Greenhouse Monitoring and Control System using IoT

A PROJECT REPORT

in partial fulfillment for the award of the degree
of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

INSTITUTE OF ENGINEERING AND TECHNOLOGY, LUCKNOW

(An Autonomous & Constituent Institute of A.K.T.U.)

Affiliated to

DR. A. P. J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW (U. P.), INDIA

Submitted by :-

Rudra Saksena (1805231045)

Deepankar Sen(1805231023)

Reema Maddheshia(1805231043)

Utkarsh Sharma(1805231060)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

INSTITUTE OF ENGINEERING AND TECHNOLOGY LUCKNOW
(An Autonomous & Constituent Institute of A.K.T.U.)

CERTIFICATE

This is to certify that **Rudra Saksena, Deepankar Sen, Reema Maddheshia, Utkarsh Sharma** have carried out the project work presented in this report entitled **"Greenhouse Monitoring and Control System using IoT"** for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** from **Dr. A. P. J. Abdul Kalam Technical University, Lucknow** under my supervision during the academic session 2021-22.

Er. Amit Kumar
(Project Guide)

ACKNOWLEDGEMENT

I would like to express my special thanks to my project guide Er. Amit Kumar who gave me the golden opportunity to do this wonderful project on the topic Greenhouse Monitoring and Control System using IoT, which also helped me in doing a lot of research and I came to know about so many new things I am really thankful to them.

Secondly I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

Rudra Saksena
Deepankar Sen
Reema Maddheshia
Utkarsh Sharma

ABSTRACT

The aim of this project is to design a greenhouse monitoring and controlling system based on the internet of things (IOT).A greenhouse is a covered area where plants grow and cultivate. It is also known as land of controlled crops and plants. Various greenhouse automation equipment like computer software and sensors are connected and used to collect data in the greenhouse environment to boost crop yields. This new innovative technology (IoT or the Internet of Things) makes use of numerous sensors linked to a central greenhouse environment control computer. The greenhouse sensor systems have elements that monitor and control temperature, humidity, soil moisture, illumination and read external weather conditions via a weather station. These are monitored and regulated using a microcontroller which is connected to sensors and external mechanical devices. Climate controlled greenhouses can help any grower get better at quality control or enhance crop yields, with advanced computer technology. There are other manual methods, but farmers who want to make the best of modern technology prefer to integrate the climate control system. This will help seasonal crops to grow throughout the year. A climate control system automates the greenhouse to reach the desired temperature as required by your crops' growing process. The system monitors and handles humidity, temperature, soil moisture and illumination. This is accomplished by the way of real time sensors, that communicate wirelessly in the greenhouse, via WiFi.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

According to the United Nations' Food and Agriculture Organization, food production must increase with 60 percent to be able to feed the growing population expected to hit 9 billion in 2050. The global population has grown from 1 billion in 1800 to 7 billion in 2012. It is expected to keep growing to reach to reach 11 billion by the end of the century. Modern farms can sprawl for hundreds of acres. Rising prices of fertilizer and electricity, combined with regulations limiting irrigation are placing increasing demands on farmers to more precisely utilize their resources. Reducing spoilage and food waste will require both better in-field monitoring as well as better monitoring and management within the field-to-shelf supply chain. It is a world where deadline pressures, a lack of information and conquering the challenges of time and distance confront individuals on a daily basis. Agriculture has been a leader for years in automation---many industrial farms rely on harvesters guided by GPS. It is also an industry starving for more data. Fluctuations in rainfall or market prices can cause profits to quickly rise or plummet. Obtaining accurate, ongoing data on operations has historically also been a challenge. Unlike cars or microprocessors, you can't mass produce identical tomatoes. Companies like CleanGlow and Solum have begun to bring Big Data to the field with tools that can dynamically calibrate moisture and other metrics. Between efforts to eat more food grown locally, a younger generation of farmers and cheaper component farming is getting an infusion of data and technology. As the concept of the 'Internet of Things' becomes increasingly prevalent, many systems are being devised to allow all manner of data to be gathered and analysed, and devices controlled via wireless data networks. 10 connected devices such as smart thermostats and lighting systems are making their way into homes, but another big opportunity for the Internet of Things could be outdoors, in the area of agriculture. The crop agriculture in greenhouse is higher affected by the surrounding conditions. The significant environmental factors for the quality and better productivity of the plants growth are temperature, relative humidity, Lighting, moisture of soil, and the CO2 amount in greenhouse. Continuous monitoring of these factors gives relevant information pertaining to the individual effects of the various factors towards obtaining maximum crop production.

1.1 Problem Definition

Complexity involved in monitoring climatic parameters like humidity, soil moisture, illumination, temperature, etc. which directly or indirectly govern the plant growth. Places like Punjab, which receive ample amount of water through river and canal irrigation system, faces problem of soil salinity due to excess irrigation. Places with limited water supply like Rajasthan, faces problem of acute water shortage for agriculture. Excessive use of fertilizers, insecticides and pesticides makes the soil dependent on them, erodes fertility, increases resistance in insects and pests, pollutes ground water and nearby water bodies whenever it rains. 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 modern proposed systems use the mobile technology as the communication schemes and wireless data acquisition systems, providing global access to the information about one’s farms. Keeping these issues in view, an IOT based monitoring and control system is designed to find implementation in the the near future that will help Indian farmers.

1.2 Objective

The project is primarily aimed at improving current agricultural practices by providing a smart and automatic system.
To always sustain a suitable climate inside the greenhouse, conditions like temperature, humidity, soil moisture and illumination are monitored using sensors and regulated using mechanical devices.
Moreover, real time data is constantly communicated to the cloud and then to the webpage for display and analysis purposes.

1.3 Dependencies

Hardware

•Arduino Uno
•Soil moisture sensors
•DHT11 (Temperature and Humidity) sensor
•Light dependent resistor
•DC Pump
•DC Fan
•Bulb

Software

•Arduino Programming
•Android Studio Programming

2. LITERATURE REVIEW

Although India receives ample amount of precipitation and have many large river systems but still only one third of the total agricultural land is connected via canal irrigation system. Remaining majority of the portion is dependent on monsoon or tube wells. Places with excess water faces. Problem of land sanity due to over irrigation and water logging. Water collected on the surface also blocks pores in the soil and kill beneficial micro organisms.

Alternatively, places with limited supply of water cannot do irrigation throughout the growing season because the requirement of water often exceeds the water supply due to conventional type of irrigation like sprinkler or in case allowing the water to just irrigate the field directly from water drainage channels. Effects of excessive and irregular irrigation on soil are as follows:

- Increases salinity
- Water logging
- Hindrance in air communication to plant roots
- Reduction in temperature of soil
- Land becomes marshy
- More nitrate formation in soil
- Acidity of soil

Hence, problem lies in the mismanaged use of water. For optimum use of water, we use drip irrigation. It is an irrigation method to save water by allowing water to target the roots of plant. Water obtained from all the sources like canal, rainwater harvesting, tube well etc. are not allowed to irrigate the fields directly, instead it is first stored into an underground tank. Tank is equipped with an ultrasonic sensor which measures the level of water continuously and alerts the user with a sms whenever water level falls below the threshold mark. 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 temperature and humidity sensor inside the smart greenhouse to measure humidity and temperature. When temperature rises above a certain level, micro controller will trigger relay attached to the foggier, 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 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 green house with plant growing lights which will turn on whenever the reading from LDR sensor falls below cut-off value. To avoid involvement of middleman and their adverse effects on farmers we proposed an IOT based solution, to inform the buyers(agencies) about the goods produces by a farmer. The farmer just have to swipe his authorized RFID card and then automatically it will send an e-mail to the buyer-sharing the information of quantity of goods produced at that instant of time.

3. METHODOLOGY

3.1 Block Diagram

The basic block diagram of greenhouse system is as shown in figure 3.1.

The system is a greenhouse system in which there are four sensors. These sensors act as input to the micro controller system. The input feed provided to the micro controller is in the form of analog data. This data is converted by the controller into digital format. The data is shown on the LCD display and also on the android phone via Wifi. Thus the monitoring of temperature, moisture and other parameters is done automatically. Once the parameter values are monitored they can be controlled by the embedded system which is built with coding. This is automating controlling system. The android phone is operated by the user. The android application is used for controlling as per the user knowledge and required output.

