“ Smart Agriculture Monitoring System Using IoT”

Author

SS Abinesh

R.M.D ENGINEERING COLLEGE

Chennai, TamilNadu India

uec20102@rmd.ac.in

Abstract :

The Internet of Things (IoT) play a crucial role in smart agriculture. Smart farming is an emerging concept because IoT sensors are capable of providing information about their agriculture fields. The paper aims making use of evolving technology ie. IoT and smart agriculture using automation. Monitoring environmental factors is the major factor to improve the yield of efficient crops. The feature of this paper includes monitoring temperature and humidity in agricultural fields through sensors using CC3200 single chip. The camera is interfaced with CC3200 to capture images and send that pictures through MMS to the farmer's mobile using Wi-Fi. Agriculture is the primary occupation in our country for ages. But now due to migration of people from rural to urban, there is a hindrance in agriculture. So to overcome this problem we go for smart agriculture techniques using IoT. This project includes various features like GPS-based remote-controlled monitoring, moisture & temperature sensing, intruder scaring, security, leaf wetness, and proper irrigation facilities. It makes use of wireless sensor networks for nothing to the soil properties, and environmental factors continuously.

Key Words :

Crop Prediction

Fertilizer

Internet of things

Machine learning

smart Farming.

Introduction:

We are going to construct a smart agricultural monitoring system which can collect crucial agricultural data and send it to an IoT platform called Thing speak in real time where the data can be logged and analyzed. The logged data on Thingspeak is in graphical format, a botanist or a reasonably knowledged farmer can analyze the data (from anywhere in the world) to make sensible changes in the supplied resources (to crops) to obtain high quality yield.

What is Smart Agriculture Monitoring System?

Smart agriculture monitoring system or simply smart farming is an emerging technology concept where data from several agricultural fields ranging from small to large scale and its surrounding are collected using smart electronic sensors. The collected data are analyzed by experts and local farmers to draw short term and long term conclusion on weather pattern, soil fertility, current quality of crops, amount of water that will be required for next week to a month etc.

We can take smart farming a step further by automating several parts of farming, for example smart irrigation and water management. We can apply predictive algorithms on microcontrollers or SoC to calculate the amount of water that will be required today for a particular agriculture field.

Say, if there was rain yesterday and the quantity of water required today is going to be less. Similarly if humidity was high the evaporation of water at upper ground level is going to be less, so water required will be less than normal, thus reducing water usage.

Role of IoT in smart farming: Why IoT used in smart farming?

IoT is here to reduce the manual labour involved in collecting these crucial agricultural data. If manual labour is involved we have to deploy several thousands of personnel to different agricultural sites to collect the tedious readings every single day and there will be no assurance in the data integrity since we are humans we may get inert and may manipulate the data which could push the expert conclusions in wrong direction.

Using IoT we can directly send the collected data to a central server in real time. Since we have automated the date collection, the data integrity is assured and since the data processing is done using computers, experts may get advanced analytical software tools to draw most accurate predictions.

Scope for Big Data Analytics in Smart farming:

What is Big Data Analytics?

In layman terms “Big data analytics” is a name given to a field where extraction of useful data from humongous amount of raw data is done. This is accomplished with the help of huge computing power and involves skilled people like data scientists to draw meaningful conclusion and predictions.

What is the need for Big Data Analytics in smart farming?

Collecting data from single person's agriculture farm may help his/her farm to grow. But for a government who is concerned about whole nation's agricultural output, data from single point is not that useful. Ideally they should collect data from every agricultural farm possible.

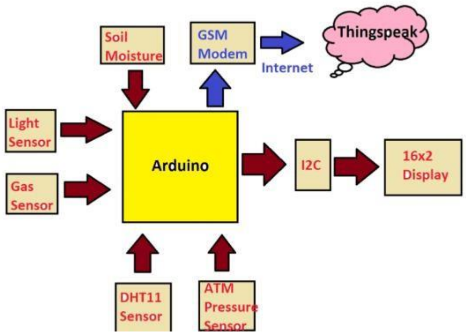
When they do so, they will collect terabytes and petabytes of useful and not so useful data per day. Using big data analytics a government can extract useful data and discard or archive unwanted data.

What to do with the extracted data?

Using these data a government can make useful decisions and announcements on country's agricultural throughput like: how was past year's output, how it is going to be next year, how individual farmers can improve their production with current climate situation and even how much a government need to allocate monetary fund for next financial year for agriculture.

