Title of Mini project: Microcontroller based Plant Hydrator Using Weather Prediction and Soil Analysis.

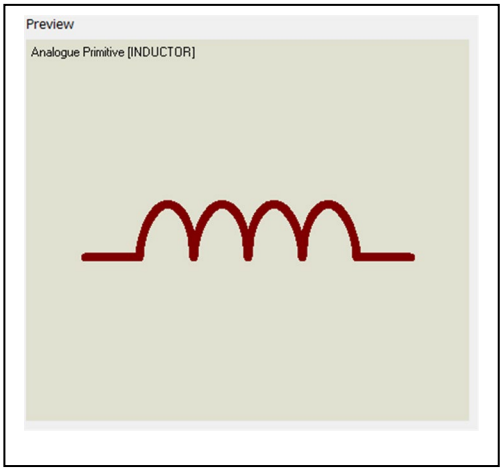
Objectives of Mini project:

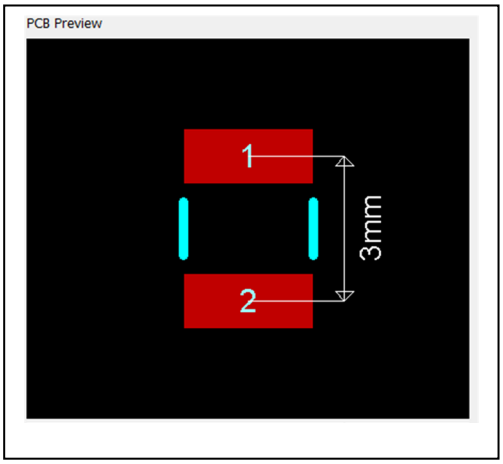
The objective of this project is to provide a combination of manual supervision and partial automation and is similar to manual set-up in most respects, but it reduces the labor involved in terms of Irrigation design is simple, easy to install, microcontroller-based circuit to monitor and record the values of temperature, soil moisture (Transistor circuit) that are continuously modified and controlled in order optimize them to achieve maximum plant growth and yield.

Also, the use of easily available components reduces the manufacturing and. The design is quite flexible as the software can be changed any time. It can thus be made to the specific requirements of the user. This makes the proposed system to be an economical, portable and a low maintenance solution for greenhouse applications, especially in rural areas and for small scale agriculturists.

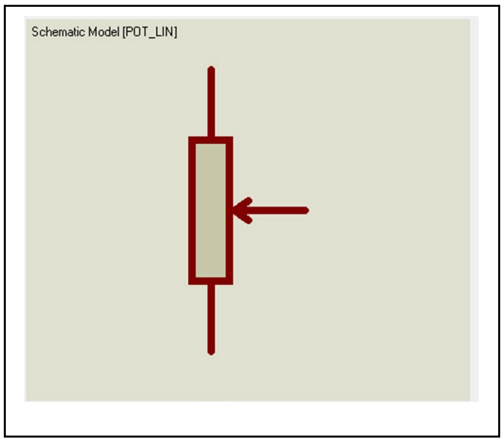
Tools required:

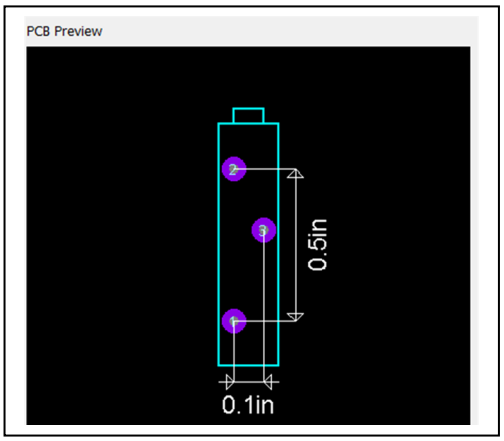
1. COIL .27uH IRON SMD



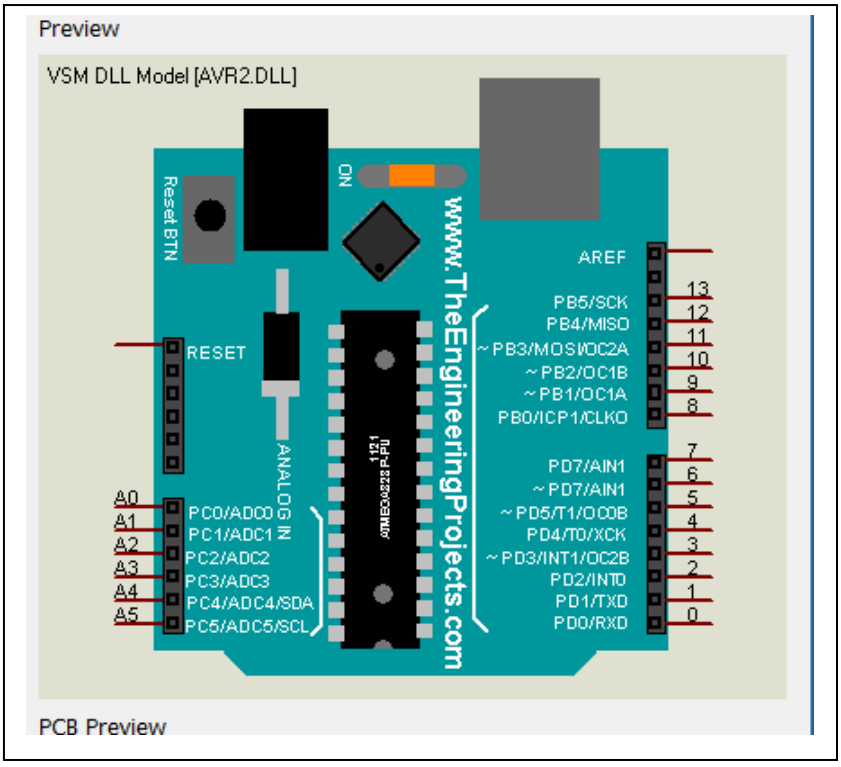


2. POT 10K OHM 3/4-inch RECT CERM MT

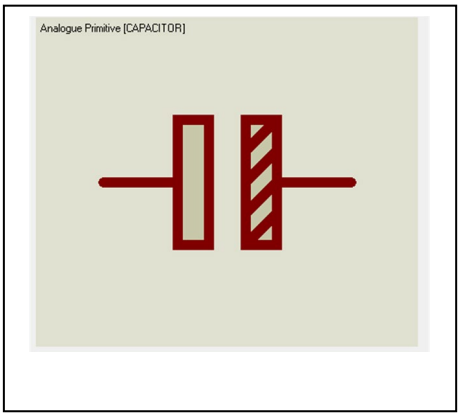


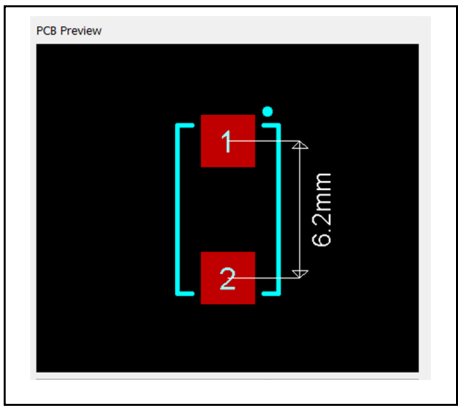


3. Arduino UNO R3

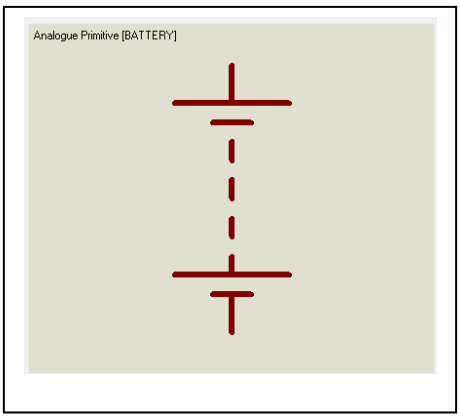


4. CAP TANTALUM 100uF 6.3V 10% SMD

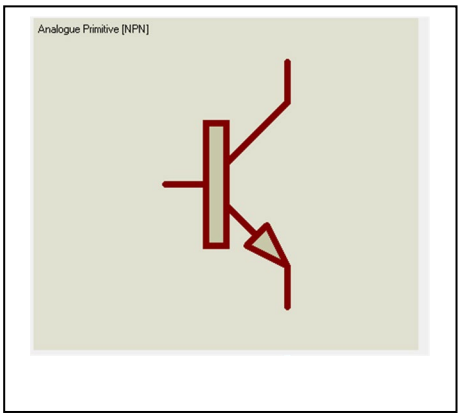


