IOT SMART AGRICULTURE MONITORING SYSTEM

SPECIAL TOPIC REPORT

TEAM MEMBERS:

- **AAYUSH SUBRAMANIAM** PES1UG19EC002
- **VAISHNAVI SRINIVAS** PES1UG19EC337
- **TARUN AIYAPPA** PES1UG19EC327
- **VAJJHALA KEERTANA** PES1UG19EC338
- **VIVEK AGARWAL** PES1UG19EC355

GUIDE AND MENTOR – ARUN P. S

SESSION: AUG-DEC 2021

DEPARTMENT OF ELECTRONICS AND COMMUNICATIONS

ABSTRACT

India is one of the largest agricultural countries with a population of 1.3 billion, making farming in India labour-intensive and absolute with one-third of the nations’ funds coming from agriculture. After decades of cultivation practice, the country is lagging in maximizing the yield thereby hampering the progress of the nation. To overcome this, there is a need for promoting cultivation practices for producing a high yield of crops. With the availability of IT and the internet, the Internet of Things is proliferating at an unprecedented rate. The perception of agricultural IoT (Internet of things) utilizes networking equipment in farming construction. The hardware part of this project includes processors with data processing capability and sensors which are used to measure various parameters like temperature, humidity, moisture and gases. In this project, the sensor node is designed to monitor the environmental conditions that are vital for the healthy growth of crops. The collected data received are analyzed for proper monitoring and improving the yield of the crop.

CONTENTS

CHAPTER-1
INTRODUCTION

PROBLEM STATEMENT:

To devise a smart agricultural monitoring system that can collect crucial relevant data, directing it to an IoT platform- Thingspeak in real-time where it can be logged and analyzed. The logged data on Thingspeak is in a graphical format so that a botanist or a reasonably knowledgeable farmer can analyze the data to make sensible changes in the supplied resources producing a high-quality yield.

MOTIVATION:

Agriculture is an important aspect in developing countries. In India,it shares about 16.6% of the total gross domestic product. Statistically, more than half of the workforce is engaged with agriculture and its allied sectors. So to fulfil the needs of a continuously growing population, farmers have started to utilize various agricultural techniques, fertilizers and pesticides, hybrid seeds, etc. to improve the declining yield rate due to fertilizer abuse, reducing arable land, fragmentation of agricultural land, agricultural indebtedness, water waste, low soil fertility, climate change etc. In this respect, IoT plays a significant role in implementing the concept of smart farming to automate the farming operations. It is a new computing and communication paradigm in which the objects of everyday life are equipped with sensors, microcontroller and transceiver to sense the surrounding environmental parameters. In addition, communication of the sensed data with one another or user, becoming an integral part of the Internet system. In IoT, every Object used in our daily life with a unique identifier is connected with each other so that they can send data over the network without human intervention.

LITERATURE SURVEY:

V. Puranik, Sharmila, A. Ranjan and A. Kumari, "Automation in Agriculture and IoT," *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*

T. M. Bandara, W. Mudiyanselage and M. Raza, "Smart farm and monitoring system for measuring the Environmental condition using wireless sensor network - IOT Technology in farming," *2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA)*, 2020

Ji-chun Zhao, Jun-feng Zhang, Yu Feng and Jian-xin Guo, "The study and application of the IOT technology in agriculture," *2010 3rd International Conference on Computer Science and Information Technology*, 2010

CHAPTER-2
WORKING PRINCIPLE
&
MODEL

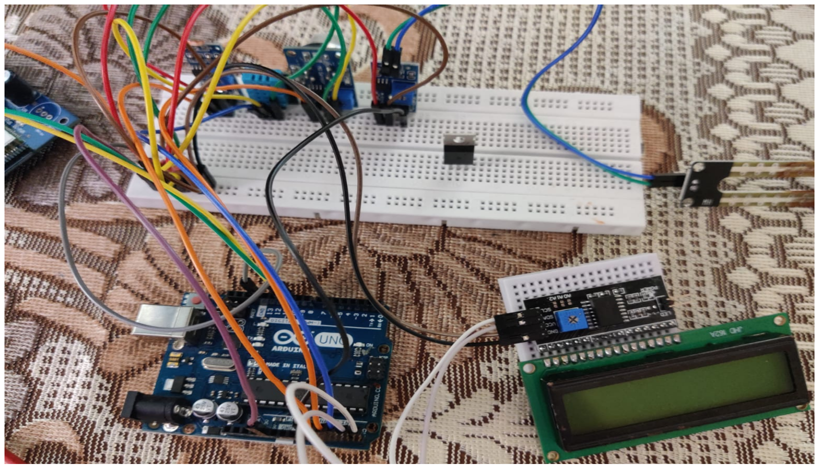
WORKING PRINCIPLE:

Utilizing the sensors (MQ135, DTH11, LDR, soil moisture) to obtain relevant data which is further processed by the arduino, while the sensor data is sent to the thingspeak server with the help of a GSM module in the circuit. The same information is also displayed on the LCD, thus ensuring that the user is constantly updated and aware of the state of his crops.

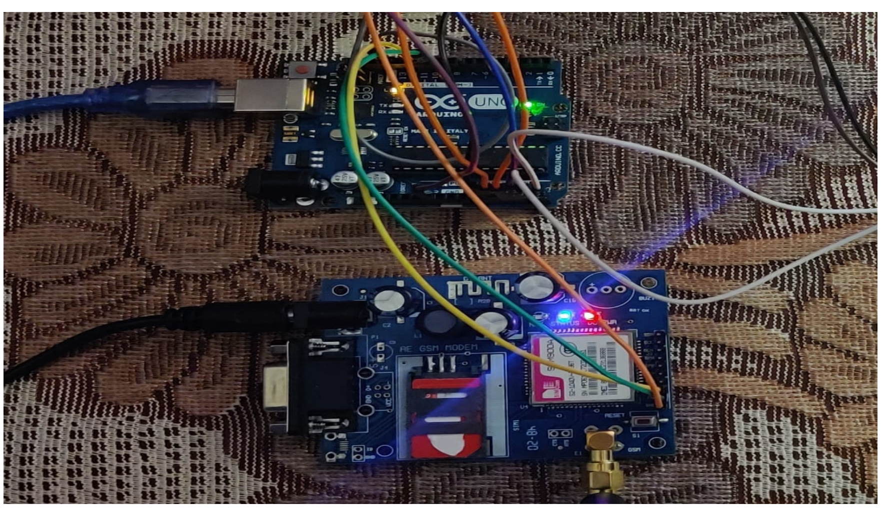
WORKFLOW:

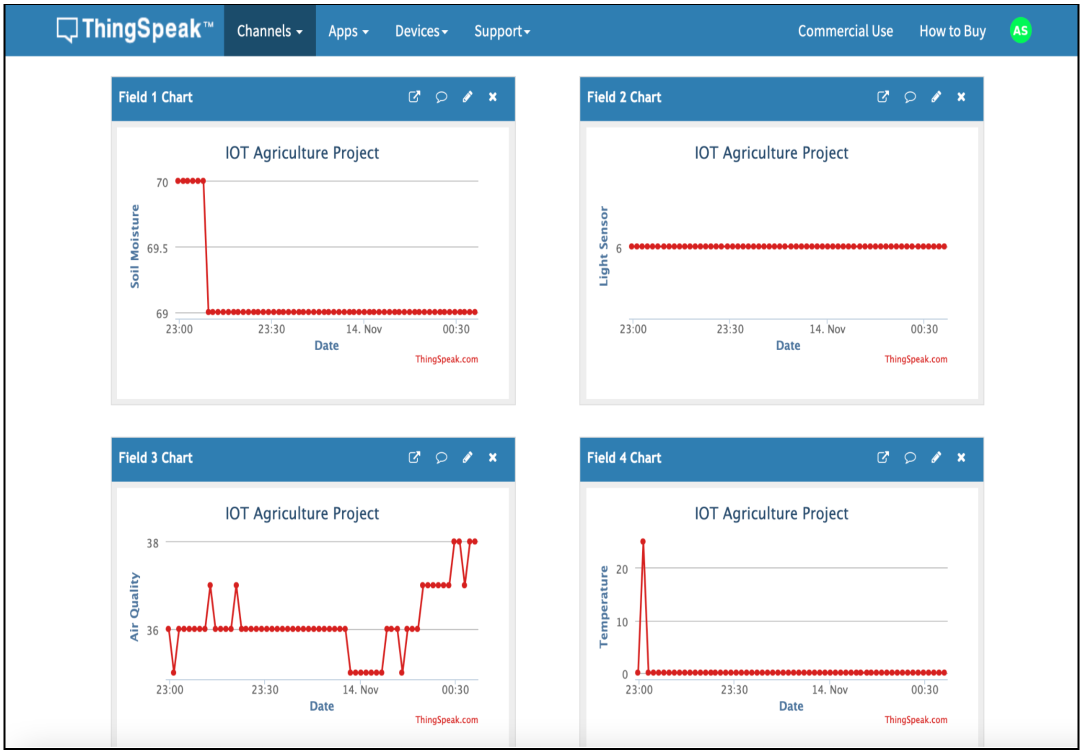
This project consists of two iterations:

1) **LCD Iteration:** This is the data processing segment of the project. 4 Sensors are interfaced with the arduino (using analog and digital pins). When MQ 135 detects toxic gases, the analog output value increases which is further converted into 10-bit digital value and ppm. Temperature and humidity is measured by a digital sensor called DHT11 connected to the digital pins of the arduino. The soil moisture is measured using the illustrated sensor, having two prongs (electrodes) to be inserted on the top layer of soil. This being an analog sensor which will output analog values to Arduino. The LDR sensor changes its electrical resistance depending on the amount of light incident on it, data is converted to 10-bit digital value and ppm. The sensor's data is processed and displayed on the LCD which is interfaced with the I2C adapter thus ensuring a real time monitoring system. (keeping the user updated)



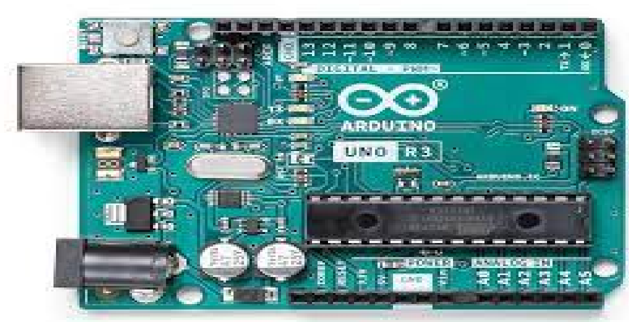
2) **<u>IOT ITERATION</u>:** In this iteration, an IoT component GSM module is used. The Relevant data from the five sensors is processed and directed to thingspeak using a GSM module SIM 800 / 900 connected to GPRS internet for sending sensor data to thingspeak server. Thingspeak is an IoT analytics platform which allows the user to aggregate, visualise and analyze live data streams in the cloud. It also plots graphs of the processed sensor data in ppm.



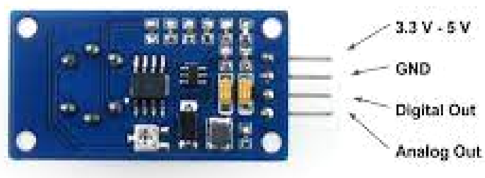


COMPONENTS:

1) **Arduino Uno**: Arduino Uno is the preferred microcontroller as it is easily compatible with the sensor modules that are being used in this project. The values obtained by the sensors are further displayed on the LCD thus ensuring constant monitoring of data.

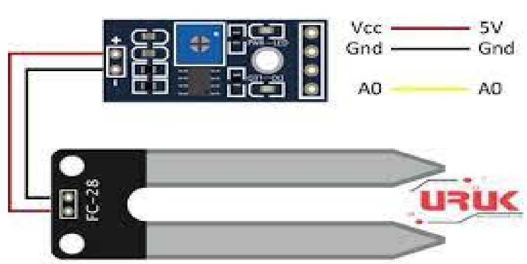


2) MQ-135 Sensor: MQ-135 is the chosen sensor over the other pollution sensors as it is cost-effective and measures a wide range of gases like CO,CO2, NOx, NH3 and Benzene.

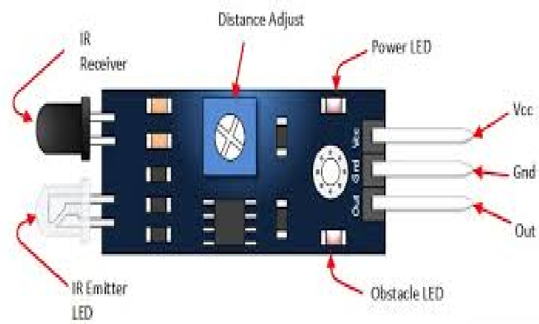


3) Temperature and Humidity sensor (DHT11): The DHT11 is chosen as it's a basic digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal which is displayed on the LCD.

