**Project Based Learning Report**

on

**Data Logging Using BMP 180 sensor and Esp8266 on ThingSpeak**

Submitted in the partial fulfilment of the requirements

For the Project based learning in **Industrial IOT & ML**in

Electronics & Communication Engineering

By

**Name of the student PRN**

Ali Shan Ahmad 2114110429

Nur Mohammed Ali 2114110430

MD Gafrujama Ansari 2114110431

Under the guidance of Course In-charge

Dr. Arundhati A.Shinde

# Department of Electronics & Communication

**Engineering**

Bharati Vidyapeeth

(Deemed to be University)

College of Engineering,

Pune – 411043

Year 2023-24

**Bharati Vidyapeeth**

**(Deemed to be University)**

**College of Engineering,**

**Pune – 411043**

**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

# CERTIFICATE

Certified that the Project Based Learning report entitled on –

**Data Logging Using BMP 180 sensor and Esp8266 on ThingSpeak** is a bonafied work done by

**Name of the student PRN**

Ali Shan Ahmad 2114110429

Nur Mohammed Ali 2114110430

MD Gafrujama Ansari 2114110431

in partial fulfilment of the requirements for the award of credits for Project Based Learning (PBL) in Industrial IOT & ML Bachelor of Technology Semester VI, in Electronics and Communication.

**Date: 11/04/24**

**Dr. Arundhati A.Shinde Dr. Arundhati A.Shinde**

**Course In-charge Professor & Head**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Content** | **Page No.** |
| **1.** | Problem statement | **1** |
| **2.** | Project Description | **1** |
| **3.** | Software Used | **2** |
| **4** | Components Used | **3** |
| **5.** | Algorithm | **5** |
| **6.** | Program | **6** |
| **7.** | Circuit Diagram | **8** |
| **8.** | Output | **9** |
| **9.** | Applications | **10** |
| **10.** | Conclusion & Course Outcome | **12** |
| **11.** | References | **13** |

**Problem Statement**

Data Logging Using BMP 180 sensor and Esp8266 on ThingSpeak

**Project Description**

In this project, we aim to implement a system for data logging using the BMP180 sensor and ESP8266 microcontroller, with data visualization on ThingSpeak, an Internet of Things (IoT) platform. The BMP180 sensor is utilized to measure environmental parameters such as temperature, pressure, and altitude, while the ESP8266 serves as the microcontroller for data acquisition and transmission.

**Project Objectives –**

1. **Sensor Integration:** Interface the BMP180 sensor with the ESP8266 microcontroller to acquire accurate and reliable temperature, pressure, and altitude data.
2. **Data Transmission:** Establish a stable Wi-Fi connection using the ESP8266 to transmit sensor data securely and efficiently to the ThingSpeak platform.
3. **Data Logging:** Implement robust data logging mechanisms on ThingSpeak to store incoming sensor data persistently for future analysis and reference.
4. **Real-time Visualization:** Configure ThingSpeak channels with appropriate field mappings to enable real-time visualization of sensor data using graphs, charts, and gauges on the ThingSpeak dashboard.
5. **Remote Monitoring:** Develop monitoring capabilities to allow users to remotely access and monitor environmental parameters in real-time from any internet-connected device.
6. **Alerting System:** Implement an alerting system on ThingSpeak to notify users of abnormal sensor readings or predefined thresholds through email or SMS alerts.
7. **Data Analysis:** Enable data analysis functionalities on ThingSpeak to perform statistical analysis, trend detection, and anomaly detection on collected sensor data.
8. **User Interface Enhancement:** Enhance the user interface of the ThingSpeak dashboard to provide an intuitive and user-friendly experience for interacting with sensor data.
9. **Scalability and Flexibility:** Design the system architecture to be scalable and flexible, allowing for easy integration of additional sensors, expansion of monitoring capabilities, and adaptation to varying environmental conditions.

**Software Used**

**Arduino IDE**

The Arduino Integrated Development Environment, often referred to as the Arduino IDE, is a user-friendly and open-source software platform specifically designed for programming and developing applications for Arduino boards. This IDE provides a rich set of tools and features to simplify the process of writing, compiling, and uploading code to microcontroller-based projects. The Arduino IDE is highly favoured by both hobbyists and professionals in the field of embedded systems and electronics.