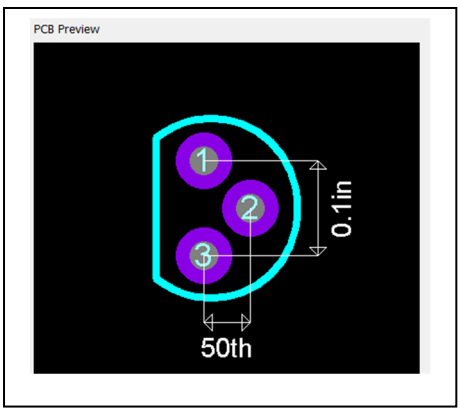


5. DC Voltage Source(battery) 12v

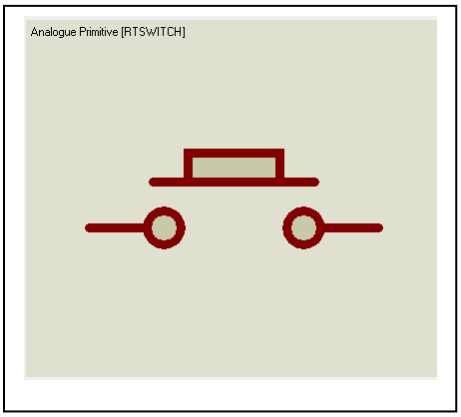


6. Silicon NPN Low Power Bipolar Transistor (625mW, 150ºC)

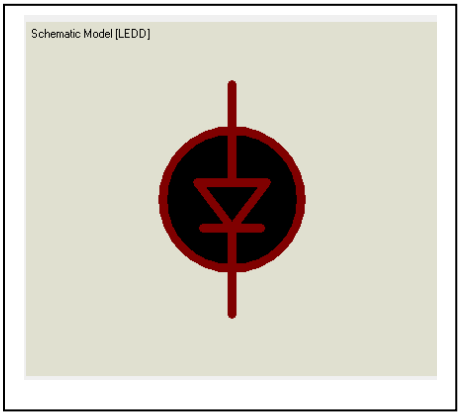




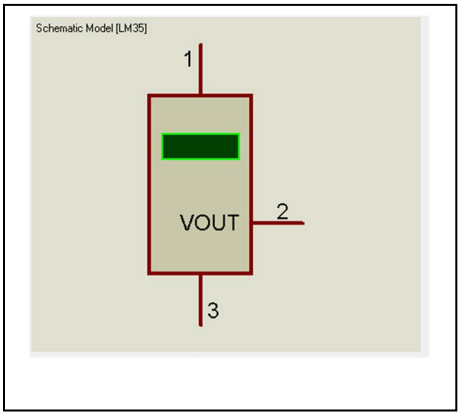
7. SPST Push Button

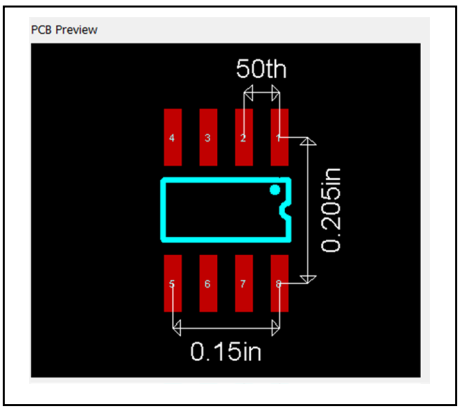


8. Animated LED model (Blue)

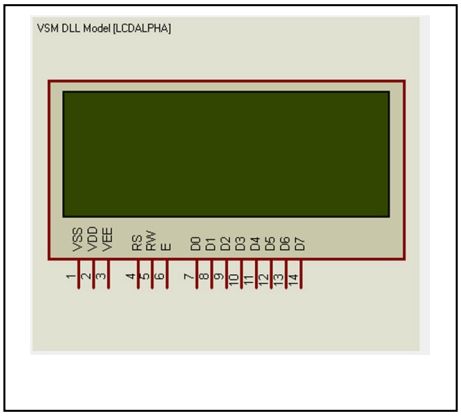


