VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

BECE320E

Embedded C Programming

Digital Assignment

Ajay Maurya (21BEC0775)

Saptarsi Banik (21BEC0828)

Raghavendran Ganesan (21BEC0996)

Title: IoT Based Weather Monitoring System Using ESP8266

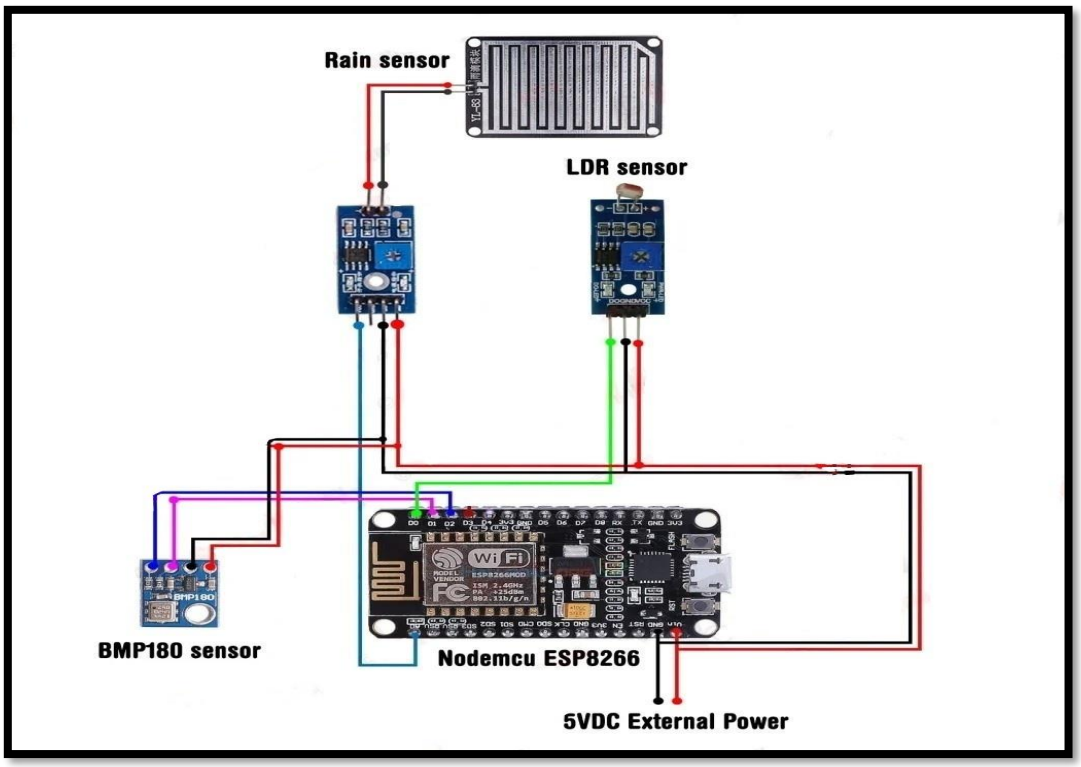
Abstract:

Our project aims to develop an IoT-based weather monitoring system using an ESP8266 module to collect data on various environmental parameters such as atmospheric pressure, temperature, humidity, air quality, and light intensity. The system is designed to provide real-time updates on the weather conditions. The data collected from advanced electronic sensors will be transmitted wirelessly to an IoT cloud service, namely Thingspeak. This cloud platform will serve as a repository for storing and analyzing the gathered data, enabling continuous monitoring and detailed analysis of weather patterns and environmental changes over time. By leveraging the capabilities of the ESP8266 module and integrating it with a range of sensors, our system offers a comprehensive solution for monitoring and reporting weather-related information. The integration with Thingspeak enhances the functionality by providing a user-friendly interface to access and visualize the data in real time. This project contributes to the advancement of IoT applications in weather monitoring, offering valuable insights into local weather conditions and facilitating informed decision-making based on accurate and timely data.