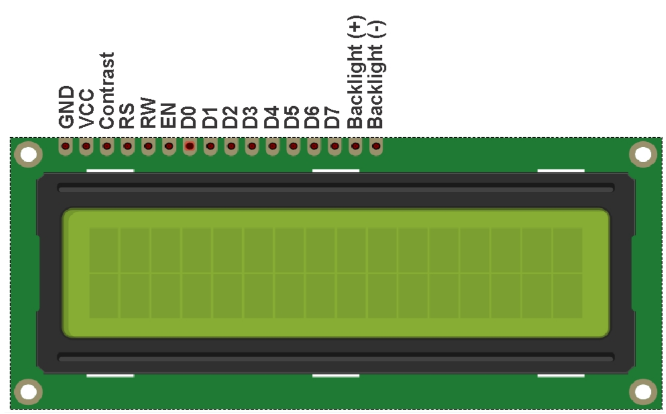
4) Soil moisture sensor: soil moisture sensors measure the volumetric water content indirectly by using some other properties of the soil, such as electrical resistance and dielectric constant.



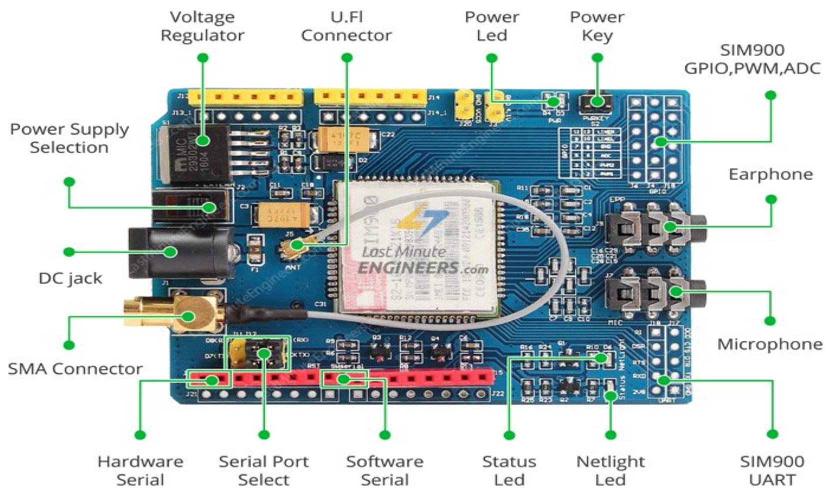
5) **LDR sensor:** LDRs (light-dependent resistors) are used to detect light levels simultaneously recording the real-time values of the physical parameters and feeding them to the microcontroller which is displayed on the LCD.



6) **LCD Display:** The 16x2 LCD is used to display the data obtained by the sensors.



7) GSM module sim 900: A customised Global System for Mobile communication (GSM) module is designed for wireless radiation monitoring through Short Messaging Service (SMS). This module is able to receive serial data from Arduino and transmit the data as a text SMS to a host server.



8) I2c converter: An I2c converter is used to interface the LCD to the arduino.

CHAPTER- 3
CONSTRUCTION
&
STEPS

CONSTRUCTION STEPS :
The circuit is built using the respective hardware components.

- Upon completing the hardware setup (referring to the block diagram), A valid SIM card having a working mobile data plan is inserted in the GSM module.
- Program code is uploaded using a USB cable with the correct APN and write API key (without powering the circuit from the wall adapter).
- Once the program is uploaded, the USB is removed and the circuit is turned on using a 9V to 12V wall adapter.
- The circuit will boot and inform the user 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.
- After one minute the LCD will display all the sensor data (two at a time).
- The network LED of the GSM module starts blinking fast which means that the device is connected to the internet and is sending data to Thingspeak.
- Open your Thingspeak account and click on private view tab; you will see this (output of this project)