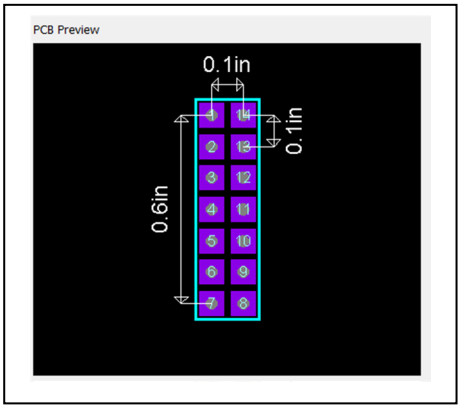
9. Precision Centigrade Temperature Sensor (-55C to 150C)



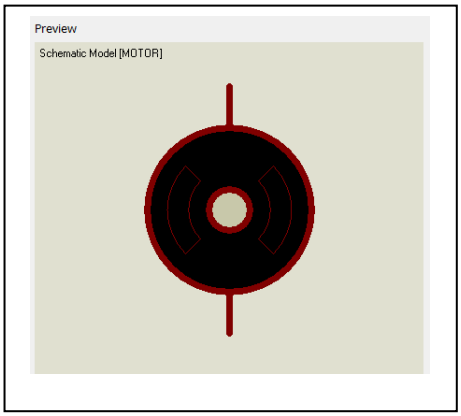


10. 20x4 Alphanumeric LCD

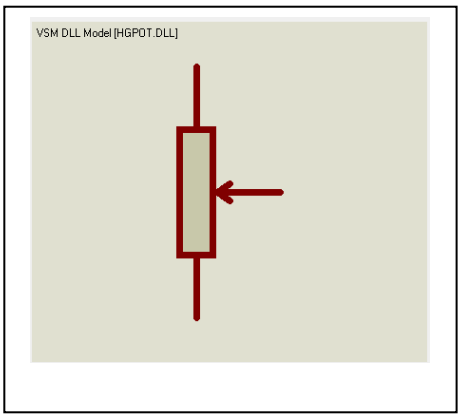




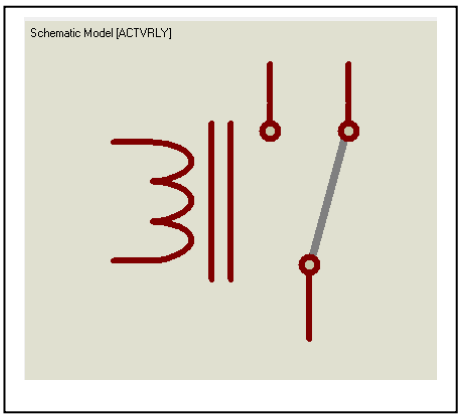
11. Simple DC Motor model



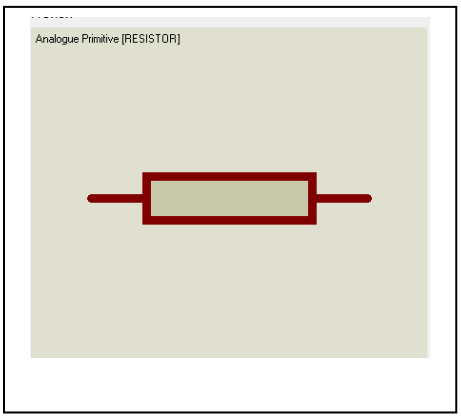
12. High Granularity Interactive Potentiometer (Lin, Log or Antilog Law)