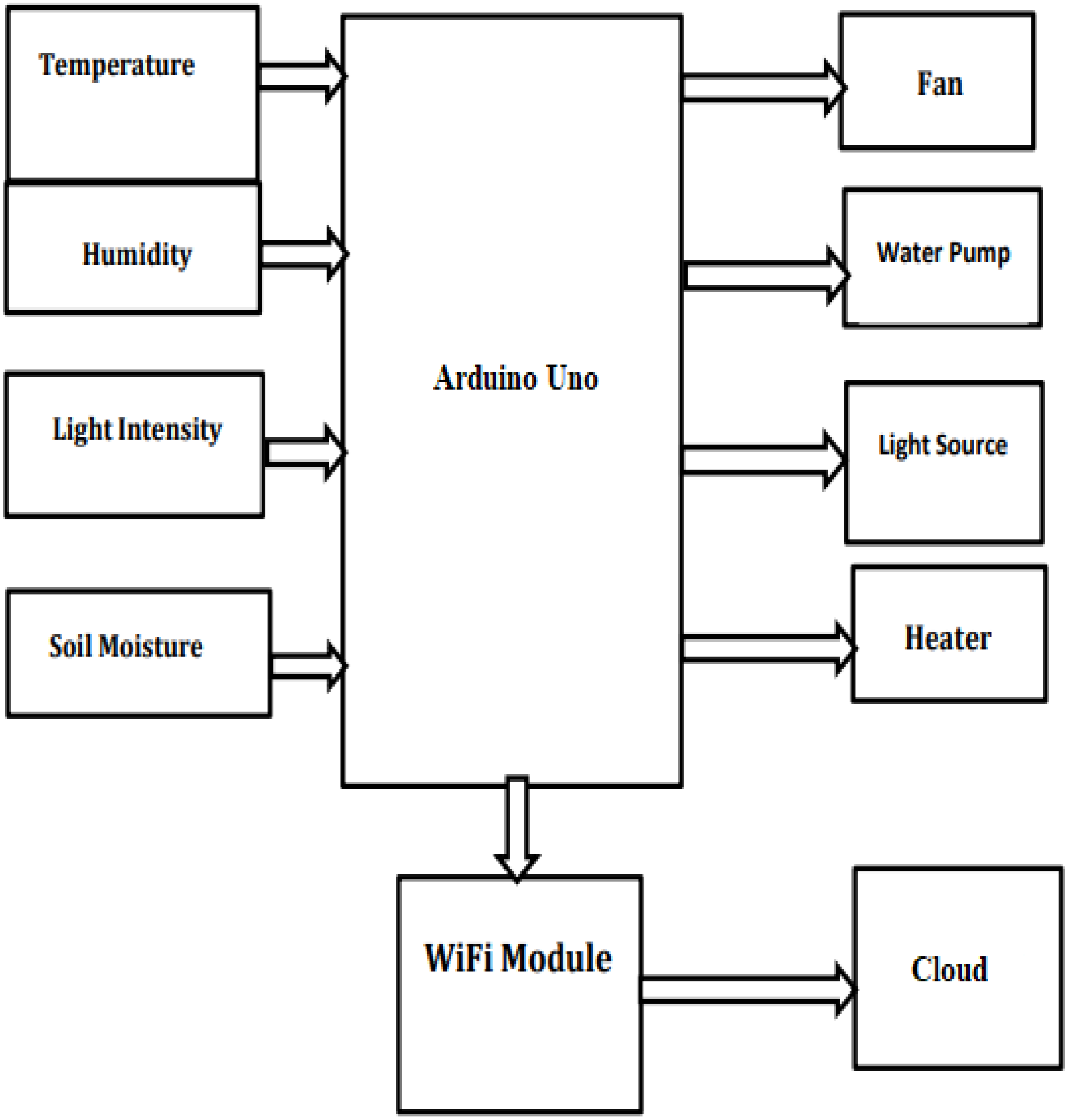


Figure 3.1: Block diagram

3.2 Flow Chart

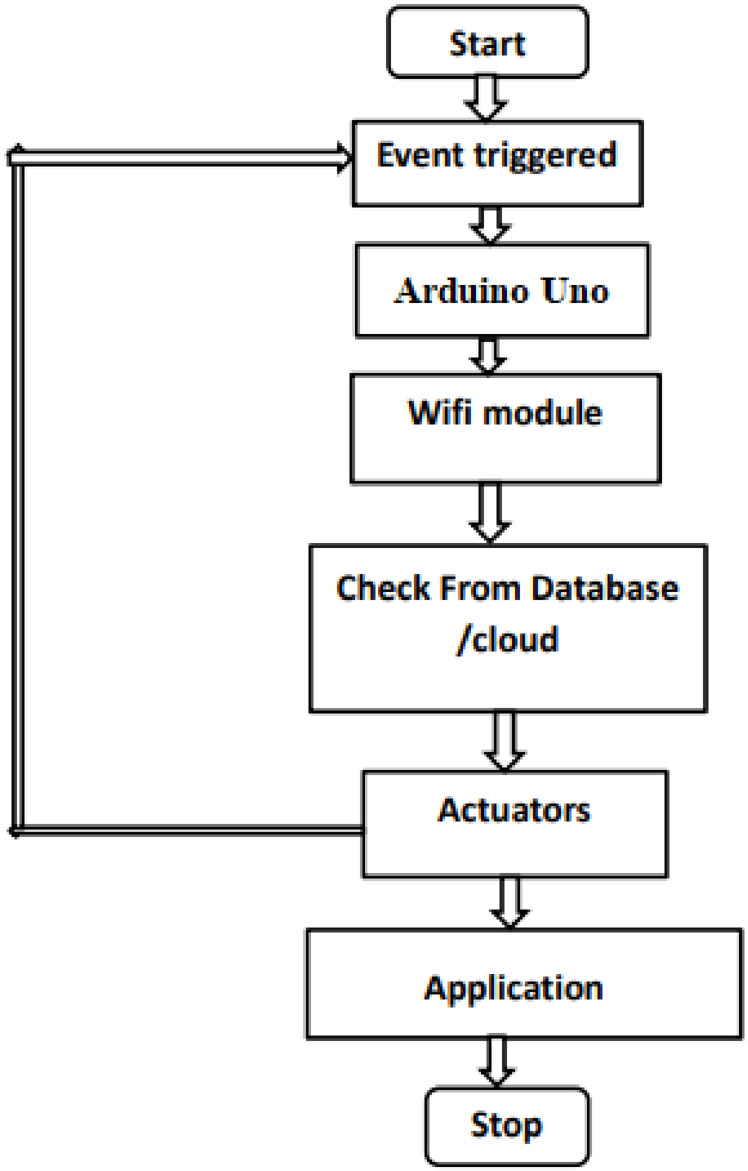


Figure 3.2: Flow Chart

3.3 Project Execution Steps

- The IOT greenhouse monitoring system employs a PC or phone base system for keeping the owner continuously informed.

- This is a micro controller-based circuit which monitors and records the values of temperature, humidity, soil moisture and sunlight of the natural environment that are continuously updated as a log in order to optimize them to achieve maximum plant growth and yield.

- The system constantly monitors the digitized parameters of the various sensors. Monitoring and controlling of a greenhouse environment involves sensing the changes occurring inside it which can influence the rate of growth in plants.

- The important parameters are the temperature inside the greenhouse which affects the photosynthetic and transpiration process, humidity, moisture content in the soil, the illumination etc

3.4 Interfacing Sensors With Arduino

The proposed system uses four sensors to collect the data of the parameters required to monitor the greenhouse. The important parameters are the temperature inside the greenhouse which affects the photosynthesis and transpiration process, humidity, moisture content in the soil, the illumination etc.

The system consists of four subsystems in it and they are:

- Temperature monitoring and control system
- Light intensity monitoring and control system
- Humidity level monitor and control system
- Soil moisture monitoring and control system

The system's temperature monitor and control system works according to the temperature value set by the user. First it gets the value from the user and maintains the temperature on the LED screen for user reference. The temperature of the greenhouse is reduced by the fan that is placed inside the greenhouse.The temperature of green house is increased by using heater which is placed at the floor of the greenhouse to ensure that the whole greenhouse is warmed equally.