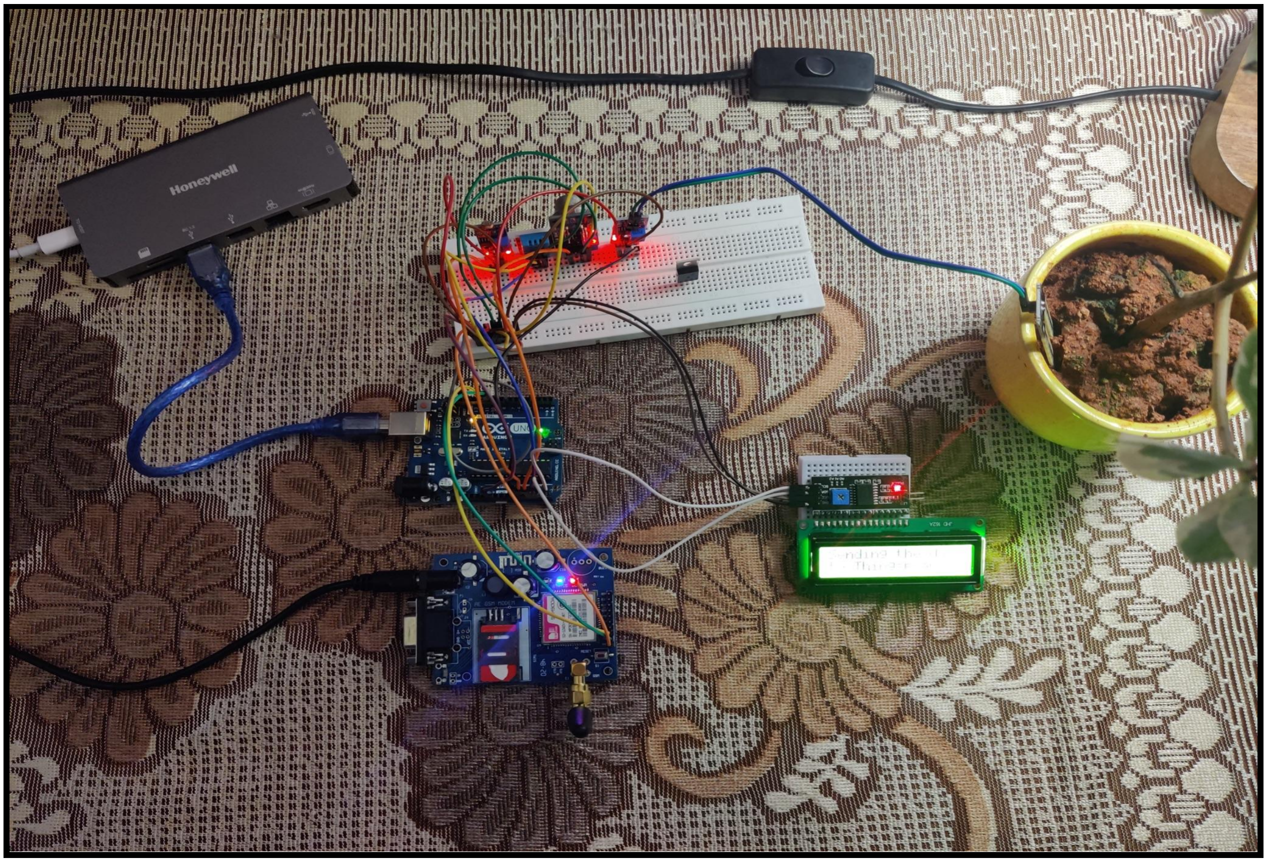
IMPORTANT SNIPPETS OF CODE:

```
void loop()
{
  chk = DHT.read11(DHT11_PIN);
  temp = DHT.temperature;
  humi = DHT.humidity;
  soil = analogRead(A0);
  light = analogRead(A1);
  gas = analogRead(A2);
  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);
```

