TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

KATHMANDU ENGINEERING COLLEGE
Kalimati, Kathmandu

Minor Project
Final Defense Report On

Smart Green House for Horticulture

Submitted by:

Mausam Gurung (KAT076BEI014)
Shubham Bista (KAT076BEI028)

Department Of
Electronics, Communication
And Information Engineering

Kathmandu, Nepal

April, 2023

ACKNOWLEDGMENT

We would like to express our sincere appreciation to Sir **Er. Sagun Manandhar**, our Minor Project supervisor, for his unwavering support and guidance throughout the project, helping us turn our ideas into reality. His valuable experience served as a guide, instilling us with the confidence and curiosity to tackle problems and see the bigger picture.

We are also deeply grateful to **Er. Rajan Lama**, the Head of the Department of Electronics, Communication, and Information Engineering, for providing us with the opportunity to work on this project.

Our heartfelt thanks go out to our project coordinator, **Er. Sujan Shrestha**, for effectively managing and coordinating the project, and for his timely support and guidance.

We would like to extend our gratitude to all the teaching staff of the Department of Electronics, Communication, and Information Engineering for their constant encouragement, support, constructive criticism, and guidance, and for their keen interest in our project. We are also fortunate to have received support from research paper authors whose interest in our work helped to clear up any blind spots we encountered and refine our ideas. Without their support, our project would not have progressed as smoothly.

ABSTRACT

The main focus of this project is to enhance current greenhouse horticulture practices by incorporating modern technologies for better crop yield. The project involves the development of a smart greenhouse model that automates farm work, reducing the need for manual inspection by farmers. The closed structure of the greenhouse provides protection to plants from extreme weather conditions such as wind, hailstorms, ultraviolet radiation, and pest attacks. Automatic micro-irrigation is used to irrigate the agricultural field, ensuring that the optimal amount of water is applied to the plants based on soil moisture threshold set accordingly. Micro-fertigation techniques are employed to provide plants with the necessary nitrogen, phosphorus, potassium, and other minerals, based on data from soil health card.

In addition, growing lights are used to provide plants with the required wavelength of light during the night. Humidity and temperature sensors are used to monitor and control temperature and air humidity, and a fogger is used to regulate them. Readings from various sensors, including soil moisture, pH, air temperature and humidity (DHT11), BMP180 and LDR, are collected and uploaded to an IOT cloud service. These readings can be accessed via an internet, enabling farmers to monitor important parameters for the quality and productivity of plant growth. The smart greenhouse model not only simplifies the irrigation system, but also makes it more efficient and cost-effective.

List of Content

.

List of Figures

List of Tables

List of Abbreviations

pH: Potential of Hydrogen

LDR: Light dependent Resistor

DHT: Digital Temperature and Humidity Sensor

VCC: Voltage Common Collector

IOT: Internet of Things

LED: Light Emitting Diode

BMP: Barometric Pressure Sensor

CHAPTER 1

INTRODUCTION

1.1 Background Theory

PROTECTED CULTIVATIONS: The use of greenhouse and plastic house techniques has contributed significantly to the improvement of water-use efficiency. The plastic or glass cover creates a special microclimate in which radiation and wind movement are lower but relative air humidity is higher than in the open field, favoring a reduction in evapotranspiration. Furthermore, the higher temperature results in increased plant growth rate and higher yield per unit area of cultivated land. Increase in yield and reduction in water consumption under protected cultivation. Protected cultivation produces higher yields with less water: water-use efficiency is improved. The efficient use of water in greenhouses is also reflected in the efficient use of fertilizers. Many reports on this subject indicate that in plastic houses protected cultivation and soilless culture techniques improve the nutritional conditions and nutritional problems not easily solved under open field conditions. Our smart greenhouse is dedicated to the horticulture. Horticulture is the art of cultivating plants in gardens to produce food and medicinal ingredients, flowers, fruits and nuts, vegetables and herbs.

For the quality and productivity of plant growth and development our project is aimed to automate irrigation system using micros sprinkler, environment observation of temperature and humidity, soil pH for fertile, water contain in soil and light detector.

Soil fertility and the water contain in soil is measured using soil sensor and p^H sensor. If below the threshold point micro sprinkler a low pressure, low medium volume automatically irrigates the plant applying water as uniformly as possible to fill the root zone of the crop along water-soluble fertilizers injected through a micro irrigation thereby significantly reduce cost compared to traditional fertilizer applications. Sometimes Relative Humidity (RH) affects leaf growth, photosynthesis, pollination rate and finally crop yield. Prolonged dry environment or high temperature can make the delicate sepals dry quickly and result in the death of flower before maturity. Hence it is very crucial to control air humidity and temperature. We place DHT11 temperature and humidity sensor inside the smart greenhouse to measure humidity and temperature. When temperature rises above a certain level, Arduino as a microcontroller will trigger relay attached to the fogger, which will sprinkle tiny water droplets of size of micron which will remain suspended in the air and bring the temperature down. In case the air moisture falls below the set value, similar mechanism will be triggered and the small water droplets will maintain the relative humidity (RH). Various wavelengths of light play specific roles for plant growth since different photosynthetic pigments within plants utilize different wavelengths. During morning, leaves receive it directly from sun but in order to boost up the rate of growth, we have provided the greenhouse with plant re-growing lights which will turn on whenever the reading from LDR sensor falls below cut-off value. Advancement in LED technology have made it possible to build LEDs that emit the light in a very specific spectra to achieve very specific outcomes in plant growth. We can affect plants primary and secondary metabolism, which are directly associated with the output quality.

Use of soil moisture, pH sensor, air temperature and humidity sensor monitor to automate irrigation. LDR sensor monitor the presence of sunlight for photosynthesis a plant growing LED bulb. All the monitor data store in the IOT cloud platform which provide statistics in graph to user on their mobile app or web app. The system use part dc part ac for power supply to make the system energy efficient.

saving their time, money and monitored data for the future upgrade. Hence the model through the use of present technology, makes the irrigation system easier, efficient and cost effective thereby saving their time, money & power of the farmer and optimize the water usage and also reduces the need for constant vigilance to provide an appropriate atmosphere for better and efficient crop production.