Now you why there is need for IoT and Big Data Analytics in smart farming.

Block Diagram of Smart Agriculture Monitoring System:



This project consists of Arduino as brain and we are utilizing 5 sensors which measures six different environmental factors that crop's growth and nourishment depend on:

1) Temperature and humidity sensor.

2) Air Quality / gas sensor.

3) Light sensor

4) Soil moisture sensor.

5) Barometric pressure sensor.

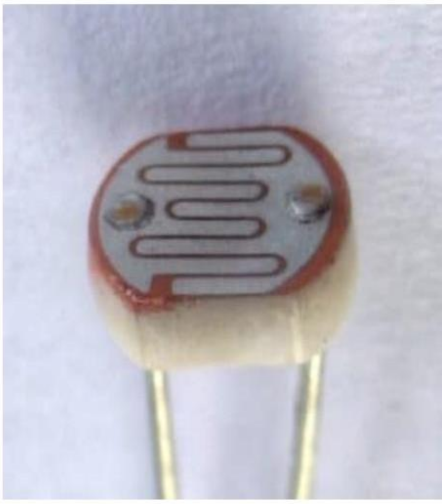
We are also utilizing a GSM module SIM 800 / 900 in the circuit, which can connect to GPRS internet for sending sensor data to thing speak server.

A 16×2 LCD display will display the sensor data which is interfaced with I2C adapter module for reducing the number of Wire that connects from Arduino to LCD.

Now let’s see what are the functions of each sensors mentioned here.

Sensors and its function:

- **Light Sensor**



Photoresistor

The LDR changes its electrical resistance depending on amount of light incident on it. The amount of light is converted to 10-bit digital value and further converted to percentage out of 100.

Zero means no light and 100% means a lot of light.

- **Air Quality Sensor / Gas Sensor:**

MQ135

It is less known fact to many that plants and trees need fresh air for nourishment and growth. Polluted air will make the plants grow sick and we may not get best quality of fruits and vegetables. Polluted air may also make the crops less immune to disease and bugs.

So air quality is a very important parameter to judge the growth of crops, to do this we are using MQ 135 air quality sensor. MQ 135 comes with a breakout board as shown and it has 4 terminals and we are going to use just 3 of them Vcc, GND and Aout which is analog output of the sensor, Dout is not here.

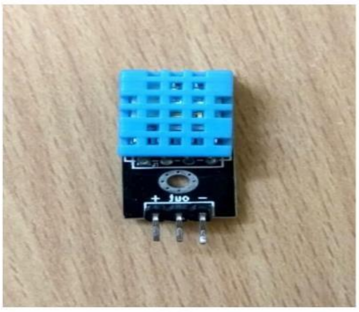
When MQ 135 detects toxic gases the analog output value increases and vice versa.The analog output is converted into 10-bit digital value and converted to percentage out of 100.

100% means lot of air contamination and 0% means least air contamination, so lower the value better the air quality.

The main disadvantage of MQ135 sensor is that it cannot say which pollutant gas is detected.

NOTE: MQ-135 will able give correct output only when the sensor reaches optimum temperature. The sensor will stay heated during its operation and it will take about 5 minutes to reach its optimum temperature.

Temperature and Humidity sensor:

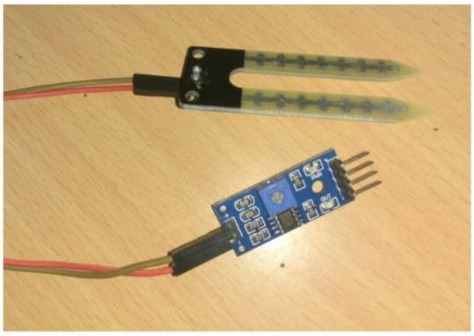


Plants are evolved to be temperature and humidity sensitive just like human or any other living begins. We prepare our self for upcoming winter or summer or rainy seasons so that we can stay comfortable. Similarly plants do prepare themselves for upcoming seasons either to adapt them for the worst or to flourish with fruits and flowers.

So temperature and humidity are important factors in deciding when crops and fruits will get ready to cultivate or begin to produce. This parameter is measured by a digital sensor called DHT11 which can measure both temperature and humidity.

DHT11 sensors are sourced from several different manufacturers and their pin configurations could be different from the one shown in circuit diagram. So it is your task to find the correct pins (Vcc, GND and output) for your DHT11 sensor.