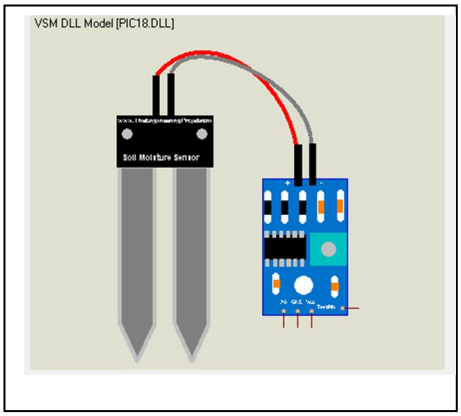
13. Animated Relay model

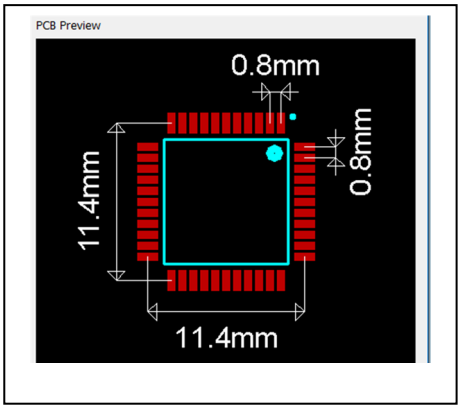


14. Analog resistor primitive

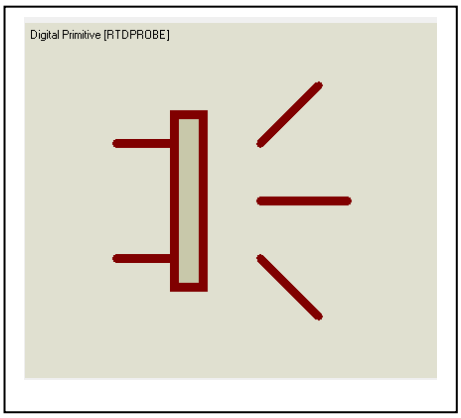


15. Soil Moisture Sensor

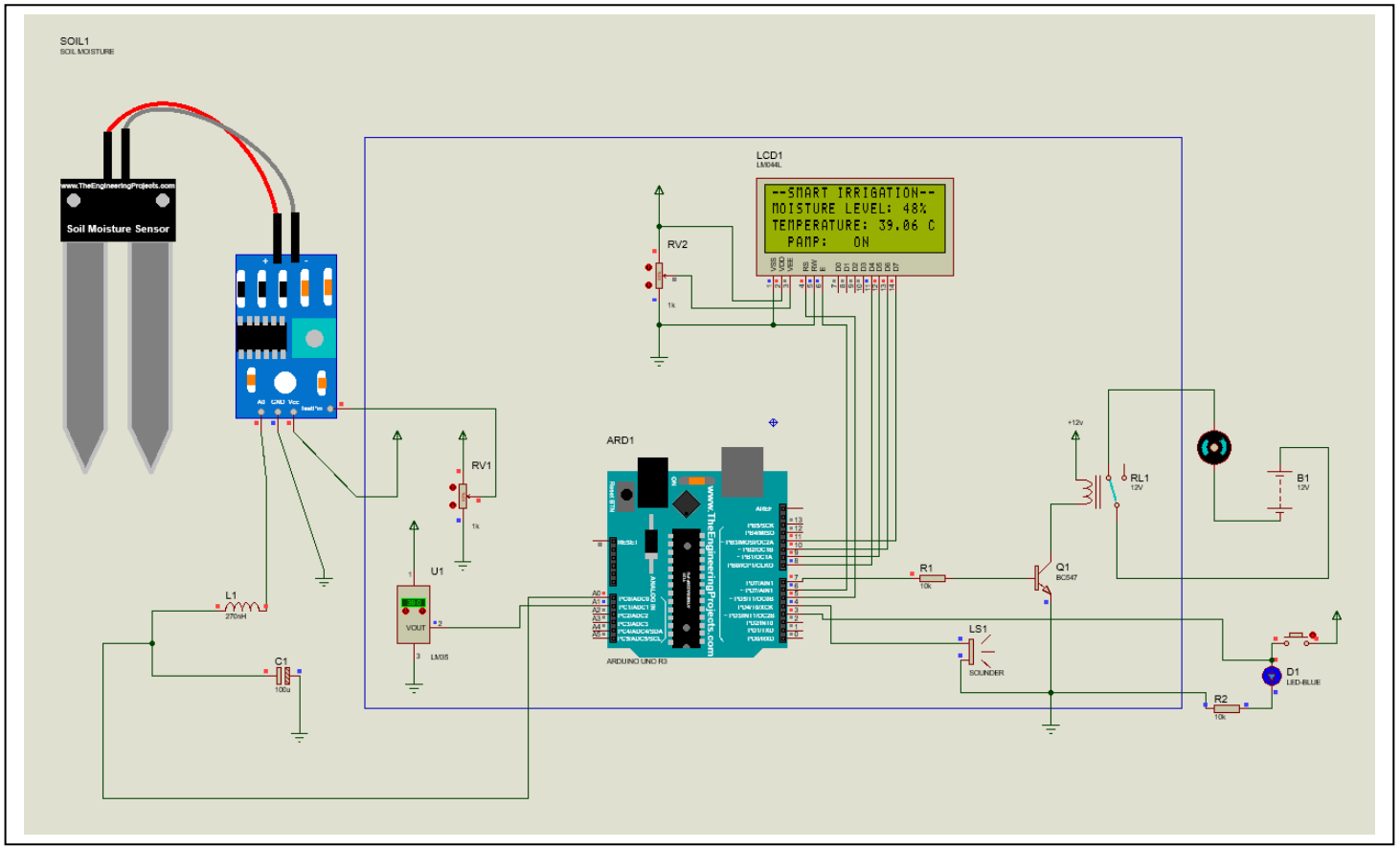




16. Piezo Sounder model (Digital) - Outputs Via Sound Card



Circuit Design



Code

```
#include<LiquidCrystal.h>

LiquidCrystal lcd(5,6,8,9,10,11); // RS,E,D4,D5,D6,D7

int SensorPin = A0;
int tempSensorPin = A1;
int relay = 7;
int ledLight = 3;
int sound = 4;
float temp, com = 30;
int pampVal = 0;

void setup() {

  pinMode(relay, OUTPUT);
  digitalWrite(relay, LOW);

  pinMode(sound, OUTPUT);
  digitalWrite(sound, LOW);

  pinMode(ledLight, OUTPUT);
  digitalWrite(ledLight, LOW);

  lcd.begin(20,4);

}

void loop() {
  int SensorValue = analogRead(SensorPin);
  SensorValue = map(SensorValue, 1019, 19, 0, 100);
  float temp = analogRead(tempSensorPin);
  temp = temp*5000/(1024*10);

  lcd.setCursor(0,0);
  lcd.print("--SMART IRRIGATION--");
  lcd.setCursor(0,1);
  lcd.print("MOISTURE LEVEL:");
```

```
lcd.setCursor(0,2);