1.2 Problem Statement

Poorly designed systems which apply water non-uniformly will waste water and chemicals applied with the water.

- Excess applications of water and the resulting leaching of chemicals can result in the pollution of surface or ground water supplies.

Commercially farming people has a responsibility to provide to their commitment for the supply for the food, nursery plant but their lack guarantee.

- Problem with that responsibility is without a smart decision and guarantee their consumer the promise they give because they have no real data they can used for money they need to invest and the expected production they want. Using technology IOT they stored data.

Different plants require different amount of moisture, humidity, temperature and light wavelength, and lack of awareness of this information or negligence of a person cultivating land can cause plants to die before maturing.

The traditional farmland irrigation techniques require manual intervention. With the automated technology of irrigation, the human intervention can be minimized.

1.3 Objectives

- To optimize plant growth and increase production.
- To minimize plant damage and ensure all plants survive.
- To apply pesticides in a safe, effective, and uniform manner.
- To automate irrigation for efficient water usage and prevent over watering and root rot.
- To reduce labor hours and decrease human workforce.
- To collect and analyze statistics on cost, power consumption, fertilizer requirements, and water resources for decision-making purposes.
- To remotely track greenhouse parameters and store data in the cloud for future analysis and decision-making.

1.4 Scope and application of the project

1. Commercial Horticulture
    - Vegetables
    - Flowering and bedding
    - Nursery Ornamentals
    - Plant propagation
2. Bioscience research

Depth of control capability and unique features for demanding bioscience research such as climate monitoring, water management and nutrient studies.

3. Specialty Growing
4. Rooftop Greenhouse
5. Small city area

1.5 Organization of the Report

Chapter 1 is the introductory part of this project report. It deals with the background, objectives, and application of the project. It also contains the scope and problem statement of the project. Chapter 2 deals with the literature review that describes the past works that were undertaken related to this project and also the components that were used in the past. Chapter 3 deals with the conceptual design and outline of the project. Chapter 4 talks about the working methods of various components we use in the project. And finally, chapter 5 includes the result and analysis portion of this report.

CHAPTER 2

LITERATURE REVIEW

The current agricultural practices can be improved using modern technologies for better yield. Greenhouse, is the protected closed structure and control environment. Using automatic drip irrigation, optimal amount of water is applied to the plants according to the soil moisture threshold. This system provides growing light, fogger for temperature and humidity control and GSM module for tube well control. Bee-hive boxes are deployed for pollination and boxes are monitored using ultrasonic sensors to measure honey and send mails to the buyers when they are filled. Further, the readings collected from storage containers are uploaded to cloud service (Google drive) and can be forwarded to an e-commerce company. [1]

Building greenhouse monitoring system based on Internet of Things in which the software for the development board with sensor has been developed with the embedded system and communication technology. Using an analog to digital converter programmable PIC16F877A series microcontroller interface between sensing arrangement and GSM/GPRS (Global System for Mobile Communication) and Internet of Things (IOT) provide information. [2]

The real time remote monitoring system using Raspberry Pi which enables the user to track the different parameters in green house remotely for improving plant growth. The data storage in the database on the cloud for future use in any internet enabled device. This facilitates farmer to take right decisions at right time to obtain desired results in plant growth. [3]

Smart Agriculture reduces wastage of water, fertilizers pesticides and increases the crop yield. By monitoring soil moisture, humidity and temperature, Light Sensor the irrigation system can be automated if soil moisture is low. [4]

The results from the greenhouse parameters are analyzed and represent with the help of graphical representation based on the practical values taken by the IOT kit make analysis easy to analysis. [5]

Due to poorly irrigation system which apply water non-uniformly will waste water and chemical applied with water. Some areas will be over-irrigated, while others will be under-irrigated, the under-irrigated areas will suffer yield or quality reductions due to water stress, while yield or quality reductions due to leaching of water and chemicals will occur in the over-irrigated area. [6]

Most of the people loves to grow plants at home, but due to their work schedule they very often take care of plants. People often waste lot of time for watering the plants so an efficient management of water develop in this project and main theme is to increase the plant growth condition. [7]

CHAPTER 3

RELATED THEORY

For the implementation of any system, hardware and software components are the essential parts. Thus, the integral parts of the hardware that we used in our minor project are mentioned below:

3.1 Introduction of IOT

IOT has emerged as a new technology that is used to express a modern wireless telecommunication network, and it can be defined as an intelligent and interoperability node interconnected in a dynamic global infrastructure network, also it seeks to implement the connectivity concept of anything from anywhere at anytime

The vision of IOT is to enable networked devices to propagate their information about physical world objects through the web. IOT stands for Internet of Things, which means accessing and controlling daily usable equipment and devices using Internet.

3.2 Green House

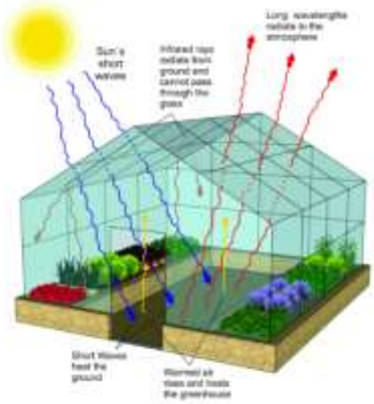


Figure 1: Green house

A greenhouse is a structure with walls and roof made chiefly of transparent material, such as glass, in which plants requiring regulated climatic conditions are grown.

A greenhouse works by **converting light energy into heat**. Light enters the greenhouse and is trapped there by the glass and absorbed by the plants and other objects. This causes the light energy to be converted to heat energy, which is trapped inside the greenhouse. Greenhouses are used to provide optimal environments for plant growth and development.

Our greenhouse is dedicated to the horticulture (Horticulture is the art of cultivating plants in gardens to produce food and medicinal ingredients, flowers, fruits and nuts, vegetables and herbs.).