- ## Soil Moisture Sensor:



Soil Moisture Sensor

Plants are of 90% water. The amount of water required varies from plants to plants. The amount of water to be irrigated every day also varies; this depends on how well the soil can hold moisture, current season, temperature and humidity as mentioned in the beginning of this post.

Many farmers irrigate their crops more than sufficient most of the times just to be sure that all their crops received adequate water; this will lead to inefficient management of water.

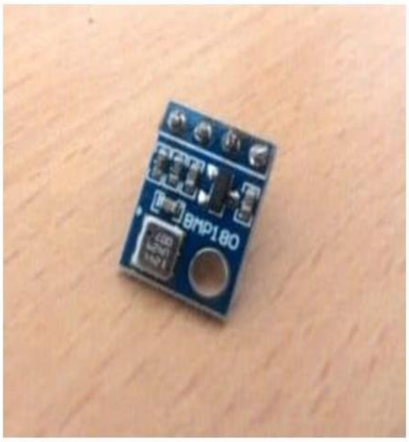
The soil moisture can be measured using the illustrated senor, which has two prongs (electrodes) which are to be inserted on top layer of soil. This is an analog sensor which will output analog values to Arduino.

Soil Moisture Pins

We are going to use only the analog output of this sensor, just like other analog sensors mentioned here; the output is converted to 10-bit digital value and finally to percentage out of 100.

0% means the soil is dry 100% means the soil wet. **But with this sensor we found that anywhere between 50% to 70% reading**, **the soil was fully wet**.

- ### Barometric Pressure Sensor BMP180:



BMP180 Sensor

A barometric pressure sensor can be used for measuring atmospheric pressure. Using atmospheric pressure data you can predict weather for short term and also can be used for studying how plants behave in different atmospheric pressure conditions.

BM180 is a digital sensor and connects to I2C bus and **operates at 3.3V**; it can measure ATM pressure, Altitude and temperature. We are going to extract only the ATM pressure data but you can edit the code and include altitude data to see how plantations behave at different altitude. Temperature data is ignored from this sensor because we already use DHT11 which can measure temperature.

That concludes about the sensors used in this project.

Connecting the project to internet: Why GSM modem is used here instead of ESP8266 or NodeMCU?

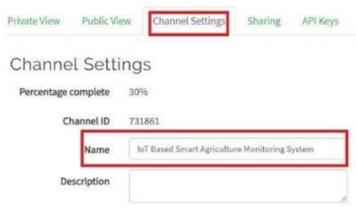
If you are already familiar with some IoT projects, you would have come across ESP8266 or NodeMCU which are IoT enabled boards that connect our projects to internet via Wi-Fi.

In this particular project we are using GSM modem to access GPRS internet, this is because our project will be placed outdoors like middle of an agriculture field where providing Wi-Fi could get difficult and even if we set a Wi-Fi network outdoors anyone could hack into the network.

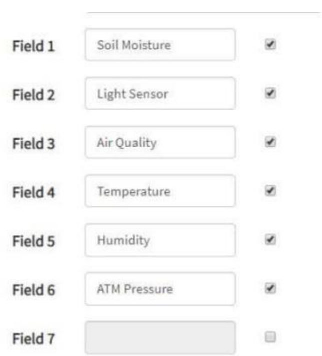
So for these reasons we are using cellular network to connect the project to internet.

Setting up Thingspeak Account for Receiving Sensor Data:

You need to ***sign up for Thing speak*** by entering your E-mail ID and filling up all the necessary credentials asked. Now your channel is ready, but you need to make the following changes in your channel to receive 6 sensor data.**Click on channel settings and name / rename your channel as shown:**



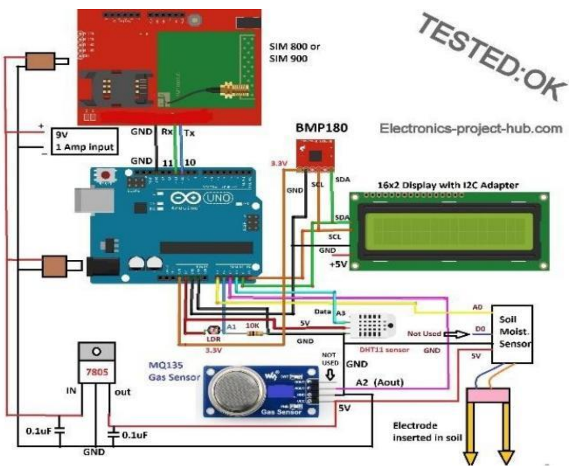
Now enable only 6 fields and name it as shown



Write API Key:

Write API key is a private key consisting of letters and numbers which is used to identify and write values to your channel. This key should be kept confidential if you don't want others writing to your channel. This API key will be inserted in the given program code.