lcd.print("TEMPERATURE: ");
lcd.setCursor(2,3);
lcd.print("PAMP: ");

if(SensorValue >= 70)
{
    lcd.setCursor(16, 1);
    lcd.print(SensorValue);
    lcd.print("%");
    lcd.setCursor(13, 2);
    lcd.print(temp);
    lcd.print(" C");
    lcd.setCursor(10, 3);
    lcd.print("OFF");

    digitalWrite(relay, LOW);
    digitalWrite(ledLight, LOW);
    digitalWrite(sound, LOW);

}
  else if(SensorValue < 70 && temp < 40)
  {
    lcd.setCursor(16, 1);
    lcd.print(SensorValue);
    lcd.print("%");
    lcd.setCursor(13, 2);
    lcd.print(temp);
    lcd.print(" C");
    lcd.setCursor(10, 3);
    lcd.print("ON");

    digitalWrite(relay, HIGH);
    digitalWrite(ledLight, HIGH);
    digitalWrite(sound, LOW);

  }
```

```
    else if(temp > 40)
    {
      lcd.setCursor(16, 1);
      lcd.print(SensorValue);
      lcd.print("%");
      lcd.setCursor(13, 2);
      lcd.print(temp);
      lcd.print(" C");
      lcd.setCursor(10, 3);
      lcd.print("OFF");

      digitalWrite(relay, LOW);
      digitalWrite(ledLight, LOW);
      digitalWrite(sound, HIGH);

    }


  delay(5000);
  lcd.clear();


}
```