3.3 Hardware

3.3.1 Arduino MEGA 2560

Figure 2: Arduino MEGA 2560

The **Arduino Mega 2560** is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. Detail table about Arduino mega is as shown below:

MICROCONTROLLER	ATmega2560
OPERATING VOLTAGE	5V
INPUT VOLTAGE (RECOMMENDED)	7-12V
INPUT VOLTAGE (LIMIT)	6-20V
DIGITAL I/O PINS	54 (of which 15 provide PWM output)
ANALOG INPUT PINS	16
DC CURRENT PER I/O PIN	20 mA
DC CURRENT FOR 3.3V PIN	50 mA
FLASH MEMORY	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
CLOCK SPEED	16 MHz
LED_BUILTIN	13
LENGTH	101.52 mm
WIDTH	53.3 mm
WEIGHT	37 g

Table 1: Details of Arduino mega

Pin Configuration Diagram

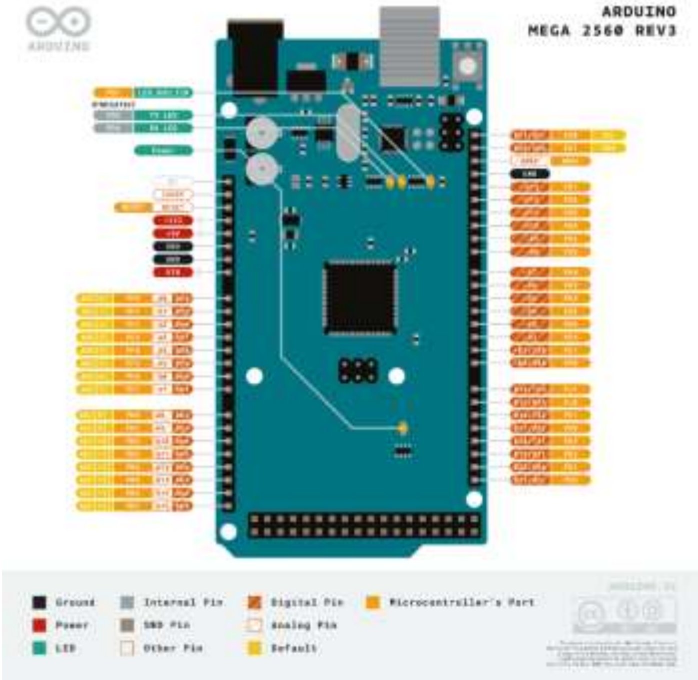


Figure 3: Pin configuration Diagram

3.3.2 DHT 11

Figure 4: DHT11 Sensor

The DHT11 sensor is a digital humidity and temperature sensor that can be used to measure air temperature and humidity in various applications and can be easily interfaced with microcontrollers such as Arduino boards. The DHT11 sensor consists of a capacitive humidity sensor and a thermistor for temperature measurement, both of which are connected to a single-wire digital interface. The sensor is powered by 3.3 to 5V DC and can output data in a digital signal format.

The humidity measurement range of the DHT11 sensor is from 20% to 80% with an accuracy of ±5%, while the temperature measurement range is from 0°C to 50°C with an accuracy of ±2°C. The response time of the sensor is typically 2 seconds or less, making it suitable for real-time monitoring applications.

3.3.3 LDR sensor

Figure 5: LDR Sensor

Light dependent resistor (LDR): The resistor whose conductivity changes according to the amount of light falling on it is called an LDR. The intensity of light is recorded. As soon as the threshold of light intensity is reached, the flippers are operated in order to cover the field and protect the crops from high intensity of light

A Light Sensor generates an output signal indicating the intensity of light by measuring the radiant energy that exists in a very narrow range of frequencies basically called "light", and which ranges in frequency from "Infra-red" to "Visible" up to "Ultraviolet" light spectrum.

The light sensor is a passive device that convert this "light energy" whether visible or in the infra-red parts of the spectrum into an electrical signal output. Light sensors are more commonly known as "Photoelectric Devices" or "Photo Sensors" because the convert light energy (photons) into electricity (electrons)

3.3.2 BMP 180 sensor

Figure 6: BMP 180 sensor

BMP180 is a high-precision digital barometric pressure sensor that can also measure temperature. It is commonly used in various applications such as weather stations, altitude sensing, and navigation systems. The BMP180 sensor measures atmospheric pressure by sensing the deflection of a diaphragm caused by the pressure of the atmosphere. The pressure is converted into a digital signal and can be read through an I2C interface. The temperature measurement is also integrated into the sensor, allowing for temperature compensation of the pressure measurement.

The pressure measurement range of the BMP180 sensor is from 300 hPa to 1100 hPa with an accuracy of up to ±0.12 hPa, while the temperature measurement range is from -40°C to +85°C with an accuracy of up to ±2°C. The sensor also has a low power consumption mode, making it suitable for battery-powered applications.

3.3.4 Soil moisture sensor

Figure 7: Soil moisture sensor

A soil moisture sensor is a device that measures the amount of moisture present in the soil. It is an important tool for monitoring soil moisture levels and ensuring proper irrigation for plants. The sensor typically works by measuring the dielectric constant of the soil, which is related to the soil's water content.

There are various types of soil moisture sensors, including resistive sensors, capacitive sensors, and TDR (Time Domain Reflectometry) sensors. Resistive sensors measure the resistance of the soil, which is related to its water content, while capacitive sensors measure the capacitance of the soil, which is also related to its water content. TDR sensors, on the other hand, measure the time it takes for an electromagnetic wave to travel through the soil, which is also related to the soil's water content.

Soil moisture sensors are commonly used in agriculture and horticulture to optimize irrigation and ensure optimal growing conditions for crops. By monitoring soil moisture levels, farmers and gardeners can ensure that their crops receive the appropriate amount of water, which can help to maximize crop yield and quality, as well as reduce water waste.