The light control system controls the light falling on the greenhouse. When there is not enough light the LDR detects this, and the light bulbs are switched ON.When there is lighter the light bulbs are turned OFF. But at night the system will automatically get turned ON and this has harmful effects on the plants therefore a manual option is available to OFF the lights at times when we feel the lights are unnecessary.

The humidity level monitoring system monitors and it maintains around a predefined value. When the system detects the drop in humidity level with the help of soil moisture sensor the pipelines installed inside the greenhouse allows water to floor. The pipes contain small hoes and hence allow water to reach the soil quickly. When the humidity level reaches the desired value and detected by soil moisture sensor, the system OFF the pump and flow of water stops.

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, mac OS, Linux) that is written in functions from C and C++.It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards.
Arduino programs can be divided in three main parts: Structure, Values (variables and constants), and Functions.
Software structure consist of two main functions −

Setup( ) function
Loop( ) function

The setup() function is called when a sketch starts. Use it to initialize the variables, pin modes, start using libraries, etc. The setup function will only run once, after each power up or reset of the Arduino board.

Syntax:

```
Void setup ( )
{
}
```

After creating a **setup()** function, which initializes and sets the initial values, the **loop()** function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

Syntax:

```
Void Loop ( )
{
}
```

Digital I/O

pinMode()

Configures the specified pin to behave either as an input or an output. See the Digital Pins page for details on the functionality of the pins.
Syntax:
pinMode(pin, mode)
Parameters: pin- the Arduino pin number to set the mode of.

digitalRead()

Reads the value from a specified digital pin, either High or low
Syntax:
digitalRead(pin) , returns High or Low

Parameters: pin- the Arduino pin number you want to read

digitalWrite()
\

Write a HIGH or a LOW value to a digital pin.
If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

Syntax:
digitalWrite(pin, value)
Parameters: pin- the Arduino pin number.
value- HIGH or LOW.

Analog I/O

analogRead()

Reads the value from the specified analog pin. Arduino boards contain a multichannel, 10-bit analog to digital converter. This means that it will map input voltages between 0 and the operating voltage(5V or 3.3V) into integer values between 0 and 1023

Syntax:

analogRead(pin)

Parameters: pin- the name of the analog input pin to read from (A0 to A5)

analogWrite()

Writes an analog value to a pin. Can be used to light a LED at varying brightnesses or drive a motor at various speeds

Syntax:

analogWrite(pin, value)

Parameters: pin- the Arduino pin to write to. Allowed data types: int
value: between 0 (always off) and 255 (always on).
Allowed data types: int

Important Time function

delay()
Pauses the program for the amount of time (in milliseconds) specified as parameter. (There are 1000 milliseconds in a second.)
Syntax:
delay(ms)

3.5 Sensors

3.5.1 Temperature and Humidity Sensor

We place temperature and humidity sensor inside the smart greenhouse to measure humidity and temperature. If the temperature exceeds beyond the limit set then a fan will be automatically switched ON as a coolant to reduce the temperature. When it reaches the desired temperature the fan will be switched OFF automatically. But if the temperature decreases below the optimum temperature a bulb as a heater will be switched ON to set the temperature within the desired range.

Figure 3.3: Temperature and humidity sensor

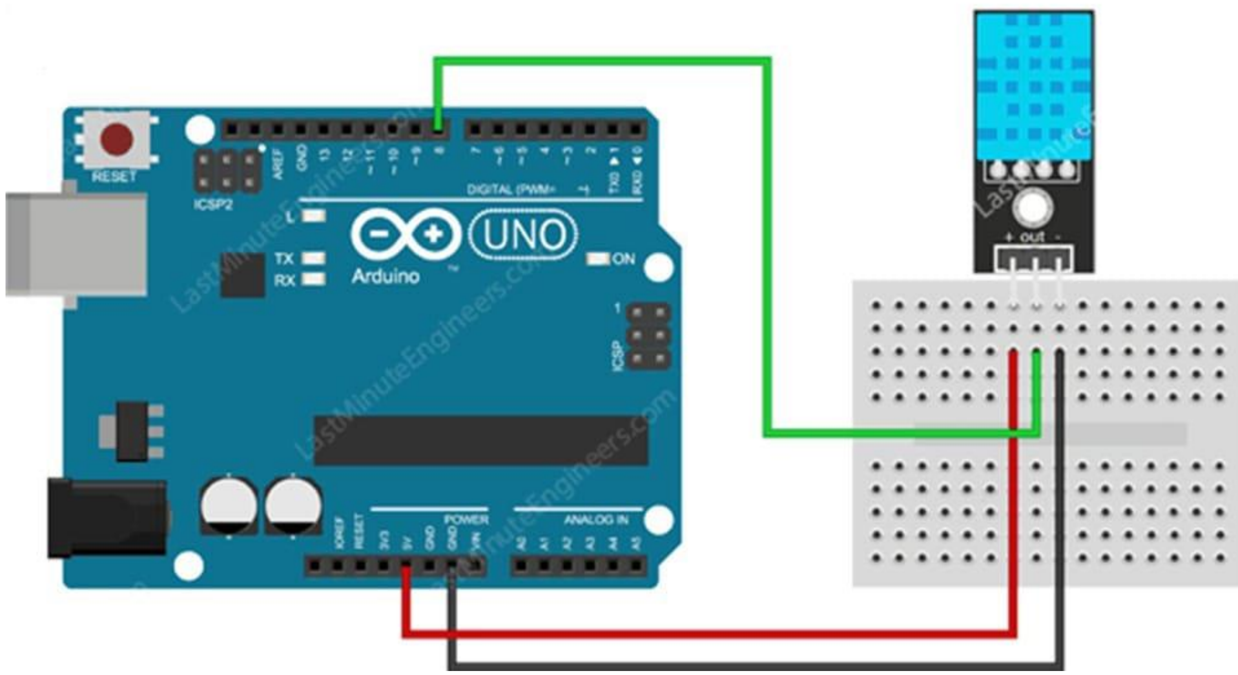
Humidity sensor is used for sensing the vapours in the air. The change in RH (Relative Humidity) of the surroundings would result in display of values. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin(no analog input pins needed). Its fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 second

FEATURES

- Full range temperature compensated
- Relative humidity and temperature measurement
- Calibrated digital signal
- Outstanding long-term stability
- Extra components not needed
- Long transmission distance
- Low power consumption
- 4 pins packaged and fully interchangeable

DETAILS:

This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high performance 8-bit micro controller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness. Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor’s internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and upto 20 meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. Communication Process: Serial Interface (Single-Wire Two-Way) The interesting thing in this module is the protocol that uses to transfer data. All the sensor readings are sent using a single wire bus which reduces the cost and extends the distance. In order to send data over a bus you have to describe the way the data will be transferred, so that transmitter and receiver can understand what says each other. This is what a protocol does. It describes the way the data are transmitted. On DHT-11 the 1-wire data bus is pulled up with a resistor to VCC. So if nothing is occured the voltage on the bus is equal to VCC.



DHT11 Code :

```
#include <dht11.h>
#define DHT11PIN 4

dht11 DHT11;

void setup()
{
  Serial.begin(9600);

}
```

```
void loop()
{
  Serial.println();

  int chk = DHT11.read(DHT11PIN);

  Serial.print("Humidity (%): ");
  Serial.println((float)DHT11.humidity, 2);

  Serial.print("Temperature (C): ");
  Serial.println((float)DHT11.temperature, 2);
  delay(2000);

}
```

3.5.2 Soil Moisture Sensor

The two copper leads act as the sensor probes. They are immersed into the specimen soil whose moisture content is under test. The conductivity of soil depends upon the amount of moisture present in it. It increases with increase in the water content of the soil that forms a conductive path between two sensor probes leading to a close path to allow current flowing through.