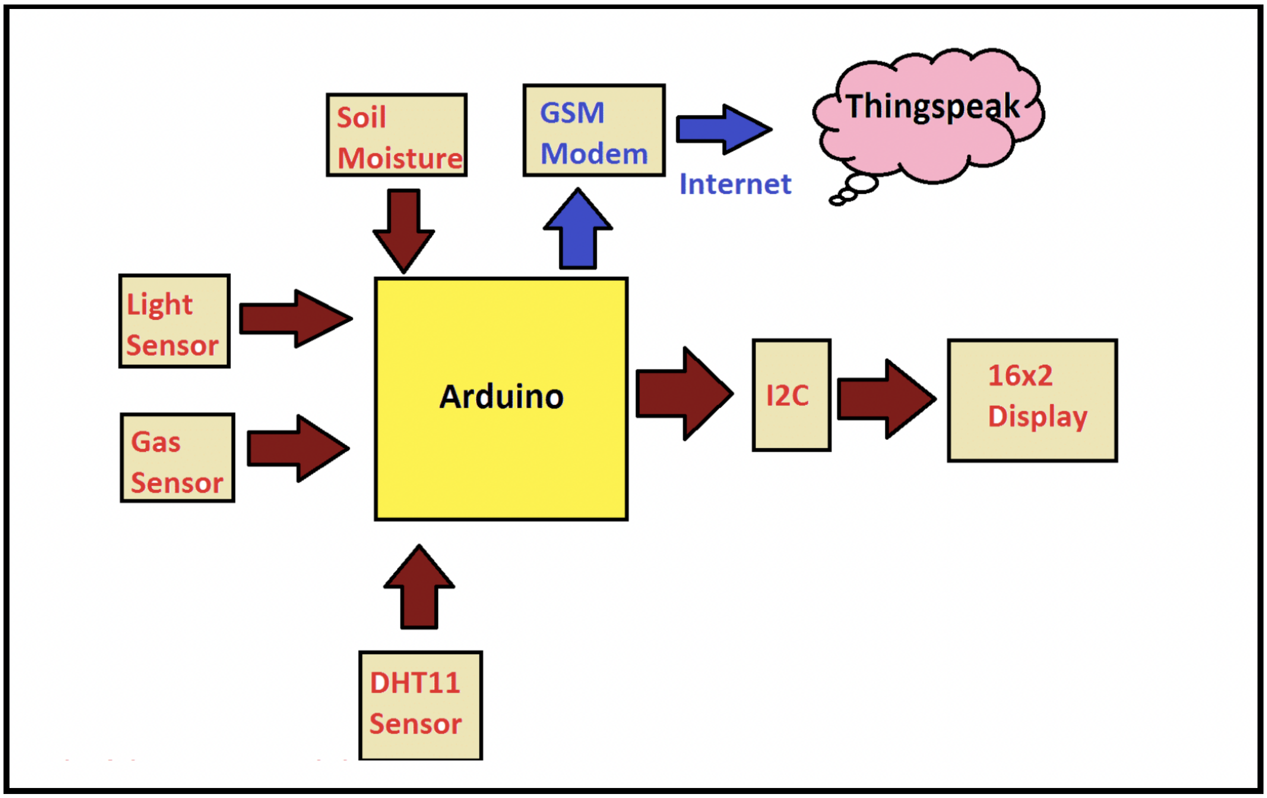
```
    lcd.setCursor(0, 1);
    lcd.print("Humidity:");
    lcd.print(humi);
    lcd.print("%");
  }
  else
  {
    temp = 0;
    humi = 0;
    lcd.print("Temp:");
    lcd.print("No Data");
    lcd.setCursor(0, 1);
    lcd.print("Humidity:");
    lcd.print("No Data");
  }
  delay(3000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Air Qlt: ");
  gas = map(gas, 0, 1023, 0, 100);
  lcd.print(gas);
  lcd.print("%");
  lcd.setCursor(0, 1);
  delay(3000);
  Send_data();

}
```

```
gprsSerial.println("AT+CSTT=\"airtelgprs.com\"");//start task and setting the APN,
delay(1000);

ShowSerialData();

gprsSerial.println("AT+CIICR");//bring up wireless connection
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIFSR");//get local IP adress
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CIPSPRT=0");
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",\"80\"");//start up the connection
delay(6000);

ShowSerialData();

gprsSerial.println("AT+CIPSEND");//begin send data to remote server
delay(4000);
ShowSerialData();

String str="GET https://api.thingspeak.com/update?api_key=ULDRU2ZMYGNI71D1&field1=" + String(soil) +"&field2="+
Serial.println(str);
gprsSerial.println(str);//begin send data to remote server

delay(4000);
ShowSerialData();
```