3.3.5 pH sensor

Figure 8: pH sensor

A pH sensor is a device that measures the acidity or alkalinity of a liquid or solution. pH is a measure of the concentration of hydrogen ions in a solution, and it ranges from 0 to 14, where 0 is highly acidic, 7 is neutral, and 14 is highly alkaline.

A typical pH sensor consists of two electrodes, a reference electrode and a measuring electrode. The reference electrode is typically a glass electrode filled with a special solution that has a known pH value. The measuring electrode is usually a metal electrode coated with a thin layer of glass or plastic, which acts as a membrane that allows hydrogen ions to pass through.

When the sensor is placed in a liquid, the reference electrode and the measuring electrode generate a small electrical potential that is proportional to the hydrogen ion concentration in the solution. This potential is measured by the pH meter or microcontroller to determine the pH value of the solution.

3.3.6 ESP8266

Figure 9: ESP8266

The ESP8266 is a Wi-Fi microchip with full TCP/IP stack and microcontroller capability. This small module allows microcontrollers to connect to a Wi-Fi network and make simple HTTP requests. The ESP8266 is used to add Wi-Fi connectivity to microcontroller projects, such as home automation systems, weather stations, and IOT devices. It can operate as a standalone device, or it can be integrated into a larger microcontroller project. The chip can be programmed using the Arduino IDE.

3.3.7 Sprayer with servo motor MG995 as fogger

Figure 10: Servo motor

A fogger is a piece of equipment that turns a certain volume of liquid into a mist or fog via a high rate of pressure. Depending on the liquid, a fogger can be used to cool the greenhouse, increase humidity, or spray pest control substances. Foggers can be stationary, portable, handheld, or aerosol canisters.

In practice, most fogger irrigation produce particles in the 10–30-micron range. For nurseries, shrubs and trees where a fine mist spray is required at low volumes and low pressures. Foggers can also assist in temperature control and to increase humidity in hot, dry climates. It's rugged nylon construction and easy disassembly allow for carefree operation

3.3.8 Growing LED

Figure 11: Growing LED

Growing LED lights are switched on whenever light intensity is low for photosynthesis; this ensures faster rate of growth. Various wavelengths of light plays specific roles for plant growth since different photosynthetic pigments within plants utilize different wavelengths. During morning, leaves receive it directly from sun but in order to boost up the rate of growth, we have provided the greenhouse with plant growing lights which will turn on whenever the reading from LDR sensor falls below cut-off value. Advancement in LED technology have made it possible to build LEDs that emit the light in a very specific spectra to achieve very specific outcomes in plant growth. We can affect plants primary and secondary metabolism, which are directly associated with the output quality.

3.3.9 12v solenoid valve

Figure 12: 12v Solenoid valve

12v solenoid valves are typically used for simple ON/OFF control of gas or liquid media 12-volt solenoid is designed to work with a 12-volt direct current (DC) or alternating current (AC) power supply. A 12-volt solenoid is an electromagnetic actuation device designed to work with a 12-volt direct current (DC) or alternating current (AC) power supply.

Solenoid valve is an electro-mechanical valve that is commonly employed to control the flow of liquid or gas. Solenoid valve function involves either opening or closing an orifice in a valve body, which either allows or prevents flow through the valve

3.3.10 Sprinkler

Figure 13: Sprinkler

Sprinkler irrigation is a low pressure, low to medium volume irrigation system suitable for high value crops such as tree fruits. If managed properly, micro irrigation can increase yields and decrease water use and fertilizer and labor requirements when compared to gated pipe/ furrow irrigation systems. Micro-sprinkler irrigation saves water because of the high application efficiency and high-water distribution uniformity with little if any waste if managed properly. Water-soluble fertilizers can be injected through a micro irrigation system thereby significantly reducing cost compared to traditional fertilizer applications.

3.4 Software

3.4.1 Arduino IDE

Arduino IDE is used to provide the backend functionality to the system, a communication is established between Arduino module and other components. It provides the base to transmit and receive message from the user. Arduino ide use c language to create a mobile and web application that will provide the graphical user interface to manually control the working of the system and provide the user with the live details of various sensor readings included in the system

3.4.2 Arduino IoT Cloud

Arduino IoT Cloud is a cloud platform developed by Arduino for building Internet of Things (IoT) applications. It provides a variety of tools and services for creating, managing, and deploying IoT projects.

One of the key features of Arduino IoT Cloud is its visual programming interface, which allows users to create IoT applications using a drag-and-drop interface. It also includes a range of pre-built libraries and tools for connecting to sensors and devices, as well as integrations with other popular IoT platforms and services.

With Arduino IoT Cloud, users can create and deploy IoT applications in a matter of minutes, without the need for complex coding or technical expertise. It is suitable for a wide range of IoT projects, from simple home automation systems to more complex industrial and commercial applications.

3.4.3 JSON

JSON is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate. In the code mentioned earlier, JSON was used to store the data collected from the sensors in a structured format. The data was then sent to a web server using the HTTP protocol. On the web server, the JSON data was parsed and stored in a database or processed further as required. JSON is a commonly used data format in web development and can be used in many programming languages.

3.5 Threshold value for tomato

The ideal temperature range for a greenhouse varies depending on the plants being grown. In general, the optimal temperature for most greenhouse crops is between 18°C to 23°C during the day and 13°C to 18°C at night. For testing purpose we took the reference of tomato plant, and set the value accordingly.

pH range

The optimal pH range for growing tomatoes is between 6.0 to 6.8. pH values outside of this range may lead to nutrient deficiencies and other growth problems. It is recommended to test the soil pH and adjust it accordingly with lime or sulfur, depending on whether the pH is too acidic or too alkaline.

Soil Moisture

Generally, the optimal soil moisture range for tomato is between 60% to 80% of the soil's water-holding capacity. It's recommended to keep the soil moist but not waterlogged to avoid the development of root diseases. Inadequate soil moisture can lead to stress, reduced growth, and yield loss, while excessive soil moisture can cause root rot and other diseases. Therefore, regular monitoring of soil moisture is necessary to maintain optimal growing conditions for tomatoes.