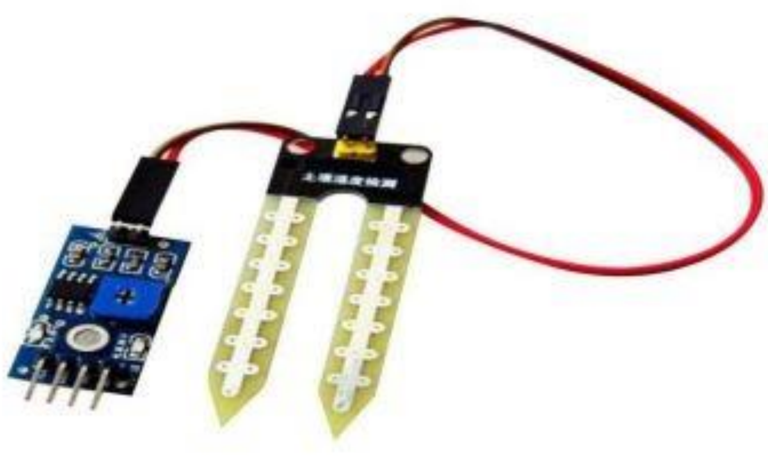
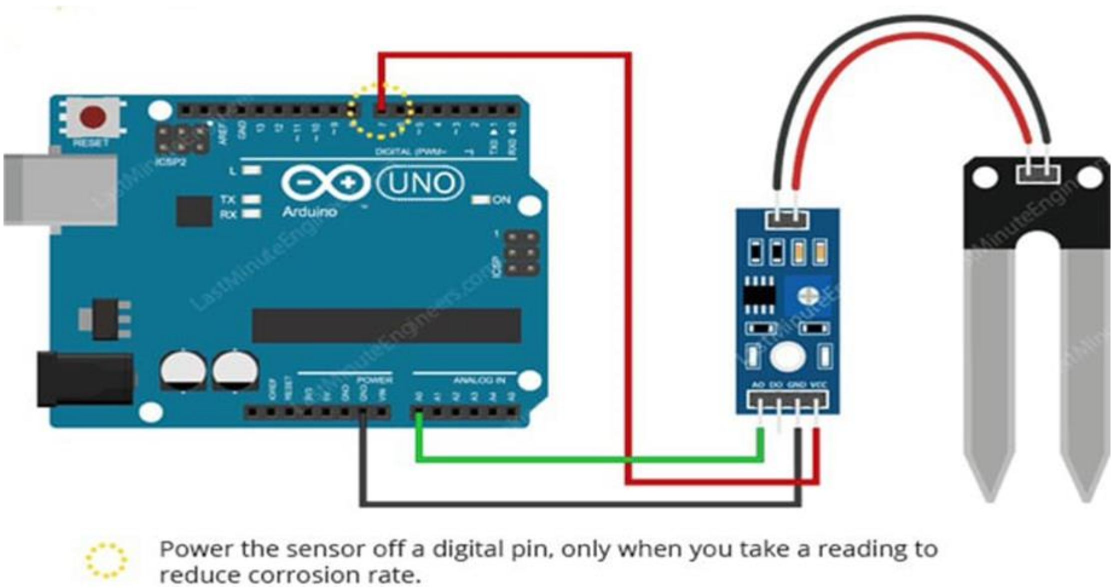


Figure 3.4: Soil Moisture Sensor



Soil Moisture Code:

```
int WET= 3;
int DRY= 2;
int Sensor= 0; // Soil Sensor input at Analog PIN A0
int value= 0;

void setup() {
  Serial.begin(9600);
  pinMode(WET, OUTPUT);
  pinMode(DRY, OUTPUT);
  delay(2000);
}

void loop() {
  Serial.print("MOISTURE LEVEL : ");
  value= analogRead(Sensor);
  value= value/10;
  Serial.println(value);
  if(value<50)
  {
```

```
    digitalWrite(WET, HIGH);
  }
  else
  {
    digitalWrite(DRY,HIGH);
  }
  delay(1000);
  digitalWrite(WET,LOW);
  digitalWrite(DRY, LOW);
}
```

3.5.3 LDR: Light Dependent Resistor

The light dependant resistor is an electronic component whose resistance decreases with increasing light intensity. It is also called as “Photo Resistor” or “Photo conductor”. The light dependant resistor uses high resistance semiconductor material. When light falls on such a semiconductor the bound electrons [ie., Valence electrons] get the light energy from the incident photos. Due to this additional energy, these electrons become free and jump in to the conduction band. The electron - hole pairs are generated. Due to these charge carriers, the conductivity of the device increases, decreasing its resistivity. The way an LDR works is that they are made of many semi-conductive materials with high resistance. The reason they have a high resistance is that there are very few electrons that are free and able to move because they are held in a crystal lattice and are unable to move.

Figure 3.5: Light Dependent Resistor

When light falls on the semi-conductive material it absorbs the light photons and the energy is interfered to the electrons, which allow them to break free from the crystal lattice and conduct electricity and lower the resistance of the LDR.

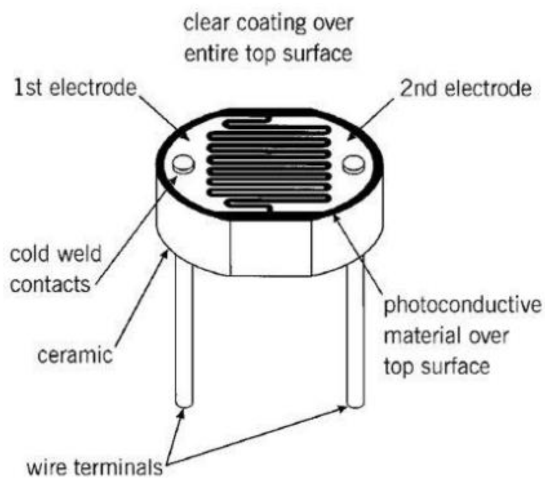


Figure 3.6: Internal structure of LDR

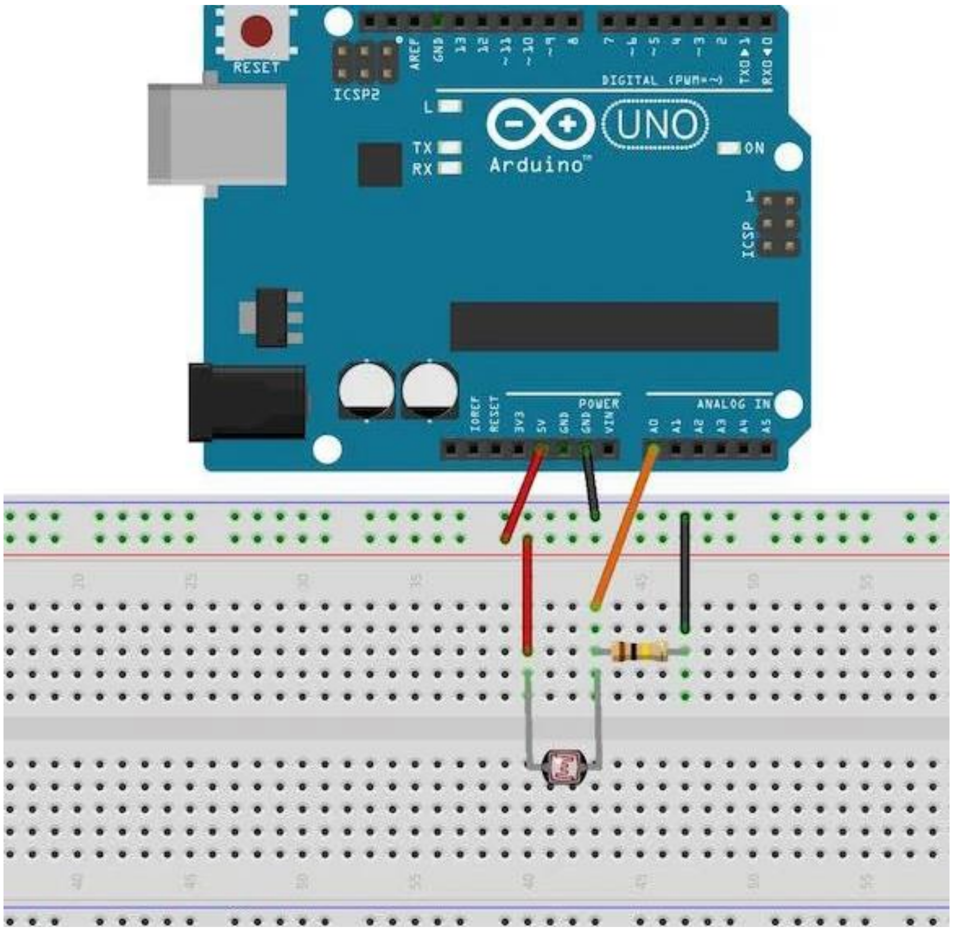
Light dependent registers have many uses, many of the uses have to do with objects that have to work in certain levels of light. Some of the uses of LDR are in photographic light meters, streetlights and various alarms light burglar alarms, re alarms and smoke alarms.