Components Required:

S.No	Component	Quantity
1	Nodemcu ESP8266	1
2	BMP180 sensor	1
3	LDR sensor	1
4	Rain Sensor	1
5	Breadboard	1
6	Jumper Wires	24

Block Diagram:



Discussion:

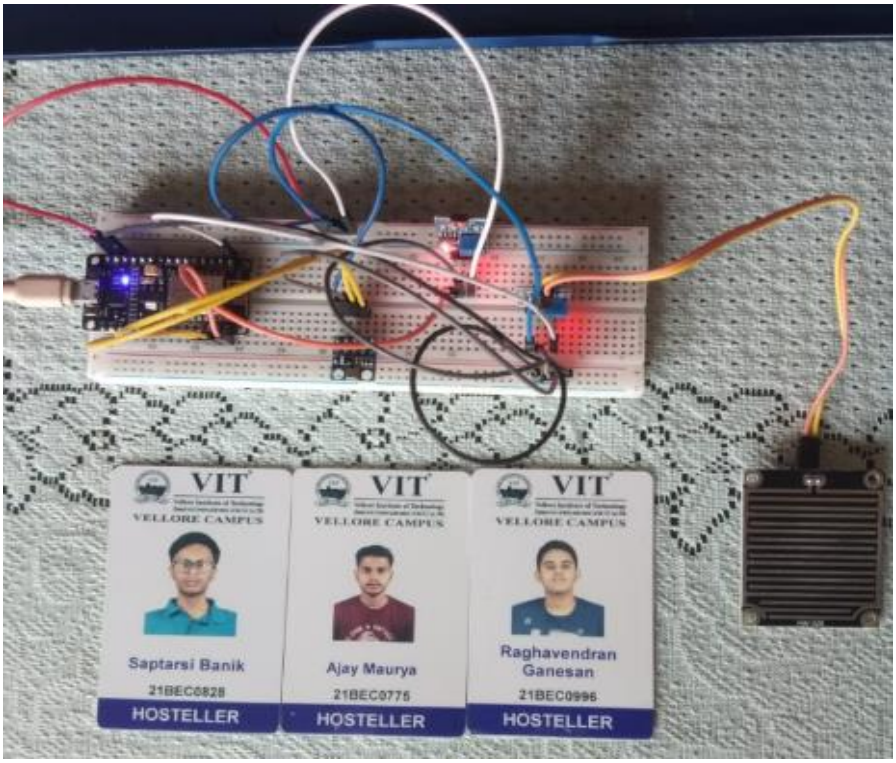
Our IoT-based weather monitoring system using the ESP8266 module presents a robust solution for gathering and analyzing environmental data in real time. The integration of advanced sensors such as the BMP180 sensor for atmospheric pressure, temperature, and altitude measurement, the LDR sensor for light intensity monitoring, and the Rain Sensor for detecting rainfall, enables comprehensive monitoring of key weather parameters. This setup provides a holistic view of weather conditions, which is essential for various applications ranging from agriculture to smart city planning.

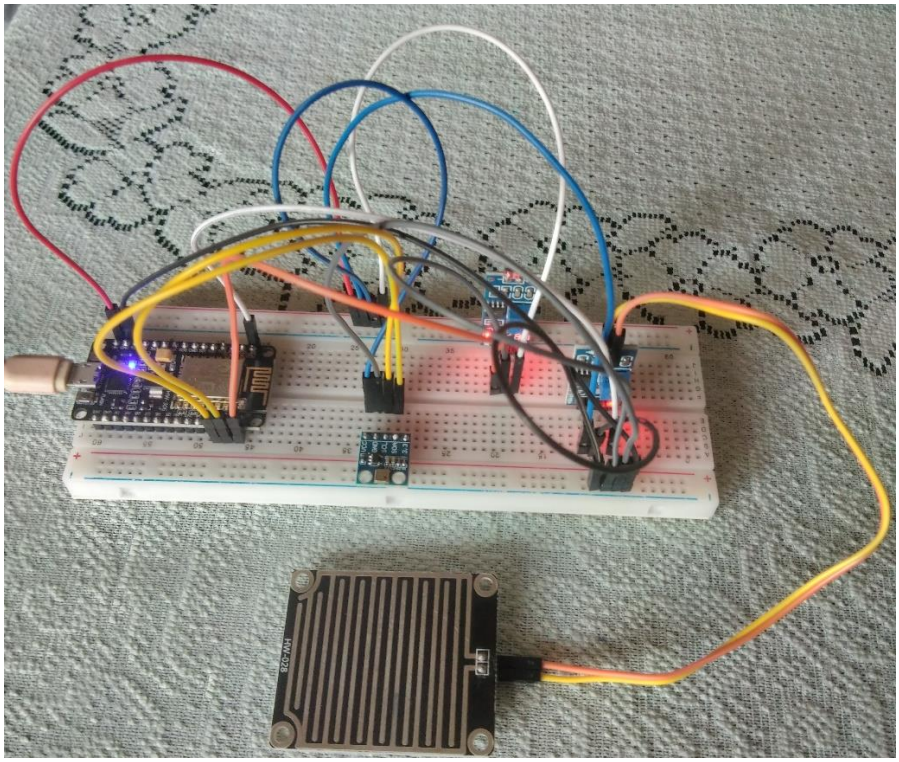
One of the significant advantages of our system is its wireless connectivity through the ESP8266 module, which allows seamless data transmission to the Thingspeak IoT cloud platform. This cloud-based approach facilitates data storage, analysis, and visualization, empowering users to access weather data from anywhere and at any time. The user-friendly interface of Thingspeak enables stakeholders to monitor weather trends, detect anomalies, and make informed decisions based on the analysed data.