Full Circuit Diagram:



The circuit diagram is self-explanatory. You have to supply 9 to 12V with at-least 1 ampere. IC 7805 is employed here for providing power to MQ135 sensor's heater element which will consumes around 150mA.

Please note that BMP180 works on 3.3V and 5V will kill it and the pins for DHT11 could be different for your DHT11 sensor.

The LCD display is interfaced with I2C adapter, that's why only 4 wires are connected to display in the circuit diagram.

I2C and LCD

You can adjust the contrast of the display by rotating the (blue) potentiometer on I2C adapter.

Installing necessary library files to Arduino IDE:

You need to download and install the following library files to your Arduino IDE before you compile the given code.

1) **LiquidCrystal_I2C.h:** https://github.com/johnrickman/LiquidCrystal_I2C

2) **Adafruit_BMP085.h:** https://github.com/adafruit/Adafruit-BMP085-Library.

Program code: Verified / No error

```
#include  <SoftwareSerial.h>

#include <Wire.h>

#include  <LiquidCrystal_I2C.h>

#include <dht.h>

#include  <Adafruit_BMP085.h>

Adafruit_BMP085 bmp;

dht DHT;

LiquidCrystal_I2C lcd(0x27, 16, 2);

SoftwareSerial gsm(10, 11); // RX, TX

#define DHT11_PIN A3

int chk;

int humi = 0;

int temp = 0;

int soil = 0;

int light = 0;

int BMP = 0;

int gas = 0;

boolean  HT;


void setup()

{

 gsm.begin(9600);

 pinMode(A0, INPUT);

 pinMode(A1, INPUT);

 pinMode(A2, INPUT);

 pinMode(A3, INPUT);

 lcd.init();

 lcd.backlight();

 lcd.setCursor(0, 0);

 lcd.print("Please wait for");

 lcd.setCursor(0, 1);

 lcd.print("60 seconds.");

 delay(20000);

 delay(20000);

 delay(20000);

 modem_init();

 data_init();

 internet_init();

 lcd.clear();

}


void loop()

{

 chk = DHT.read11(DHT11_PIN);

 temp = DHT.temperature;

 humi = DHT.humidity;

 soil  =  analogRead(A0);

 light  =  analogRead(A1);

 gas  =  analogRead(A2);

 BMP = bmp.readPressure();
```

```
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Soil:");
soil = map(soil, 0, 1023, 100, 0);
lcd.print(soil);
lcd.print("%");
lcd.setCursor(0, 1);
lcd.print("Light:");
light = map(light, 0, 1023, 0, 100);
lcd.print(light);
lcd.print("%");
delay(3000);
lcd.clear();
lcd.setCursor(0, 0);
switch (chk)
{
 case DHTLIB_OK:
  HT = true;
  break;
 default:
  HT = false;
  break;
}
if (HT == true)
{
 lcd.print("Temp:");
 lcd.print(temp);
 lcd.print(" *C");
 lcd.setCursor(0, 1);
 lcd.print("Humidity:");
 lcd.print(humi);
 lcd.print("%");
}
else
{
 temp = 0;
 humi = 0;
 lcd.print("Temp:");
 lcd.print("No Data");
 lcd.setCursor(0, 1);
 lcd.print("Humidity:");
 lcd.print("No Data");
}
delay(3000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Air Qlt: ");
gas = map(gas, 0, 1023, 0, 100);
lcd.print(gas);
lcd.print("%");
lcd.setCursor(0, 1);
lcd.print("Pressure:");
```

```
 if (!bmp.begin())
 {
  lcd.print("No Data");
  BMP = 0;
 }
 else
 {
  lcd.print(BMP);
 }
 lcd.print("Pa");
 delay(3000);
 Send_data();
}

void modem_init()
{
 Serial.println("Please wait .. ");
 gsm.println("AT");
 delay(1000);
 gsm.println("AT+CMGF=1");
 delay(1000);
 gsm.println("AT+CNMI=2,2,0,0,0");
 delay(1000);
}

void data_init()
{
 Serial.println("Please wait... ");
 gsm.println("AT");