Analog applications of LDR

- Camera Exposure Control
- Auto Slide Focus – dual cell
- Photocopy Machines – density of toner
- Calorimetric Test Equipment
- Densitometer
- Electronic Scales – dual cell

Digital applications of LDR

- Automatic Headlight Dimmer
- Night Light Control
- Oil Burner Flame Out
- Street Light Control
- Position Sensor



LDR Coding

```
const int LDR = A0;
int input_val = 0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  input_val = analogRead(LDR);
  Serial.print("LDR Value is: ");
  Serial.println(input_val);
  delay(1000);
}
```

3.6 Sending Sensor Data To Cloud Via WiFi Module

The next step in implementing is interfacing of wifi module with the Arduino. This is done to access the sensed data from the sensors to the cloud. ESP8266 wifi module is low cost standalone wireless transceiver that can be used for end-point IoT developments. ESP8266 WiFi module enables internet connectivity to embedded applications. It uses TCP/UDP communication protocol to connect with server/client. Microcontroller communicates with the module using a set of AT commands. Microcontroller communicates with ESP8266-01 WiFi module using UART having specified Baud rate. ThingSpeak is a software that can monitor our data over the internet from anywhere, and we can also control our system over the Internet from anywhere, and we can also control our system over the Internet, using the Channels and webpages provided by ThingSpeak. ThingSpeak 'Collects' the data from the sensors, 'Analyze and Visualize' the data and 'Acts' by triggering a reaction. Working of this project is based on serial communication for fetching data from the sensors. First Arduino sends a start signal to sensor and then it gives a response signal with containing data. Arduino collects and extracts the data and then sends it to ThingSpeak server. ThingSpeak displays the data in the form of graphs.

3.6.1 Relay Module

The four-channel relay module contains four 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. There are two terminal blocks with six terminals each, and each block is shared by two relays. The terminals are screw type, which makes connections to mains wiring easy and changeable.

The four relays on the module are rated for 5V, which means the relay is activated when there is approximately 5V across the coil. The contacts on each relay are specified for 250VAC and 30VDC and 10A in each case, as marked on the body of the relays.

- It is 4 Channel Isolated 5V 10A Relay Module, A wide range of microcontrollers such as Arduino, AVR, PIC, ARM and so on can control it.
- It is also able to control various appliances and other types of equipment with large current.
- Relay output maximum contact is AC250V 10A and DC5V 10A.
- They are commonly used in order to drive different kinds of loads. Like DC Pump or Fan with Microcontrollers (like Arduino, PIC Microcontroller or 8051 Microcontroller etc.)

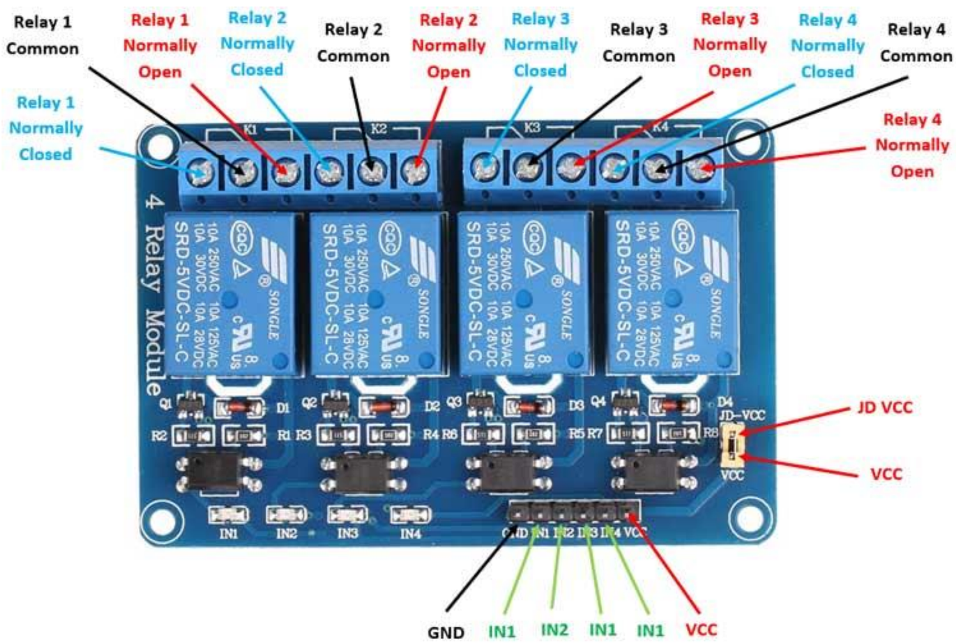


Figure 3.7: Relay Module

How To Use The Four-Channel Relay Module

The four-channel can be used to switch multiple loads at the same time since there are four relays on the same module. This is useful in creating a central hub from where multiple remote loads can be powered. It is useful for tasks like home automation where the module can be placed in the main switchboard and can be connected to loads in other parts of the house and can be controlled from a central location using a microcontroller.

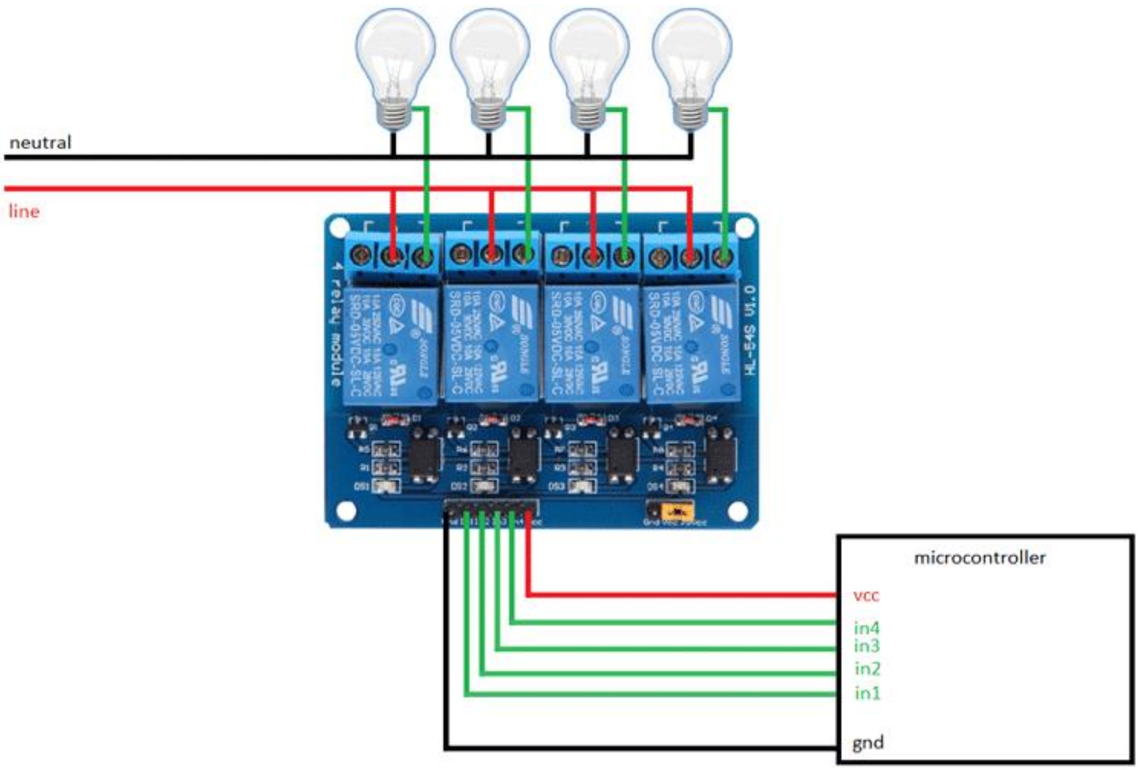
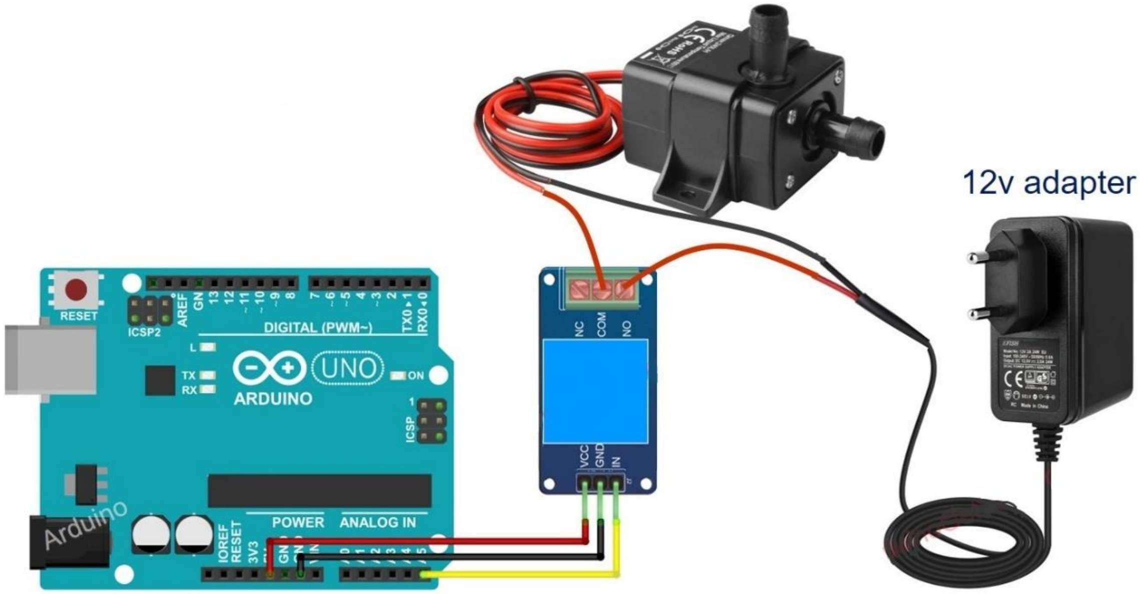