Additionally, our system's scalability and modularity are noteworthy aspects. The use of a breadboard and jumper wires allows for easy expansion and

integration of additional sensors or functionalities as needed. This adaptability makes our weather monitoring system suitable for various environments and applications, including home weather stations, agricultural monitoring systems, and industrial weather monitoring setups.

Hardware – Setup Picture:





Arduino IDE Code:

```
//Weather monitoring system
//21BEC0775 21BEC0828 21BEC0996
#include <SFE_BMP180.h>
#include <Wire.h>
#include <ESP8266WiFi.h>
#define DO_PIN D0
SFE_BMP180 bmp;
double T, P;
char status;
WiFiClient client;
String apiKey = "EF5W03FG6MXI7LFG";
const char *ssid =  "vivo 1723";
const char *pass =  "12345678";
const char* server = "api.thingspeak.com";
void setup() {
  pinMode(DO_PIN, INPUT);
  Serial.begin(115200);
  delay(10);
  bmp.begin();
  Wire.begin();
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
}
void loop() {
  //BMP180 sensor
```

```
status =  bmp.startTemperature();
if (status != 0) {
  delay(status);
  status = bmp.getTemperature(T);

  status = bmp.startPressure(3);// 0 to 3
  if (status != 0) {
    delay(status);
    status = bmp.getPressure(P, T);
    if (status != 0) {
    }
  }
}
int r = analogRead(A0);
r = map(r, 0, 1024, 0, 100);
r=(100-r);
  int light_state = digitalRead(DO_PIN);
if (light_state == HIGH)
  Serial.println("The light is not present");
else
  Serial.println("The light is present");

if (client.connect(server, 80)) {
  String postStr = apiKey;
  postStr += "&field1=";
  postStr += String(T,2);
  postStr += "&field2=";
  postStr += String(light_state);
  postStr += "&field3=";
  postStr += String(P, 2);
  postStr += "&field4=";
```

```
   postStr += String(r);
   postStr += "\r\n\r\n\r\n\r\n";
   client.print("POST /update HTTP/1.1\n");
   client.print("Host: api.thingspeak.com\n");
   client.print("Connection: close\n");
   client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
   client.print("Content-Type: application/x-www-form-urlencoded\n");
   client.print("Content-Length: ");
   client.print(postStr.length());
   client.print("\n\n\n\n");
   client.print(postStr);
   Serial.print("Temperature: ");
   Serial.println(T,2);
   Serial.print("absolute pressure: ");
   Serial.print(P, 2);
   Serial.println("mb");
   Serial.print("Rain in percentage:");
   Serial.println(r);
   Serial.print("Light: ");
   Serial.println(light_state);

  }
  client.stop();
  delay(1000);
}
```

Code Explanation:

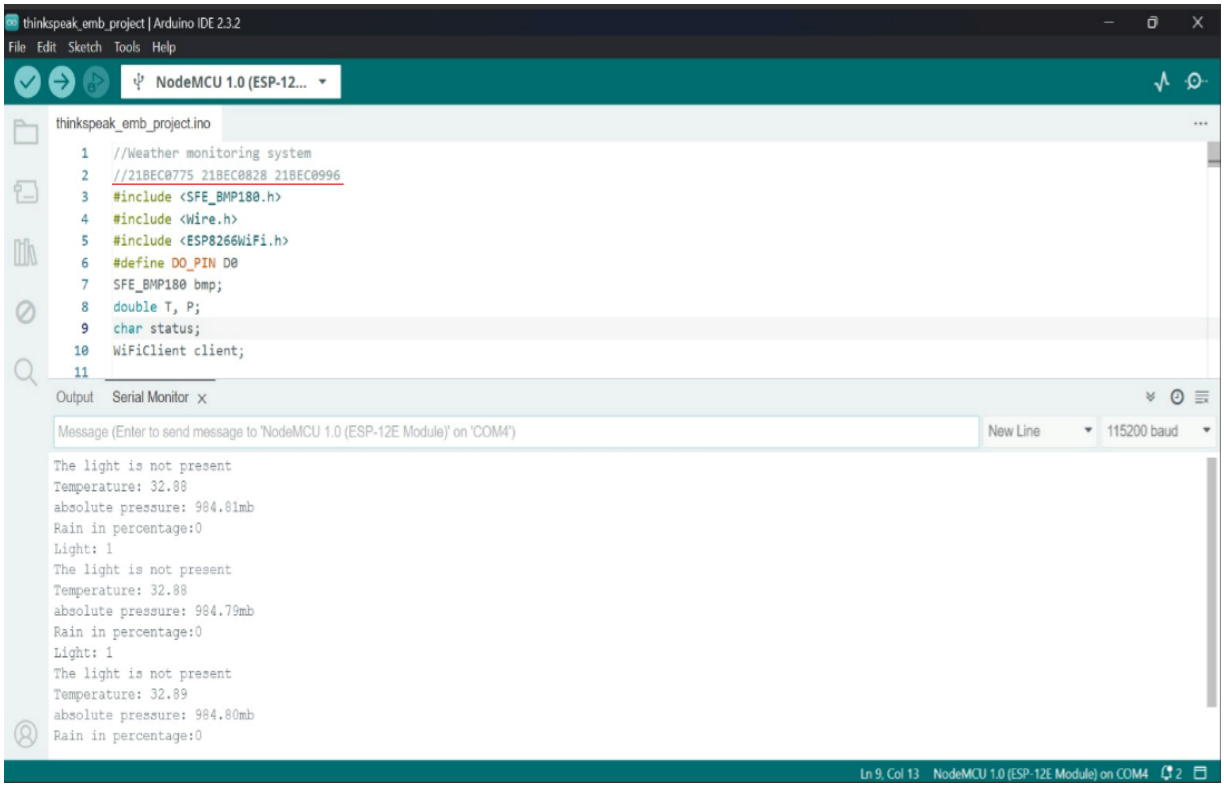
This weather monitoring system code integrates several components to gather and transmit environmental data. It employs an ESP8266 module for connectivity, a BMP180 sensor for temperature and pressure readings, and an LDR sensor to detect light presence. The setup begins by initializing necessary libraries, pins, and establishing a Wi-Fi connection. Upon successful

connection, the main loop starts, where the BMP180 sensor is utilized to collect temperature and pressure data. The LDR sensor's state is also monitored to determine light presence.

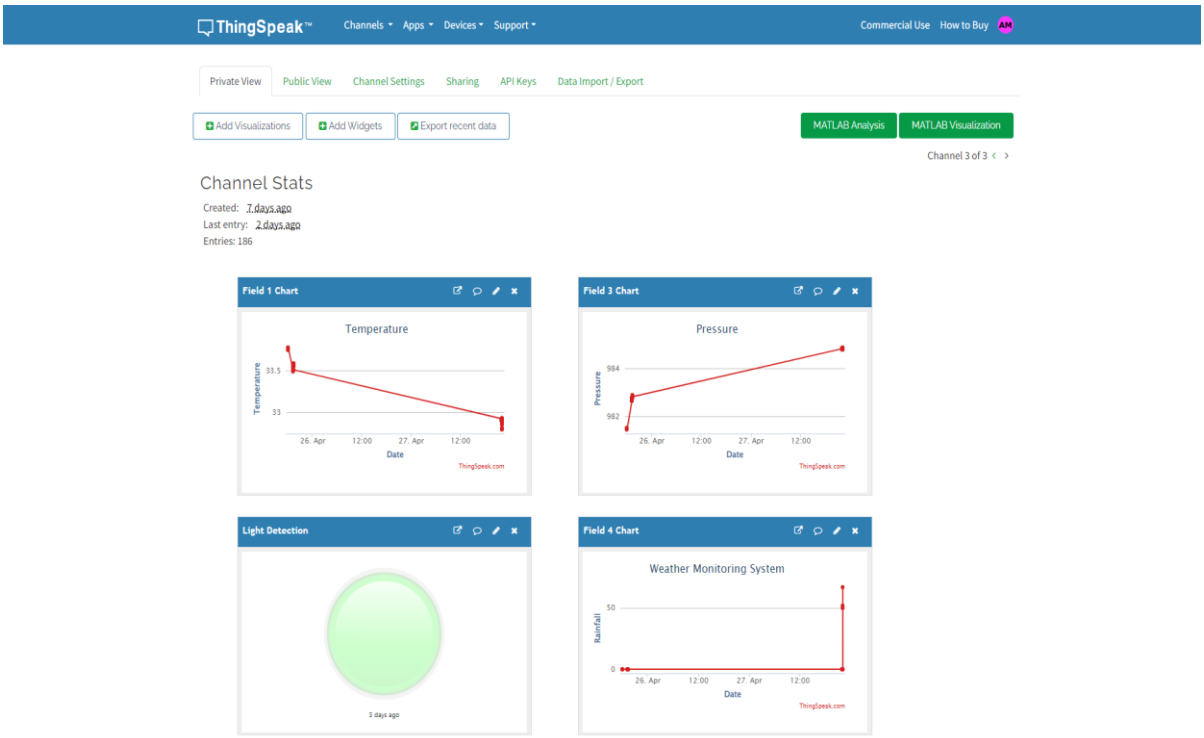
Once data is gathered, the system establishes a connection to the **Thingspeak API** using the Wi-Fi client. It formats the data into a POST request containing temperature, light state, pressure, and rain percentage. This data is then sent to the specified Thingspeak channel for storage and analysis. Additionally, the system provides real-time feedback via the serial monitor, displaying temperature, pressure, rain percentage, and light state readings.

