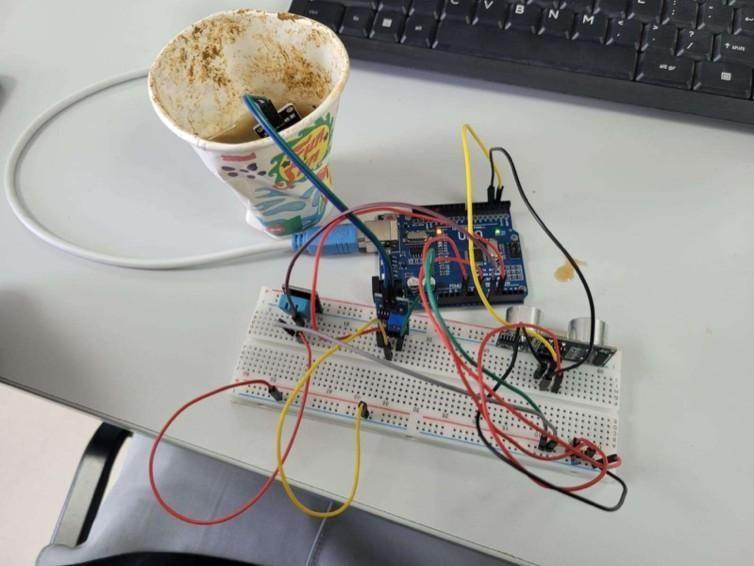
Name:- Sunkari Srinivas Akshit.

Aim: The aim of this project is to design a smart irrigation system that utilizes temperature, ultrasonic, and soil moisture sensors, coupled with an Arduino microcontroller, to optimize water usage, reduce water waste, and develop an alarm system for low water levels in the reservoir.

Components Required:

* + Arduino board (e.g., Arduino Uno)
  + Temperature sensor (e.g., DHT11)
  + Ultrasonic sensor (e.g., HC-SR04)
  + Soil moisture sensor (e.g., FC-28)
  + Resistor and capacitor for sensor connections
  + Jumper wires
  + Breadboard or PCB for circuit connections
  + Power supply (battery or AC adapter)
  + Water reservoir

Procedure:

* + 1. Set up the Arduino board and connect it to the computer.
    2. Connect The Temperature Sensor,ultrasonic sensor,and soil moisture sensor to the appropriate digital or analog pins of the Arduino.
    3. Establish the necessary power connections for the sensors and the Arduino board.
    4. ConfiguretheArduinosoftwaretoreadthesensorvalues.
    5. Implement a control algorithm to determine the irrigation requirements based on the sensor data.
    6. Connect the water pump to the Arduino using a relay module to control its activation.
    7. Display the sensor values and irrigation status on an LCD display.

NewPIng sona (tr,lgf'in,, eclloF'In,); // Crearte ai Newf'Ing object for ultrasonlc sensor

(Oi'i!St Int mols urePIn .. AO; *I.I* 5411molsture :sensor an,abg p.ln,

(On!St Int bu!ZteriPln,.. 4; *II*uner pin,

(On!St float waterThre:shold" 30.0; //Thresllold for low water level In the reservol r

void se,tupO (

Serial.be ln( 600}; *.1/* lnlt:lallcl:e serlaI oommunlca;t1on pinMode(buz.zr!rPlln, OUTPUT); *I/* Set buzter pIn as output

}

void loopO { lloat temp,eratur-e"' getTempr!rature(l; *II* Read tempr!ratul'e tMm a sl'!nisor float distance .. getDlstanc-.e(); // R:ead dlstanoe from

ultrasonic sensor lrlt moisture" getMolstureO; // Read soil mols ure level

*II* Ohed:: temp,erature If (temperature ;,.;;; 2S.0.1{

*fl* Perform Innlgat1on,proi::ess,

If (dls,tanoe > OJi && mols ul'e < 501f

st:aritlnnlgat1on(l;

} else {

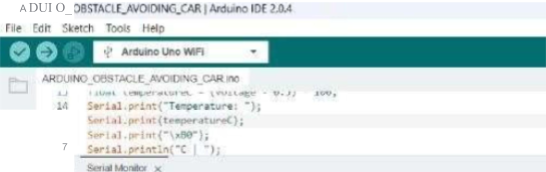
st:oplrrlgM:lon();

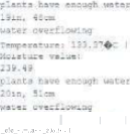
} el'!:e {

st:oplrrigalt:lon();

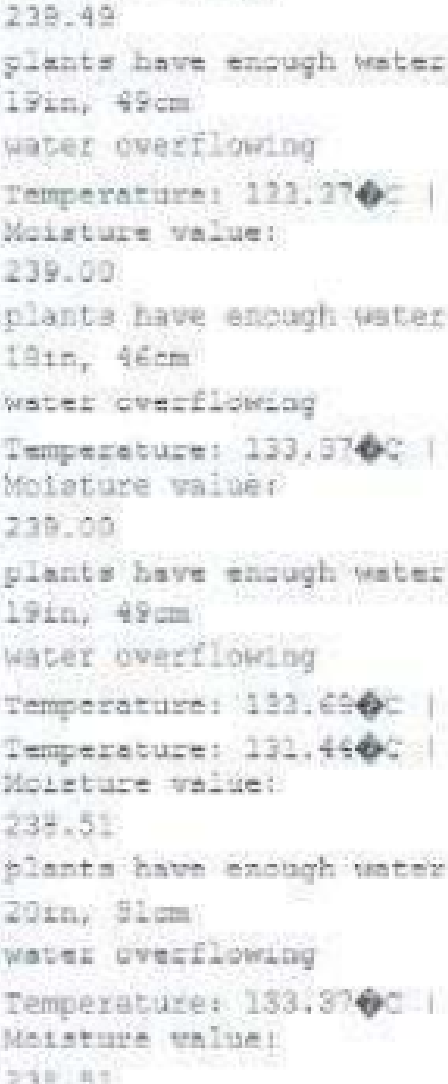


# Output:





*1* t t:. I ...11::-l





- - -

Q tlfC

# Result:

The smart irrigation system con□nuously monitors temperature, soil moisture, and distance to optimize water usage. When the soil moisture is below the desired level and the temperature is within the appropriate range, the system activates the water pump to irrigate the plants. The LCD display shows the sensor readings and irrigation status.

Additionally, an alarm is triggered when the water level in the reservoir becomes too low.

Conclusion:

The smart irrigation system with Arduino and sensor integration proved successful in optimizing water usage and reducing water waste during the irrigation process. By monitoring environmental conditions such as temperature, soil moisture, and water level, the system ensures that water is provided to the plants only when needed. The alarm system adds an additional layer of convenience by notifying the user when the water reservoir needs refilling.