Figure 3.8: Relay module working

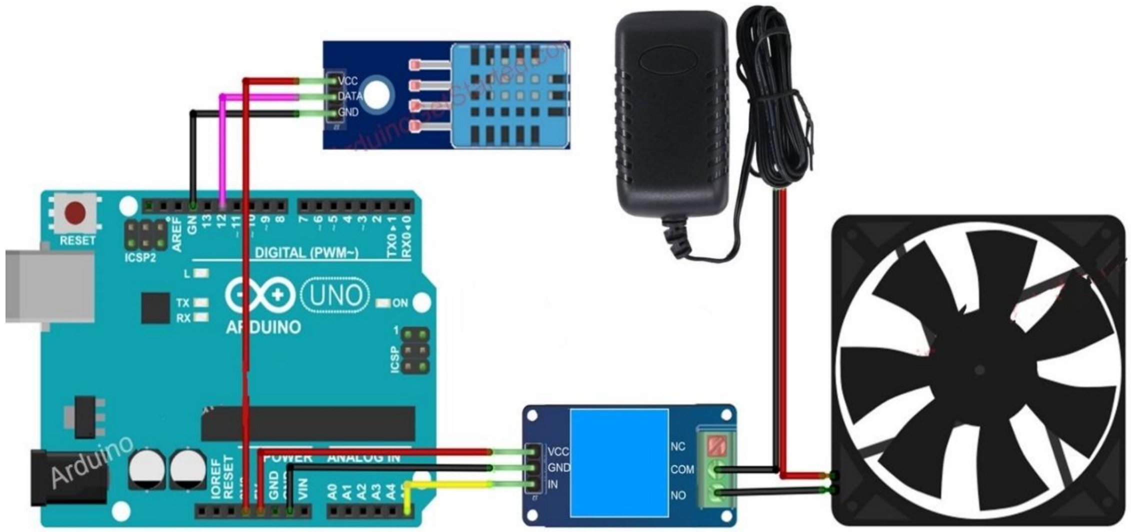
Controlling Pump with relay



Code:

```
const int RELAY_PIN = A5
void setup() {
pinMode(RELAY_PIN, OUTPUT);
}
void loop() {
  digitalWrite(RELAY_PIN, HIGH);
delay(5000);
  digitalWrite(RELAY_PIN, LOW);
  delay(5000);
}
```

Controlling Fan with DHT11 Sensor Output



Code:

```
#include "DHT.h"
#define RELAY_FAN_PIN A5
#define DHTPIN 12
#define DHTTYPE DHT11
const int TEMP_THRESHOLD_UPPER = 25;
const int TEMP_THRESHOLD_LOWER = 20;
DHT dht(DHTPIN, DHTTYPE);
float temperature;
void setup()
{

Serial.begin(9600); // initialize serial
dht.begin();        // initialize the sensor
pinMode(RELAY_FAN_PIN, OUTPUT);
}
void loop()
{
delay(2000);
temperature = dht.readTemperature();

if (isnan(temperature))
```

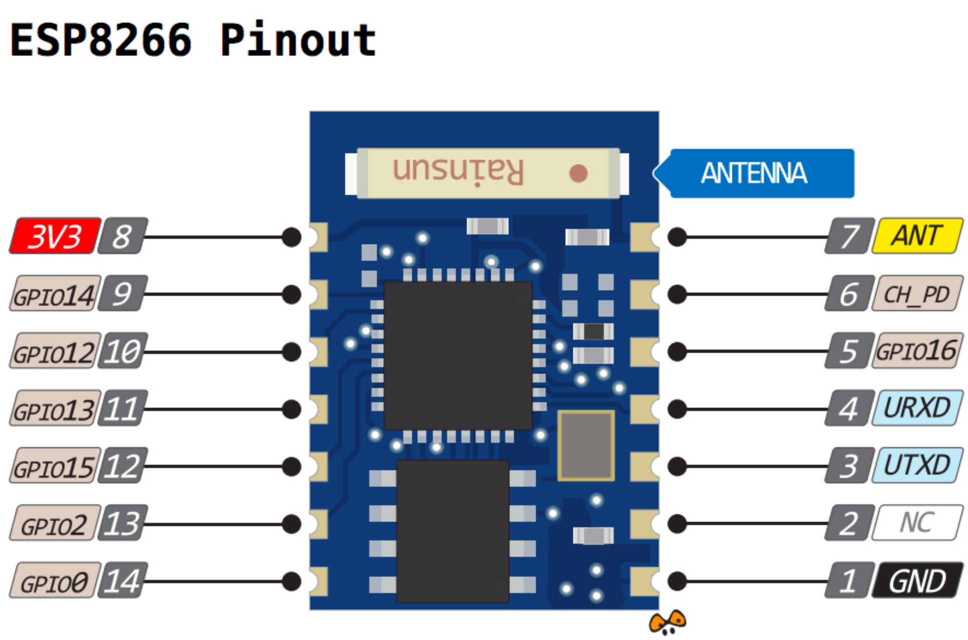
```
{
  Serial.println("Failed to read from DHT sensor!");
 }
else
 {
  if(temperature > TEMP_THRESHOLD_UPPER){
    Serial.println("The fan is turned on");
    digitalWrite(RELAY_FAN_PIN, HIGH);
  }
  else if(temperature < TEMP_THRESHOLD_LOWER)
 {
    Serial.println("The fan is turned off");
    digitalWrite(RELAY_FAN_PIN, LOW);
  }
  }
  }
```

3.7 Receiving Alerts On Mobile On Status Of Green House Parameters Via IoT

The ThingSpeak Cloud software is being used to get the status of various greenhouse parameters on the mobile phone. Android application known as ThingsView is installed on the mobile to get the alerts on the phone. The sensor data can be collectively viewed in the form of graphs in the ThingView application.