Humidity

Tomatoes generally prefer moderate humidity levels, ideally between 50% to 70%. High humidity levels can increase the risk of fungal diseases, such as powdery mildew, while low humidity levels can cause the plants to lose moisture and become stressed.

During the seedling stage, it's recommended to maintain higher humidity levels (around 70% to 80%) to support germination and growth. Once the plants begin to mature and set fruit, it's best to gradually decrease humidity levels to prevent disease and promote fruit development.

Temperature

Tomatoes are warm-season crops that require warm temperatures to grow and thrive. The ideal temperature for tomato growth is between 20°C to 30°C during the day and 15°C to 20°C at night.

Light intensity

Tomato plants need at least 6-8 hours of sunlight per day to grow and produce fruit. They can also tolerate some shade, but it may affect the yield and quality of the fruit.

CHAPTER 4

METHODOLOGY

4.1 System Block Diagram

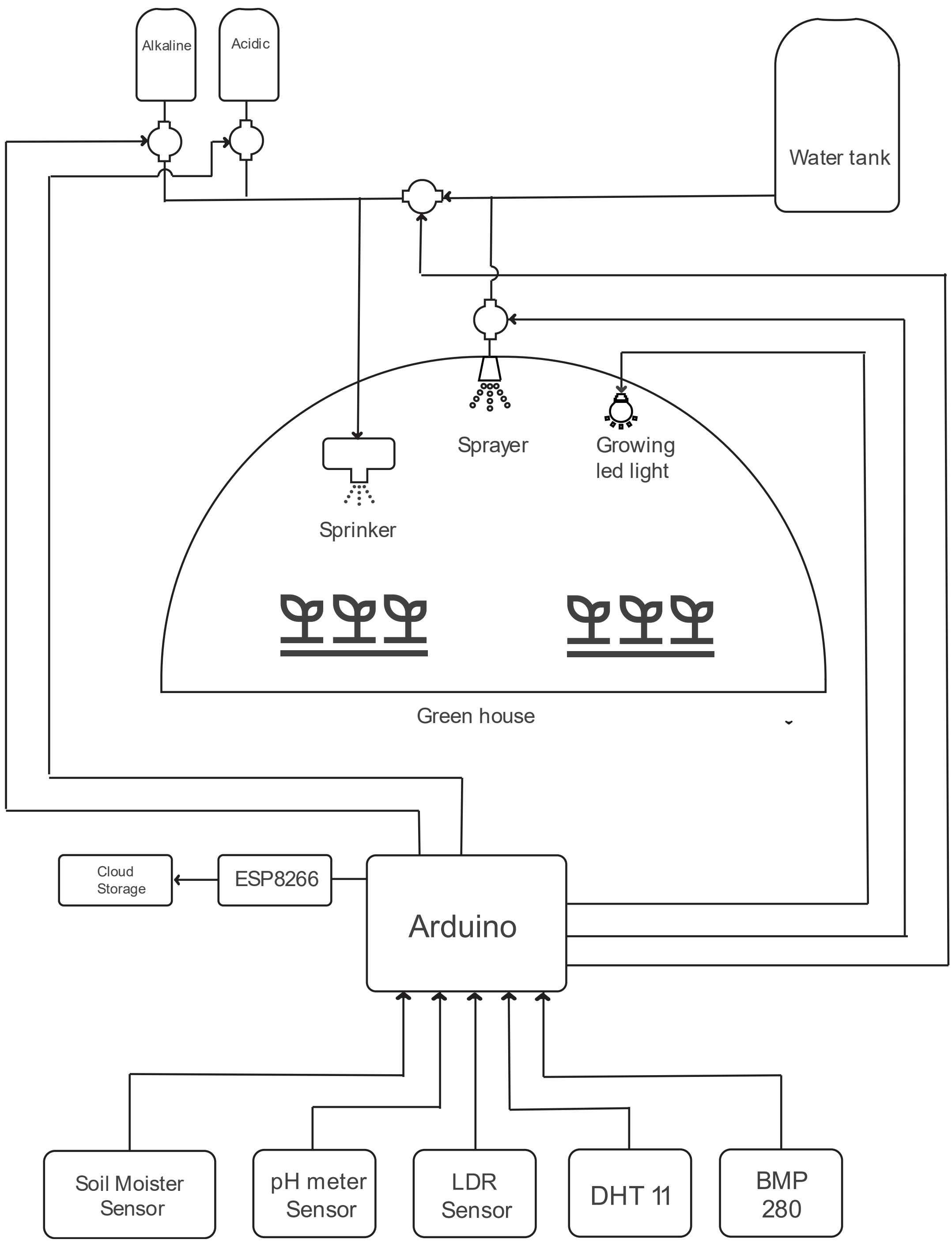


Figure 14: System Block Diagram

Our project aims to create an affordable IoT-based system for monitoring a greenhouse. We have designed a block diagram that includes sensors to monitor soil moisture, pH of soil and environmental conditions such as light intensity, humidity, and temperature. The real-time data collected by each sensor is sent to an Arduino microcontroller and then to a cloud-based system for storage with the help of ESP module.

Soil is crucial for plant growth and is monitored using a capacitive soil moisture sensor. We can determine when and how much to water the plants automatically from the water tank, which is controlled by a solenoid valve, based on the soil moisture rating. Our system also includes a pH sensor to measure soil acidity and alkalinity, with a range of 6.5-7.5 being optimal for normal plant growth. If the pH falls below or above this range, the system automatically triggers the fertilizer to be applied.

Environmental factors like temperature, humidity, and light intensity are important for plant growth. To monitor these factors, we have included a DHT11 sensor and an LDR sensor in our design. The DHT11 sensor measures temperature and humidity, and the data is used to trigger a fogger if needed. The LDR sensor measures light intensity and triggers the plant-growing LED bulb, which provides the specific wavelength of light needed for photosynthesis.

The monitor data also triggers the necessary solenoid valve and is forwarded to the cloud-based system for storage.