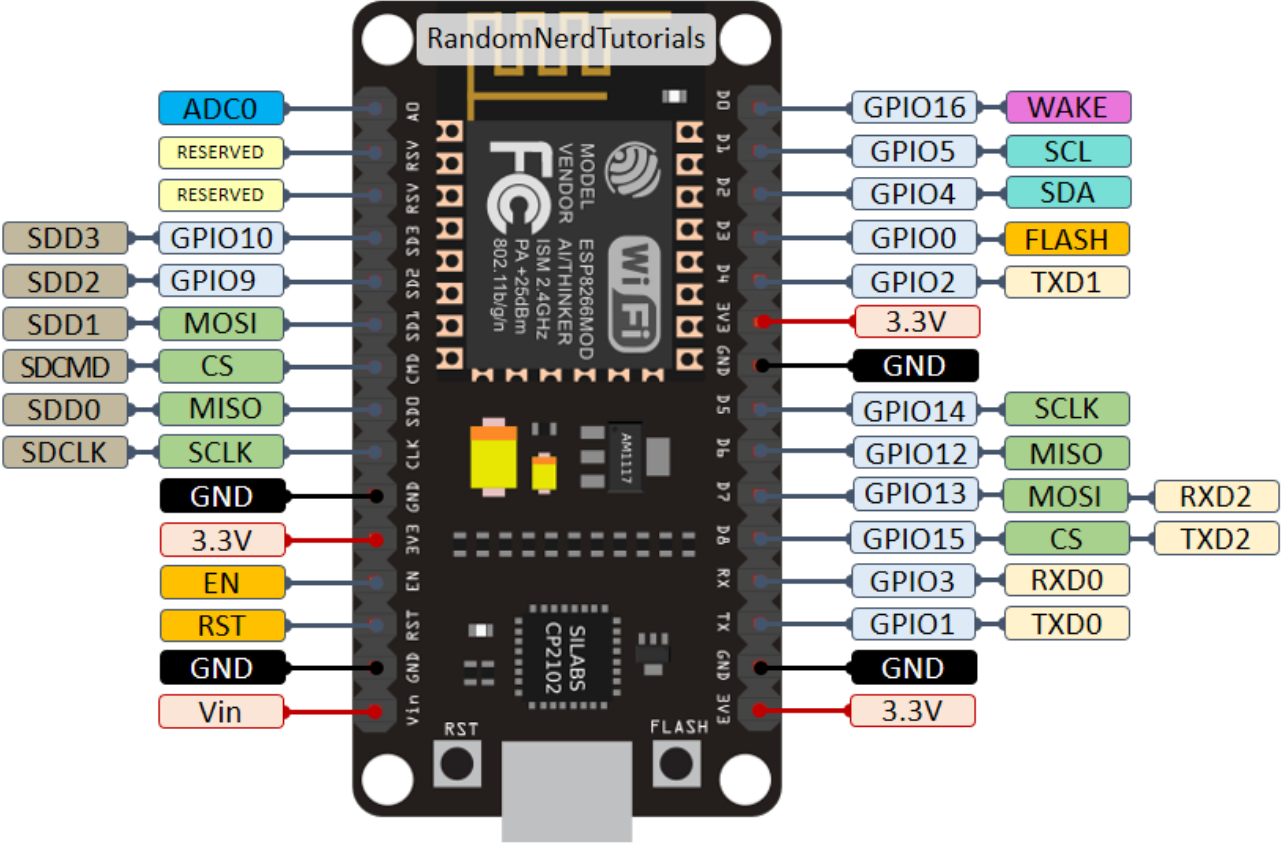
**ThingSpeak –**

"ThingSpeak, a prominent IoT platform, was utilized for data visualization and management in this project. It provides a user-friendly interface for real-time data logging, visualization, and analysis. With its powerful visualization tools and customizable dashboards, ThingSpeak enabled us to effectively monitor and analyze the sensor data collected by the BMP180 sensor and ESP8266 microcontroller. Additionally, ThingSpeak's flexibility and scalability made it suitable for accommodating future expansions and enhancements to our monitoring system."