ESP8266 Wifi Module

ESP8266 is a low-cost WiFi module that belongs to ESP's family which you can use it to control your electronics projects anywhere in the world. It has an in-built microcontroller and a 1MB flash allowing it to connect to a WiFi. The TCP/IP protocol stack allows the module to communicate with WiFi signals. The maximum working voltage of the module is 3.3v so you cant supply 5v as it will fry the module.
ESP8266 is connected to the nearby WiFi hotspot that allows it to access the internet and sends data to the Blynk server along with authentication code. An authentication code is then sent to the app that has the same authentication code and then, the app receives the data to create a secure connection between the app and the ESP8266.



pin 1 _____ Rx -- connect it to Rx of Arduino

pin 2_____GPIO 0--connect it to ground while uploading the code to arduino IDE

pin 3_____GPIO 2,

pin 4_____ GND--connect it to ground

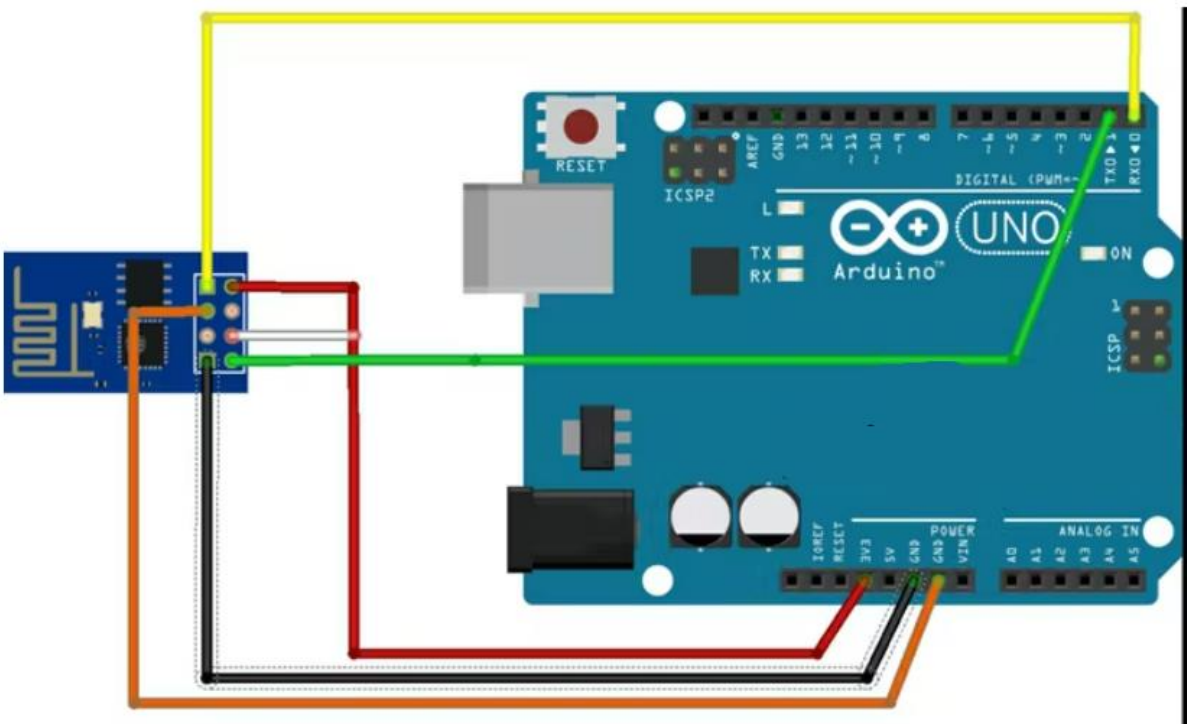
pin 5_____Tx--connect it to Tx of Arduino

pin 6_____CH_PD(EN) -- connect it to 3.3v pin 7_____RST(reset)--(not necessary) connect it to 3.3v for normal operation

and 0v(GND) for reset

pin 8____Vcc--supply 3.3v from Arduino

Schematics and connection



Outline of App development for this project

The basic idea is either to connect Ardiuno with Google Firebase Cloud System (a Google provided cloud support system) and then connect the application with the cloud. Or to direct interface the Ardiuno wifi module with the application for providing direct access support. In either cases the idea to collect information from Ardiuno with the help of different sensors connected to it.

Then that data is used to plot charts and graphs to provide insights about the temperature, soil moisture, humidity and light conditions in the greenhouse. Also using predefined threshold values, we determine if any of the controlling devices such as pump, light bulb etc. Also using the app we can override the controlling devices and control on/off operations using simple slider buttons.

Framework and technology used:

1. Android Studio for .zip, .jar and .apk file creation and updating
2. Google Firebase for authentication and database support

Languages Used:

4. Java for development and integration
5. XML for frontend development
6. Maven for adding gradle build dependencies and new libraries

Basic about the app to developed

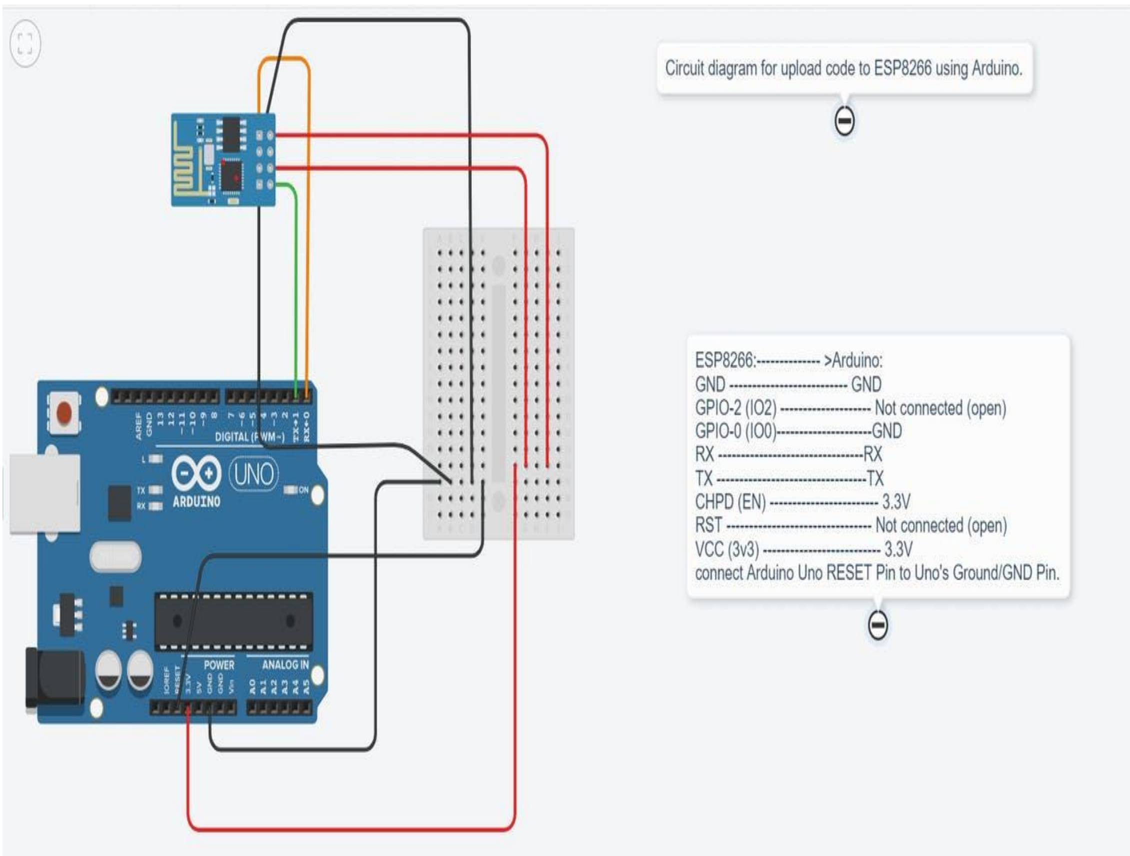
1. The given app is to developed on minimum SDK of API 19: Android 4.4(KitKat) and can be used on 99.4% of all android mobile phones.
Built language is java
2. AndroidX constraint Layout is used to allow full functionality on any screen size and dimension ratio.
3. The app is to connected with a new firebase project by adding dependencies in the gradle file and creating a new project on Google Firebase platform.
4. We use USBserial library to connect the arduino directly with a single android device, and provide overiding facilities for controlling devices connected to the Arduino module.