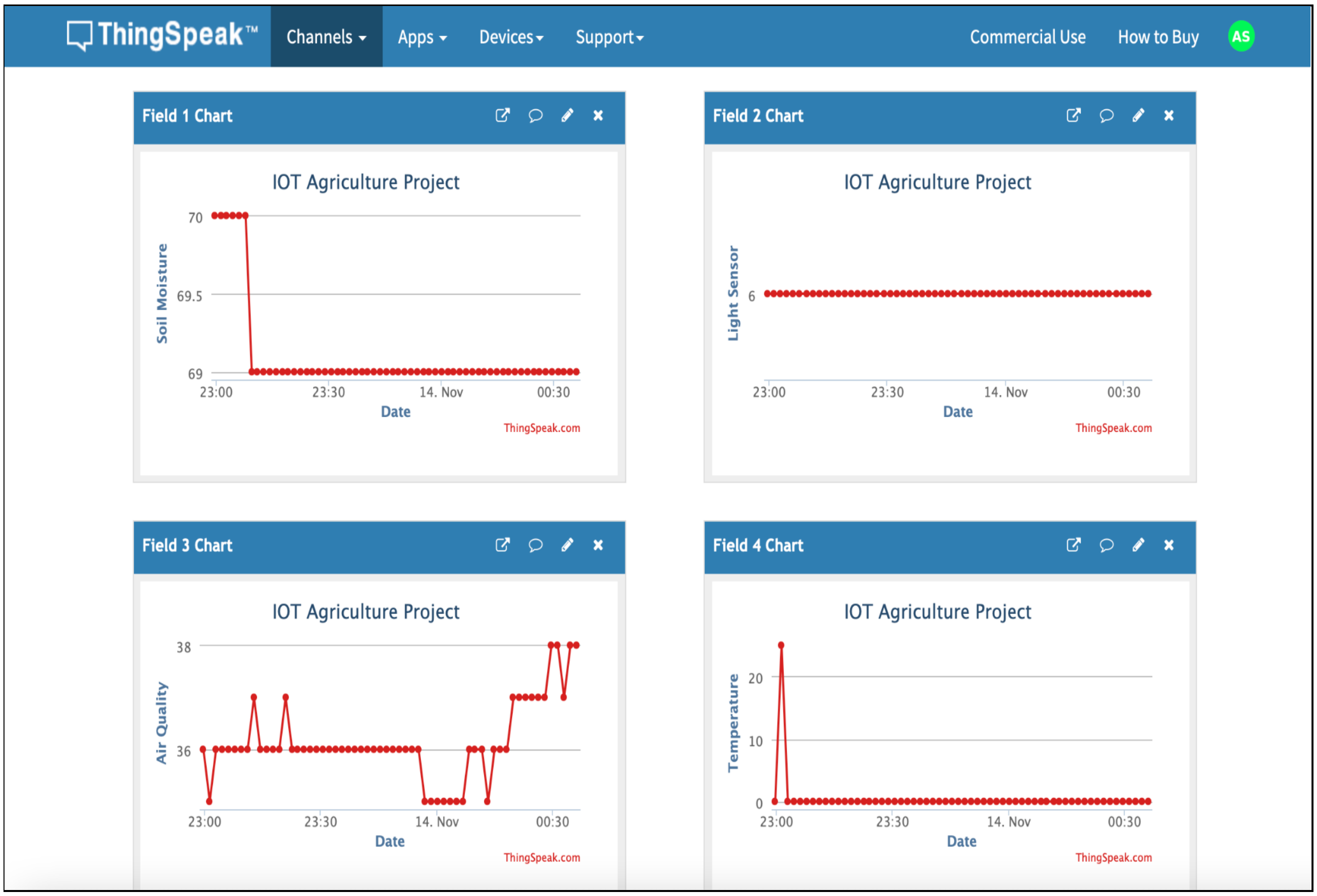
CIRCUIT:



BLOCK DIAGRAM:



THINGSPEAK OUTPUT :



CHAPTER- 4
CONCLUSION
&
FUTURE SCOPE

CONCLUSION:

The main objective here is to build a real time system which is innovative, user friendly, time saving and more efficient than the existing system.
This project Smart Agriculture Monitoring System incorporating Iot, is a simple yet efficient model which is helpful for the rural sector of our country. The system gives real time values thus keeping the user constantly updated about the status of his crops. The LCD iteration informes the user about the health of the crops at each interval of time whereas the IOT iteration alerts the user via SMS when the crops meet an unfavourable condition.
The challenges of an agriculture system in rural areas are lengthy in itself, thus making it quite hard to implement on a large scale. With the right computing power provided by our system, coupled with the fact that it can be understood by any person due to its simple UI, this proves to be a product worth exploring and worthwhile to help our farmers and their livelihood.
In conclusion, this project strides to keep a check on the agricultural area, keeping their user informed about their crop health. Thus, using smart systems not only efficiently takes an advance in environmental quality, but also helps the user dodge unfortunate situations and troubles.