 delay(1000); delay(1000);
 gsm.println("AT+CPIN?");
 delay(1000);
 delay(1000);
 gsm.print("AT+SAPBR=3,1");
 gsm.write(',');
 gsm.write("");
 gsm.print("contype");
 gsm.write("");
 gsm.write(',');
 gsm.write("");
 gsm.print("GPRS");
 gsm.write("");
 gsm.write(0x0d);
 gsm.write(0x0a);
 delay(1000); ;
 gsm.print("AT+SAPBR=3,1");
 gsm.write(',');
 gsm.write("");
 gsm.print("APN");
 gsm.write("");
 gsm.write(',');
 gsm.write("");
```

```
//------------APN ---------//
gsm.print("bsnlnet"); //APN Here
//--------------------------//
gsm.write("");
gsm.write(0x0d);
gsm.write(0x0a);
delay(1000);
gsm.print("AT+SAPBR=3,1");
gsm.write(',');
gsm.write("");
gsm.print("USER");
gsm.write("");
gsm.write(',');
gsm.write("");
gsm.print("  ");
gsm.write("");
gsm.write(0x0d);
gsm.write(0x0a);
delay(1000);
gsm.print("AT+SAPBR=3,1");
gsm.write(',');
gsm.write("");
gsm.print("PWD");
gsm.write("");
gsm.write(',');
gsm.write("");
 gsm.print("  ");
 gsm.write("");
 gsm.write(0x0d);
 gsm.write(0x0a);
 delay(2000);
 gsm.print("AT+SAPBR=1,1");
 gsm.write(0x0d);
 gsm.write(0x0a);
 delay(3000);
}

void internet_init()
{
 Serial.println("Please wait...");
 delay(1000);
 gsm.println("AT+HTTPINIT");
 delay(1000); delay(1000);
 gsm.print("AT+HTTPPARA=");
 gsm.print("");
 gsm.print("CID");
 gsm.print("");
 gsm.print(',');
 gsm.println('1');
 delay(1000);
}
```

```
void Send_data()

{

 lcd.clear();

 lcd.print("Sending the data");

 lcd.setCursor(0, 1);

 lcd.print("to Thingspeak...");

 delay(1500);

 gsm.print("AT+HTTPPARA=");

 gsm.print('"');

 gsm.print("URL");

 gsm.print('"');

 gsm.print(',');

 gsm.print('"');

 gsm.print("http:");

 gsm.print('/');

 gsm.print('/');

 //-----------------------Your API Key Here------------------------ //

 //Replace xxxxxxxxxxx with your write API key.


gsm.print("api.thingspeak.com/update?api_key=xxxxxxxxxxxxx&field1=");

 //-----------------------------------------------------------------------------//

 gsm.print(soil);// >>>>>> variable 1

 gsm.print("&field2=");

 gsm.print(light); // >>>>>> variable 2

 gsm.print("&field3=");

 gsm.print(gas); // >>>>>> variable 3

 gsm.print("&field4=");

 gsm.print(temp); // >>>>>> variable 4

 gsm.print("&field5=");

 gsm.print(humi); // >>>>>> variable 5

 gsm.print("&field6=");

 gsm.print(BMP); // >>>>>> variable 6

 gsm.write(0x0d);

 gsm.write(0x0a);

 delay(1000);

 gsm.println("AT+HTTPACTION=0");

 delay(1000);

}
```

Changes you need to make in the code:

1) You need to insert your write API key in the given code.

2) You need to insert correct APN in the code for yoSIM card (network provider).

What is APN?

APN stands for Access Point Name; it is the gate way between your modem and internet. Any basic phone or smartphone needs to be configured with correct APN for its mobile data connection. GSM module is no different from a basic phone to access internet, we have to configure it with the correct APN for the SIM card you have inserted.

Where I can find correct APN for my SIM card?

A quick Google search will tell you what your APN **for 2G/GSM/GPRS** (not for 3G or 4G or 5G). If can't find the correct APN online, just contact you customer service they will provide you with right information.