How to connect firebase to the arduino

1. We use ESP8266, a Wi-Fi microchip, with built-in TCP/IP networking software and microcontroller capability. First we install ESP8266 software on the Arduino IDE.
2. Then we add Arduino library(.ZIP file) to the IDE and then add ArduinoJSON library. ArduinoJSON is used to used to send data to firebase cloud in JSON file format. (We can use XML or HTML file format too).
3. Then we type-in firebase_HOST and firebase_AUTH to provide wifi-ssid and password.
4. After this we make a POJO file to define the format in which data is to be sent to Firebase.

5. We finally make the required circuit connections.

Circuit Connections to connect ESP8266 with Arduino



Connecting Android App with Firebase

1. Create a new project on Firebase
2. Register your application on Firebase Console by providing Android Package name.
3. Add firebase configuration by adding google-services.json on your app. Next in root-level (project-level) Gradle file (build.gradle), add rules to include the Google Services Gradle plugin.
4. Using the Firebase BoM, declare the dependencies for the Firebase Products that you want to

use in your app. Declare them in your module (app-level) Gradle file (usually app/build.gradle).
5. Here we use analytics(data-base) and authentication services to we need to add the respective library dependencies with up-to-date version using ‘implements’ command.
6. Your app and arduino is now connected to the same cloud database.

Plotting graph and providing UI

The data collected by Arduino is sent to database via ESP8266 wifi module. Then that data is sent to android app where it is plotted in real time.
We make eventListeners that detect change in synchronously incoming data, and plot the graph accordingly. We also check the incoming data(i.e. Last updated value of data) against predefined threshold values.
If the latest value crosses the predefined value we send signal to start or stop device connected to arduino.

3.8 Hardware interfacing with Arduino Uno

It is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header 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.

Figure 3.9: Arduino Uno

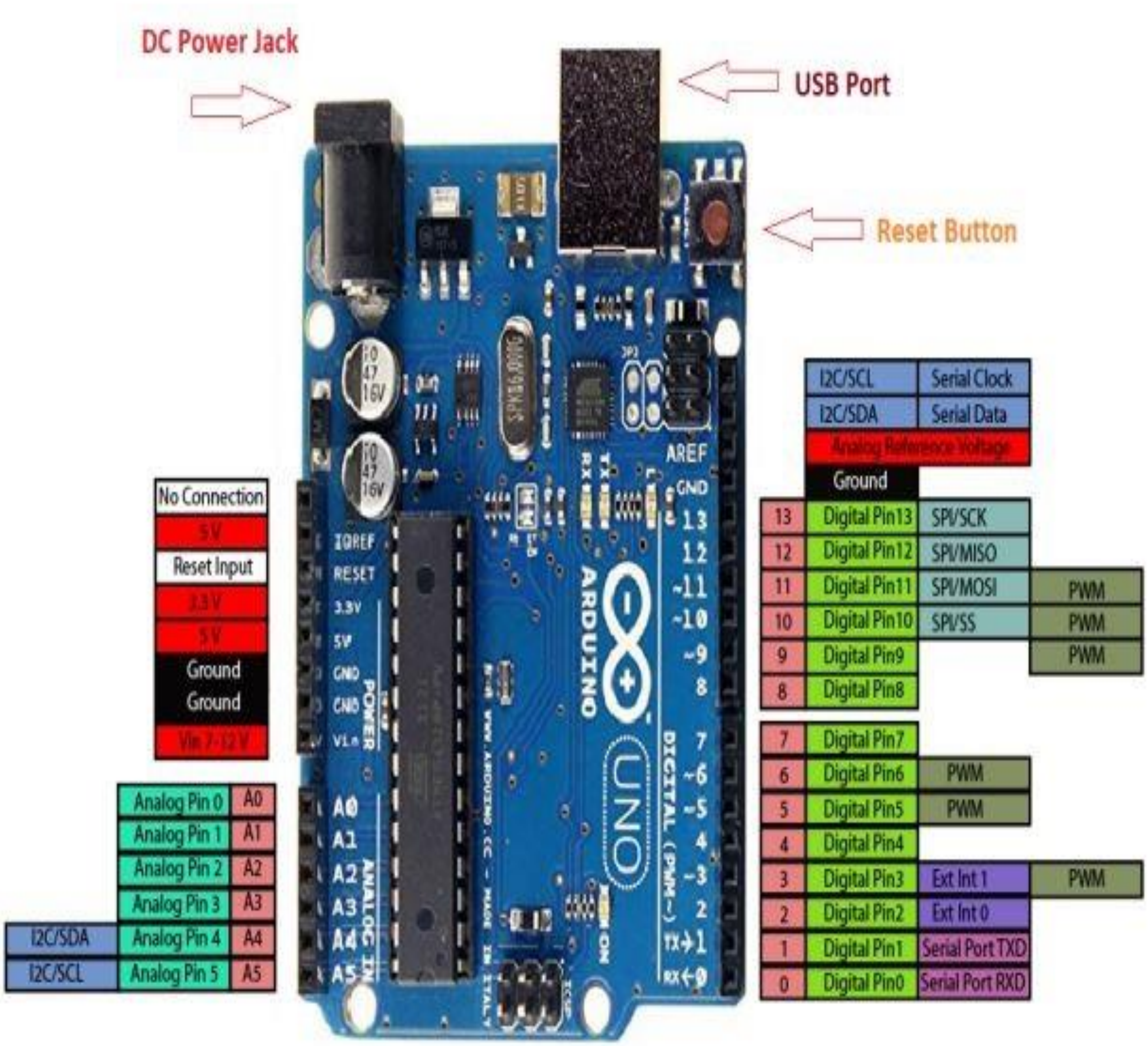


Figure 3.10: Arduino Uno internal structure pin diagram

4. RESULTS AND DISCUSSION

The sensing part of all the greenhouse parameters is being measured by the use of appropriate sensors. The sensed data is being stored in the database by the use of ThingSpeak software and also a comparative analysis is done on the parameters based on the sensor data. Mobile alerts are also sent to the concerned people from time to time so that they keep updated on the real time greenhouse environment. The name Greenhouse Monitoring and Control System using IoT is to both sense and monitor the environmental parameters from the sensor and also to stabilize the conditions if the conditions exceed the threshold. Actuators are used to control the parameters based on the sensor input. It can be done in both automatic and manual mode. In the manual mode, the actuators are controlled by the user based on the inputs obtained through SMS which is not implemented in our project. In the automatic mode, based on the database of the previous event the actuators are being controlled.

5. CONCLUSION AND FUTURE WORK

The advantage of Smart Greenhouse over conventional farming is that we were able to produce insecticide and pesticide free crops and create a climate for the proper growth of plants and even provides alternative source of income through agriculture, selling tube well water etc. Moreover this system can be installed by any individual in his house (Rooftop greenhouse), who do not have knowledge about farming. Since one can maintain any climatic condition in this type of Greenhouse, it is possible to cultivate any type of crop. Hence, we grow plants like Hibiscus which are imported to India. We can produce 70 percent to 80 percent water requirement. It also increases the yield and rate of growth and produces organic products. Most importantly, we are able to connect farmer directly to consumer using IoT, which can save him from the clutches of middlemen. It reduces effort and time if farmer and makes farming efficient and profitable activity. The smart greenhouse can be further upgraded in many ways and can be used in wide agricultural applications. It can be placed and operated in any of the environmental conditions to grow any kind of vegetation. Non conventional energy sources such as solar panels, wind mills are used to supply power to the automatic greenhouse equipment and Peltier effect for cooling purpose. Soil-less farming can be performed to further improve the nutritional value. Integration of farming with IoT can make it much more efficient and profitable activity. Smart Greenhouse has a bright scope of future in agriculture field and it will create a revolution in the way the agriculture is carried out in India.

REFERENCES

[1] Data Acquisition Of Greenhouse Using Arduino - Journal Of Babylon University/Pure And Applied Sciences/ No.(7)/ Vol.(22): 2014

[2] Greenhouse Automation System Using Psoc 3 - Journal Of Information, Knowledge And Research In Electronics And Communication Engineering

[3] Arduino Based Automatic Plant Watering System - Devika Et Al., International Journal Of Advanced Research In Computer Science And Software Engineering 4(10),October - 2014, Pp. 449-456, Volume 4, Issue 10, October 2014, ISSN: 2277 128X

[4] Remote Sensing In Greenhouse Monitoring System - SSRG International Journal Of Electronics And Communication Engineering (SSRG-IJECE) – EFES April 2015