**Components Used**

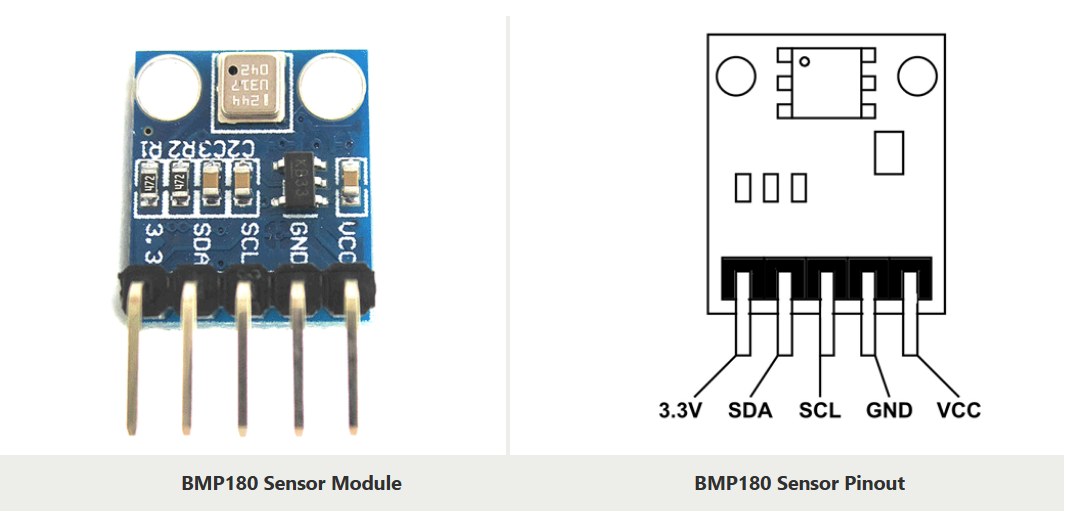
**Node MCU ESP8266 –**

* The NodeMCU, based on the ESP8266 Wi-Fi module, served as the master microcontroller platform, enabling Wi-Fi connectivity and efficient data transfer between the Arduino Uno and other devices.

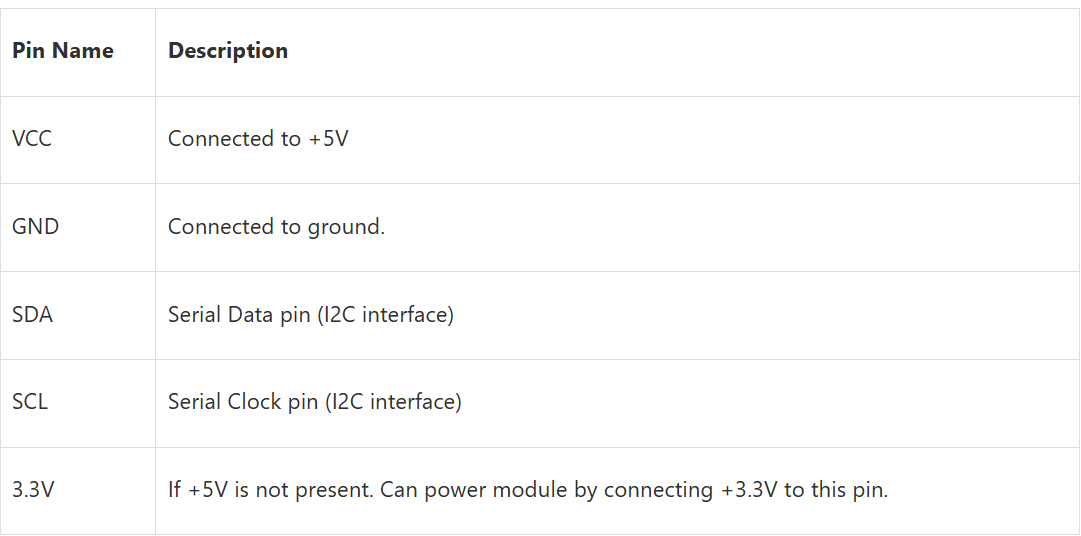
****

**BMP 180 Sensor –**

**BMP180** is one of sensor of BMP XXX series. They are all designed to **measure Barometric Pressure** or **Atmospheric pressure**. BMP180 is a high precision sensor designed for consumer applications. Barometric Pressure is nothing but weight of air applied on everything. The air has weight and wherever there is air its pressure is felt. **BMP180 sensor** senses that pressure and provides that information in digital output. Also the temperature affects the pressure and so we need temperature compensated pressure reading. To compensate, the BM180 also has good temperature sensor.