We inserted a BSNL SIM card to the GSM modem and the (GPRS) APN for BSNL is **bsnl net** and you need to insert it in the code

APN for other cellular networks:

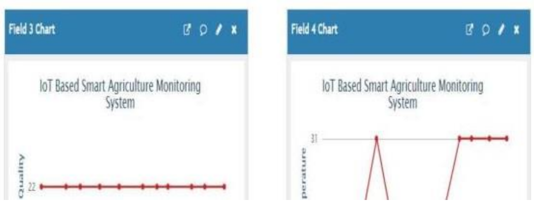
- **Airtel: airtelgrps.com**
- **Vodafone: www**
- **Idea: imis**

- ## How to operate this smart agriculture monitoring system properly:

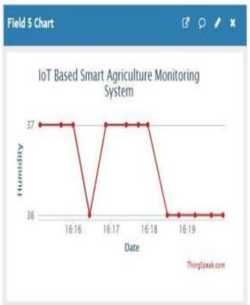
- 1) With completed hardware setup insert a valid SIM card that has a working mobile data plan.
- 2) Upload the program code using USB cable with **correct APN and write API key** (without powering the circuit from wall adapter).
- 3) Once the program is uploaded remove the USB, now turn on the circuit using 9V to 12V / 1 amp (or more) wall adapter.
- 4) The circuit will boot and it tells you to wait for 1 minute. Meanwhile the GSM module is getting ready and latching to the mobile network. The MQ 135 is also getting ready for operation by heating up the metal body.
- 5) After one minute the LCD will display all the sensor data (two at a time).
- 6) Now you can see network LED of GSM module is blinking fast which means your project connected to internet and started to send data to Thing speak.
- 7) Open your Thing speak account and click on private view tab; you will see this (output of this project):

- ## Field 1 and 2: Soil Moisture and Light Sensor

Field 3 and 4: Air Quality and Temperature



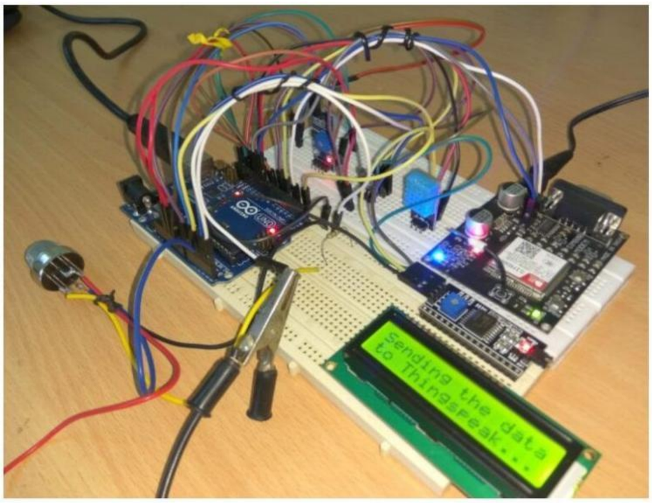
Field 5 and 6: Humidity and Atmospheric Pressure



Conclusion:

Agriculture is done in every country for ages. Agriculture is the science and art of cultivating plants. Agriculture was the key development in the rise of sedentary human civilization. Agriculture is done manually for ages. As the world is trending into new technologies and implementations it is a necessary goal to trend up with agriculture also. IoT plays a very important role in smart agriculture. IoT sensors are capable of providing information about agriculture fields. we have proposed an IOT and smart agriculture system using automation. This IoT-based Agriculture monitoring system makes use of wireless sensor networks that collects data from different sensors deployed at various nodes and sends it through the wireless protocol. This smart agriculture using IOT system is powered by Arduino, it consists of a Temperature sensor, Moisture sensor, water level sensor, DC motor, and GPRS module. When the IOT-based agriculture monitoring system starts it checks the water level, humidity, and moisture level. It sends an SMS alert on the phone about the levels. Sensors sense the level of water if it goes down, it automatically starts the water pump. If the temperature goes above the level, the fan starts. This all is displayed on the LCD module. This is also seen in IOT which shows the information on Humidity, Moisture, and water level with date and time, based on per minute. Temperature can be set on a particular level, it is based on the type of crops cultivated. If we want to close the water forcefully on IOT there is a button given from where the water pump can be forcefully stopped.

PROTOTYPE :



Reference;

1) https://nevonprojects.com/iot-based-smart-agriculture-monitoring-system-project/

2) https://youtu.be/fJehzhiAnso

3) https://circuitdigest.com/microcontroller-projects/iot-based-smart-agriculture-moniotring-system

4) https://link.springer.com/chapter/10.1007/978-981-13-6001-5_39

5) https://circuitdigest.com/microcontroller-projects/iot-based-smart-agriculture-moniotring-system.