4.2 Algorithm

Algorithm to monitor environmental conditions for a plant:

1.Initialize the necessary sensors and output devices, such as the DHT11 temperature and humidity sensor, pH sensor, soil moisture sensor, LDR sensor, and a servo motor to control a fogger.

2.Set the acceptable ranges for temperature, humidity, pH, and soil moisture.

3.Continuously read the values from the temperature and humidity sensor and check if they are within the acceptable range. Turn on a green LED if they are and turn on a red LED if they are not. Additionally, if the temperature or humidity falls below the minimum acceptable value, activate the servo motor to turn on the fogger for 10 times.

4.Continuously read the value from the pH sensor and check if it is within the acceptable range. Turn on an alkaline LED if it is too high and turn on an acidic LED if it is too low.

5.Continuously read the value from the soil moisture sensor and check if it is within the acceptable range. Turn on a green LED if it is and turn on a red LED if it is not. If the soil moisture falls below the minimum acceptable value, activate a water pump to supply water to the plants.

6.Continuously read the value from the LDR sensor and check if it is above or below a threshold value. Turn on a green LED if it is above the threshold value and turn on a red LED if it is below.

7.Repeat this process continuously to monitor the plant's environment and make adjustments as necessary.

Algorithm reading Soil water level and acidity and alkalinity.

1. Get the current soil moisture and pH readings

2. Check if the soil moisture reading is less than 20% or greater than 70%

3. If the soil moisture reading is less than 20%, output "Water the plant"

4. If the soil moisture reading is greater than 70%, output "Stop watering the plant"

5. If the soil moisture reading is between 20% and 70%, proceed to the next step

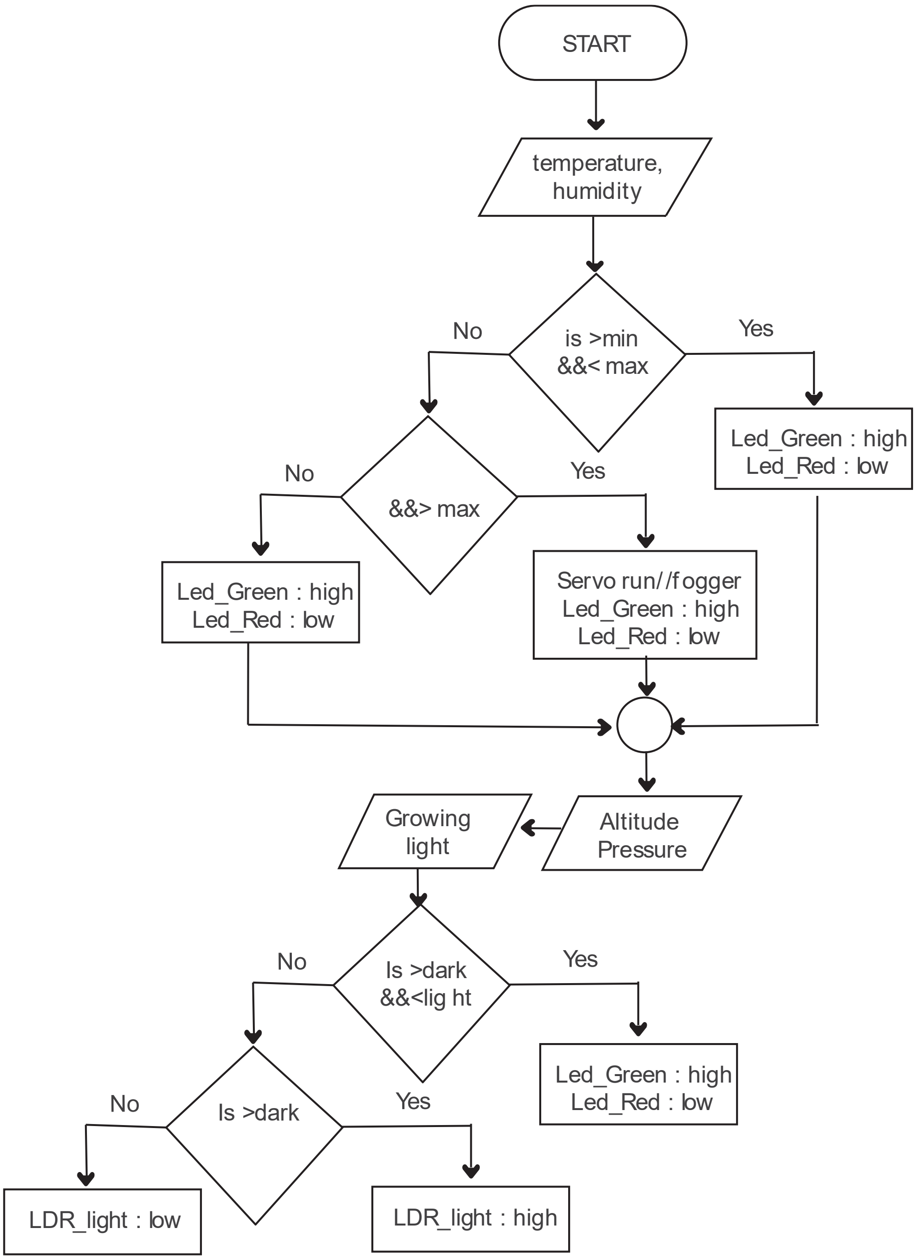
6. Check if the pH reading is less than 6 or greater than 7.5