**BMP 180 Pinout –**

****

**BMP180 MODULE Features**

* Can measure temperature and altitude.
* Pressure range: 300 to 1100hPa
* High relative accuracy of ±0.12hPa
* Can work on low voltages
* 3.4Mhz I2C interface
* Low power consumption (3uA)
* Pressure conversion time: 5msec
* Potable size

**BMP180 MODULE Specifications**

* Operating  voltage of  BMP180: 1.3V – 3.6V
* Input voltage of  BMP180MODULE: 3.3V to 5.5V
* Peak current : 1000uA
* Consumes 0.1uA standby
* Maximum voltage at SDA , SCL : VCC + 0.3V
* Operating temperature: -40ºC to +80ºC

**Algorithm**

1. **Include Libraries:**
   * Include the necessary libraries for Wire communication, BMP085 sensor, ESP8266 WiFi, and ThingSpeak.
2. **Define Constants:**
   * Define the sea level pressure constant *seaLevelPressure*\_*hPa*).
   * Set up WiFi credentials, including SSID and password.
   * Specify ThingSpeak channel settings such as Channel ID and Write API Key.
3. **Initialize Setup Function:**
   * Initialize serial communication for debugging purposes.
   * Connect to the WiFi network using provided credentials.
   * Initialize the BMP180 sensor and check for its availability.
   * Enter an infinite loop for continuous data logging.
4. **Continuous Loop (Loop Function):**
   * Check the WiFi connection status. If not connected, attempt to reconnect and delay for a short duration.
   * Read sensor data including temperature, pressure, and altitude from the BMP180 sensor.
   * Convert pressure from Pascal to kg/cm².
   * Print the sensor data (temperature, pressure in kg/cm², and altitude) to the serial monitor.
   * Begin communication with ThingSpeak using the WiFi client.
   * Set the field values for temperature, pressure, and altitude.
   * Write the data to the ThingSpeak channel.
   * Check the response from ThingSpeak. If successful (HTTP response code 200), print a success message; otherwise, print an error message.
   * Delay for a specified interval before logging the next set of data.

**Program –**

#include <Wire.h>

#include <Adafruit\_BMP085.h>

#include <ESP8266WiFi.h>

#include <ThingSpeak.h>  **// ThingSpeak library**

#define seaLevelPressure\_hPa 1013.25

Adafruit\_BMP085 bmp;

**// WiFi credentials**

const char\* ssid = "Galaxy S10";

const char\* password = "aisha1234";

**// ThingSpeak channel settings**

unsigned long channelID = 2503051; **// Your ThingSpeak Channel ID**

const char\* writeAPIKey = "RNBWWVOJLNO3XFTR"; **// Your ThingSpeak Write API Key**

WiFiClient client;

void setup() {

  Serial.begin(115200);

  WiFi.begin(ssid, password);

**// Initialize the BMP180 sensor**

  if (!bmp.begin()) {

    Serial.println("BMP180 Not Found. CHECK CIRCUIT!");

    while (1) {}

  }

}

void loop() {

**// Connect to WiFi**

  if (WiFi.status() != WL\_CONNECTED) {

    Serial.println("Connecting to WiFi...");

    delay(1000);

    return;

  }

**// Read sensor data**

  float temperature = bmp.readTemperature();

  float pressure = bmp.readPressure();

  float altitude = bmp.readAltitude();

  pressure = pressure/ 98066.5;

  Serial.print("Temperature = ");

  Serial.print(temperature);

  Serial.println(" \*C");

  Serial.print("Pressure = ");

  Serial.print(pressure);

  Serial.println(" kg/m2");

  Serial.print("Altitude = ");

  Serial.print(altitude);

  Serial.println(" m");

**// Publish data to ThingSpeak**

  ThingSpeak.begin(client);

  ThingSpeak.setField(1, temperature);

  ThingSpeak.setField(2, pressure);

  ThingSpeak.setField(3, altitude);

  int response = ThingSpeak.writeFields(channelID, writeAPIKey);

  if (response == 200) {

    Serial.println("Data sent to ThingSpeak successfully.\n");

  } else {

    Serial.print("Error sending data to ThingSpeak. HTTP error code ");