Observationally, this system allows for continuous monitoring of key environmental parameters. The output provides insights such as the absence or presence of light, current temperature, atmospheric pressure, and rain percentage. These metrics are crucial for various applications, including weather forecasting, climate monitoring, and environmental analysis. The integration of IoT technologies like the ESP8266 module and cloud services like Thingspeak enhances data accessibility and facilitates remote monitoring capabilities.

Arduino IDE Output:



ThingSpeak Output Simulation:



Excel file data:

created_at	entry_id	field1	field2	field3	field4
2024-04-25	1	34.22	0	981.38	100
2024-04-25	2	34.26	0	981.4	100
2024-04-25	3	34.5	0	981.31	100
2024-04-25	4	34.63	0	981.4	100
2024-04-25	5	34.69	0	981.37	100
2024-04-25	6	34.68	0	981.39	100
2024-04-25	7	34.63	0	981.39	100
2024-04-25	8	34.66	0	981.4	100
2024-04-25	9	34.62	0	981.39	100
2024-04-25	10	34.58	0	981.33	100
2024-04-25	11	34.53	0	981.33	100
2024-04-25	12	34.59	0	981.41	100
2024-04-25	13	34.56	0	981.34	100
2024-04-25	14	34.51	0	981.33	100
2024-04-25	15	34.47	0	981.34	100
2024-04-25	16	34.44	0	981.33	100
2024-04-25	17	34.41	0	981.3	100

Observation:

The output from the weather monitoring system provides valuable insights into the environmental conditions being monitored. Here are the observations based on the provided output:

1. **Light Presence:**
   - The system reports that the light is not present. This indicates a low light condition or darkness at the location where the monitoring system is deployed.

2. **Temperature:**
   - The temperature reading is 32°C. This temperature value gives an indication of the ambient temperature in the monitored area.

3. **Pressure:**
   - The absolute pressure reading is 994.79 mb (millibars). Pressure measurements are crucial for understanding weather patterns and atmospheric conditions.

4. **Rain Percentage:**
   - The rain percentage is reported as 0%. This suggests that there is no rainfall occurring at the monitored location during the observation period.

For Thingspeak:

Based on the observations,

1. **Temperature Plot**:
   - The temperature plot displays live data from the temperature sensor, showing readings in Fahrenheit.
   - Based on the plot, the temperature is observed to be relatively stable around 33.5 degrees Fahrenheit.

2. **Pressure Plot**:
   - The pressure plot shows real-time data from the pressure sensor, presenting readings in millimetres of mercury (mmHg).
   - Throughout the simulation, the pressure readings remain steady at approximately 944 mmHg.

3. **Light Detection Plot:**
    - The light detection plot represents data from the light sensor, indicating the intensity of light in the environment.
    - . This observation suggests a dark or low-light environment during the simulation.

4. **Rainfall Plot**:
    - The rainfall plot showcases data related to rainfall, providing insights into precipitation levels.

Conclusion:

In summary, our weather monitoring system based on the ESP8266 module successfully collects and transmits real-time data on atmospheric pressure, temperature, light intensity, and rainfall. The integration of advanced sensors like the BMP180, LDR, and Rain Sensor allows for comprehensive monitoring of key weather parameters crucial for various applications.

The ESP8266's wireless connectivity enables seamless data transmission to the Thingspeak IoT cloud platform, facilitating data storage, analysis, and visualization. This system provides a user-friendly interface for accessing weather data, empowering stakeholders with valuable insights into local weather conditions.

Additionally, the system's scalability and modularity make it adaptable to different environments and applications. Its ease of expansion and integration of additional sensors enhance its utility for tasks ranging from home weather monitoring to agricultural and industrial applications. Overall, our project showcases a practical and efficient solution for real-time weather monitoring using IoT technology.