7. If the pH reading is less than 6, output "Acidic fertilizer to the soil"

8. If the pH reading is greater than 7.5, output "Alkali to the soil"

9. If the pH reading is between 6 and 7.5, output "The soil moisture and pH are within the acceptable range"

4.3 Flowchart



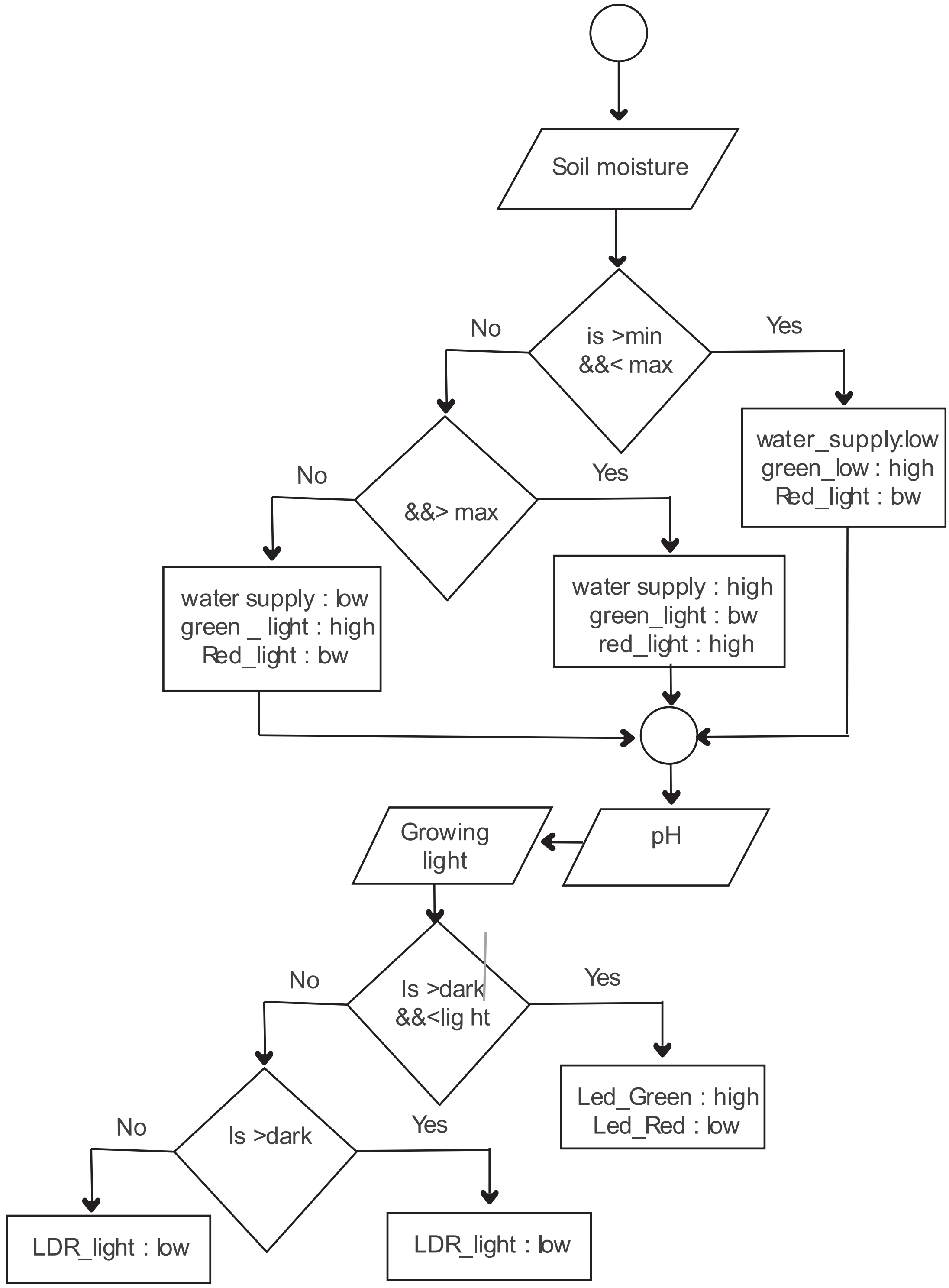


Figure 15: System Flowchart

CHAPTER 5

RESULT AND ANALYSIS

5.1 Completed Result

Output	**Output result**
Temperature	25^{o}
Humidity	29
Pressure	874.42hpa
Altitude	1225.84 meters
Growing light	618
pH	3.14
Soil moisture	97%

Table 2: Output result

5.2 Problem Encountered

We faced a lot of challenges and problems during the course of our project. Some of them are

- When attempting to connect Arduino and ESP, we faced a problem with serial communication. After some time, we discovered that using JSON could help us resolve this issue.
- Working with the hardware at once is time-consuming, and we often make mistakes in the wiring, such as connecting positive voltage to the ground and vice versa.
- Lack of sufficient knowledge about hardware results in frequent hardware failures.
- Working with fixed components limits our ability to make necessary changes, and adding new hardware components adds complexity to the system.

REFERENCES

Gajala Pathan1, S. K. (2019). IoT Based Real Time Greenhouse monitoring system using. *IRJET, 06*(04).

Haman2, F. S. (2011). *Potential Impacts of Improper irrigation.* Florida: UF/IFAS Extension, University of Florida.

Impact of improper design. (n.d.).

Miss. Aishwarya Kore1, D. D. (2019). IoT Based Real Time Greenhouse monitoring system using. *International Research Journal of Engineering and Technology (IRJET), 06*(04).

Mr. B.P. Kulkarni1, A. V. (2019). IoT based Greenhouse Monitoring using PIC16F877A. *International Research Journal of Engineering and Technology (IRJET), 06*(04).

Ravi Kishore Kodali, R. K. (2016). IoT based smart greenhouse. *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC).*

SAMBATH.M1, P. M. (2019). IoT BASED GARDEN MONITORING SYSTEM. *Journal of Physics: Conference Series, 1362*(012069).

Sharmila Agnal1, K. K. (2018). Automated Smart Greenhouse Environment Using IoT. *IRJET, 5*(10).

Wilfried Baudoin, R. N.-W. (2013). PROTECTED CULTIVATIONS. In *Good Agricultural Practices* (p. 145). Rome: FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS.