FUTURE SCOPE:

The proposed system is relatively easy to install and delivers accurate readings. Thus it could serve its purpose in different areas and hence holds a lot of scope in the near future. In the future,

- With the help of big data analytics, a single farmer's yield can increase exponentially while also reducing manual labour extensively.
- whole towns or governments could use this to analyse crop yield and requirements in real-time, without waiting for the end product.
- Monetary requirements can be incorporated on a private scale and regulation can be done for farmers and by farmers from the comfort of their own homes.

This project as of this date is a proof of concept. However after multiple iterations and use of higher end microcontrollers, sensors and better software integration a final product can be realised which can be scaled up, licensed and patented if viable.

APPENDIX-1

CODE FOR ARDUINO AND THINGSPEAK:

```
#include <ThingSpeak.h>
#include <SoftwareSerial.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <dht.h>
#include <Adafruit_BMP085.h>
#include <SoftwareSerial.h>
#include <String.h>
dht DHT;
LiquidCrystal_I2C lcd(0x27, 16, 2);
SoftwareSerial gprsSerial(10,11);
#define DHT11_PIN A3
int chk;
float humi = 0;
float temp = 0;
int soil = 0;
int light = 0;
int gas = 0;
boolean HT;

void setup()

{ gprsSerial.begin(9600);
  Serial.begin(9600);
  lcd.begin(16,2);
  gprsSerial.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);
  lcd.clear();

}

void loop()
{
  chk = DHT.read11(DHT11_PIN);
  temp = DHT.temperature;
  humi = DHT.humidity;
  soil = analogRead(A0);
  light = analogRead(A1);
  gas = analogRead(A2);
  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);
  delay(3000);
  Send_data();

}


void Send_data()
{
  Serial.println("Send Data");
  lcd.clear();
  lcd.print("Sending the data");
  lcd.setCursor(0, 1);
  lcd.print("to Thingspeak...");
  delay(1500);

  if (gprsSerial.available())
    Serial.write(gprsSerial.read());

  gprsSerial.println("AT")
```

```
delay(1000);

gprsSerial.println("AT+CPIN?");
delay(1000);
gprsSerial.println("AT+CREG?");
delay(1000);

gprsSerial.println("AT+CGATT?");
delay(1000);

gprsSerial.println("AT+CIPSHUT");
delay(1000);

gprsSerial.println("AT+CIPSTATUS");
delay(2000);

gprsSerial.println("AT+CIPMUX=0");
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CSTT=\"airtelgprs.com\"");//start task and setting the APN,
delay(1000);

ShowSerialData();

gprsSerial.println("AT+CIICR");//bring up wireless connection
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIFSR");//get local IP address
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CIPSPRT=0");
delay(3000);

ShowSerialData();

gprsSerial.println("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",\"80\"");//start up the connection
delay(6000);

ShowSerialData();

gprsSerial.println("AT+CIPSEND");//begin send data to remote server
delay(4000);
ShowSerialData();

String str="GET https://api.thingspeak.com/update?api_key=ULDRU2ZMYGNI71D1&field1=" + String(soil)
+"&field2="+String(light)+"&field3="+String(gas)+"&field4="+String(temp)+"&field5="+String(humi);
Serial.println(str);
gprsSerial.println(str);//begin send data to remote server

delay(4000);
ShowSerialData();

gprsSerial.println((char)26);//sending
delay(5000);//waiting for reply, important! the time is base on the condition of internet
gprsSerial.println();

ShowSerialData();
```

```
  gprsSerial.println("AT+CIPSHUT");//close the connection
  delay(100);
  ShowSerialData();
}
void ShowSerialData()
{
  while(gprsSerial.available()!=0)
  Serial.write(gprsSerial.read());
  delay(5000);

}
```

APPENDIX- 2

REFERENCES AND CITATIONS :

https://www.arduino.cc/reference/en/libraries/dht-sensor-library/

https://www.arduino.cc/reference/en/libraries/thingspeak/

J. Bauer and N. Aschenbruck, "Design and implementation of an agricultural monitoring system for smart farming," *2018 IoT Vertical and Topical Summit on Agriculture - Tuscany (IOT Tuscany)*, 2018, pp.

"Enabling The Smart Agriculture Revolution-The Future of Farming through the IoT Perspective", Tech. Rep., 2016.

D. Vasisht, Z. Kapetanovic, J.-h. Won, X. Jin, R. Chandra, A. Kapoor, et al., "Farmbeats: An IoT Platform for Data-driven Agriculture", Proc. of the 14th USENIX Conference on Networked Systems Design and Implementation (NSDI'17), pp. 515-528, 2017

~~THE END~~