APPENDIX

Code

```
  temperature = dht.readTemperature(); // Read the temperature from the DHT sensor
 if (isnan(temperature)) { // If the temperature reading is not a number (indicating an error)
    Serial.println("Failed to read temperature from DHT sensor!"); // Print an error message
   return; // Exit the loop function
 }

 humidity = dht.readHumidity(); // Read the humidity from the DHT sensor
 if (isnan(humidity)) { // If the humidity reading is not a number (indicating an error)
   Serial.println("Failed to read humidity from DHT sensor!"); // Print an error message
   return; // Exit the loop function
 }

 // If the temperature or humidity is outside the acceptable range
 if ((temperature > MIN_TEMP || humidity > MIN_HUMIDITY ) && (temperature < MAX_TEMP || humidity < MAX_HUMIDITY) ) {

   digitalWrite(GREEN_LED_dht11, HIGH); // Turn on the green LED
   digitalWrite(RED_LED_dht11, LOW); // Turn off the red LED

 }
 else
 {
   if(temperature < MIN_TEMP || humidity < MIN_HUMIDITY )
   {
```

```
 digitalWrite(GREEN_LED_dht11, LOW); // Turn on the green LED
 digitalWrite(RED_LED_dht11, HIGH); // Turn off the red LED
    //fogger operation through servo motor 10 times
for(int i=0;i<=10;i++)
{
   for (int position = 0; position <= 180; position += 1) { // goes from 0 degrees to 180
degrees
   myservo.write(position);              // tell servo to go to position in variable 'position'
   delay(5);                       // waits 15ms for the servo to reach the position
     }
  for (int position = 180; position >= 0; position -= 1) { // goes from 180 degrees to 0
degrees
   myservo.write(position);              // tell servo to go to position in variable 'position'
   delay(5);                     // waits 15ms for the servo to reach the position
      }
}

   }
   else{
   digitalWrite(GREEN_LED_dht11, LOW); // Turn on the green LED
   digitalWrite(RED_LED_dht11, HIGH); // Turn off the red LED

   }
  }
  delay(2000);

  // Print the temperature and humidity readings to the serial monitor
  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.print("  Humidity: ");
  Serial.print(humidity);
  Serial.println();
```

```
  //Serial.print("end");
   // Delay for 2 seconds


   // bmp ***********************************************
 sensors_event_t event;
  bmp.getEvent(&event);

  if (event.pressure) {
    Serial.print("Pressure: ");
    Serial.print(event.pressure);
    Serial.print(" hPa");

    /*float temperature;
    bmp.getTemperature(&temperature);
    Serial.print(" Temperature: ");
    Serial.print(temperature);
    Serial.print("°C");
    */
    //float
     altitude = bmp.pressureToAltitude(1013.25, event.pressure, temperature);
    Serial.print(" Altitude: ");
    Serial.print(altitude);
    Serial.println(" meters");
  } else {
    Serial.println("Sensor error");
  }

  delay(1000);
//*******************************LDr
 //int
 ldrValue = analogRead(LDR_PIN); // Read the analog value of the LDR
 /* digitalWrite(LDr_LED_PIN,HIGH);
```

```
 delay(1000);*/

  if (ldrValue < dark_threshold && ldrValue > light_threshold) { // If the LDR value
is above the threshold
     digitalWrite(LDr_LED_PIN,LOW);
 }

 else {
   if(ldrValue > dark_threshold)
   {
  digitalWrite(LDr_LED_PIN,HIGH);

   }
   else{
     digitalWrite(LDr_LED_PIN,LOW);

   }
 }
   Serial.print("Growing light Reading: "); // Print a label for the LDR reading
   Serial.println(ldrValue); // Print the LDR reading
   delay(2000);
//Assign collected data to JSON Object
//******************************

  pH = sensor.readPH();
  Serial.print("pH: ");
  Serial.println(pH);

 if (pH > min_pH && pH <  max_pH ) {
   digitalWrite(alkaline,LOW);
   digitalWrite(acidic,LOW);

 }else{
```

```
  if(pH < min_pH)
  {
   digitalWrite(alkaline,HIGH);
   digitalWrite(acidic,LOW);
  }
  else{
   digitalWrite(alkaline,LOW);
   digitalWrite(acidic,HIGH);
  }
 }
 delay(1000);

 soilMoistureValue = analogRead(soilMoisturePin);

 // check if the soil moisture value is outside of the expected range (0-1023)
 if (soilMoistureValue < 0 || soilMoistureValue > 1023) {
  // print an error message if the value is outside of the expected range
  Serial.println("Error: soil moisture value is out of range!:");
 }
 else {
  // calculate soil moisture percentage
  int soilMoisturePercentage = map(soilMoistureValue, 0, 1023, 0, 100);

  // check if the soil moisture percentage is above the minimum threshold or below the maximum threshold
   if (soilMoisturePercentage > minThresholdSoilM && soilMoisturePercentage < maxThresholdSoilM) {
    // turn on the LED if the soil moisture percentage is outside of the acceptable range
    digitalWrite(ledPin_g_soilM, HIGH);
    digitalWrite(ledPin_r_soilM,LOW);
    digitalWrite(waterSupply,LOW);
  } else {
    // turn off the LED if the soil moisture percentage is within the acceptable range
```

```
    if( soilMoisturePercentage < minThresholdSoilM)
    {
     digitalWrite(ledPin_g_soilM, LOW);
     digitalWrite(ledPin_r_soilM,HIGH);
     digitalWrite(waterSupply,HIGH);
   }
   else{
     digitalWrite(ledPin_g_soilM, HIGH);
     digitalWrite(ledPin_r_soilM,LOW);
     digitalWrite(waterSupply,LOW);
   }
  }
  Serial.print("SoilMoistur Reading in percentage: ");
  Serial.println(soilMoisturePercentage);
  }
  delay(1000);
Serial.println("*******************************************************
********************************");


  doc["temperature"] = temperature;
  doc["humidity"] = humidity;
//  doc["events.pressure"]   = pressure;
  doc["altitude"] = altitude;
  doc["ldrvalue"] = ldrValue;
  doc["pH"]= pH;
  doc["soilMoistureValue"]= soilMoistureValue;

  //Send data to NodeMCU
  serializeJson(doc, nodemcu);
  char buffer[1024];
  serializeJson(doc,buffer);
  delay(1000);
```

```
  Serial.println(buffer);

}
```