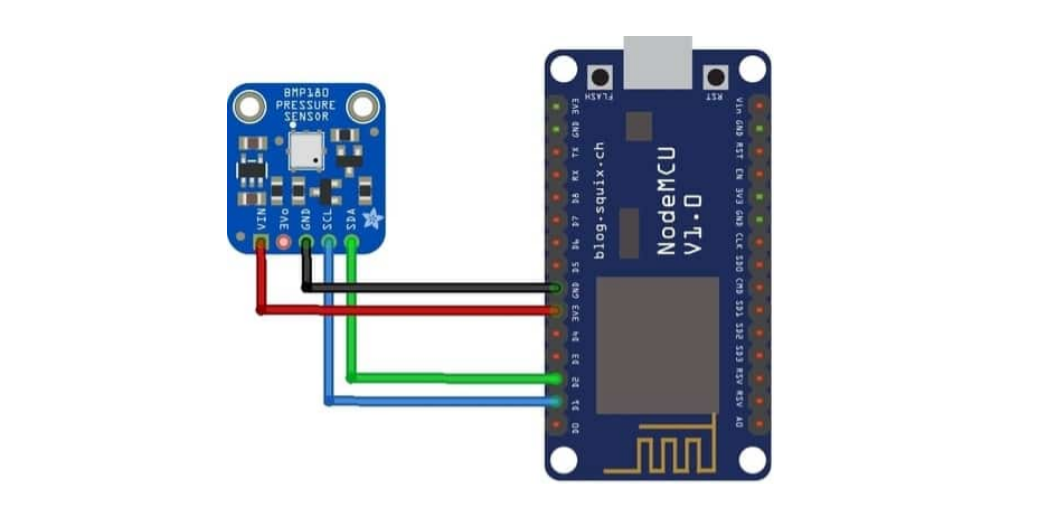
    Serial.println(response);

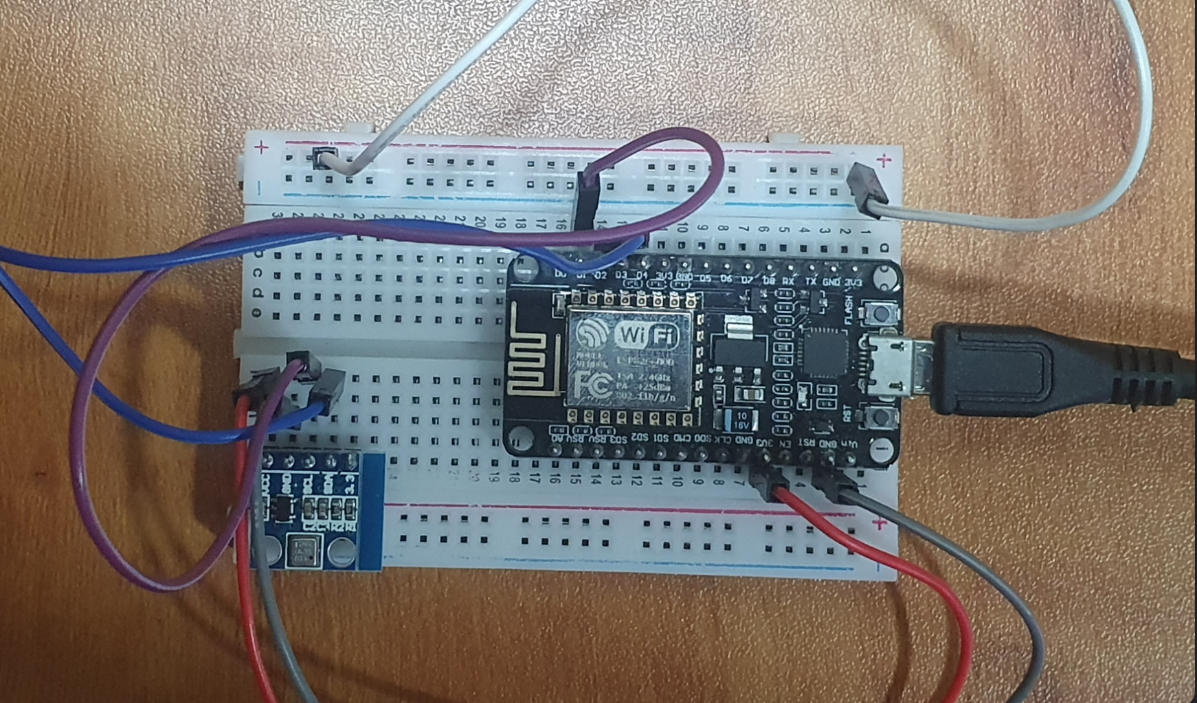
  }

  delay(2000); **// Delay before sending next data (in milliseconds)**

}

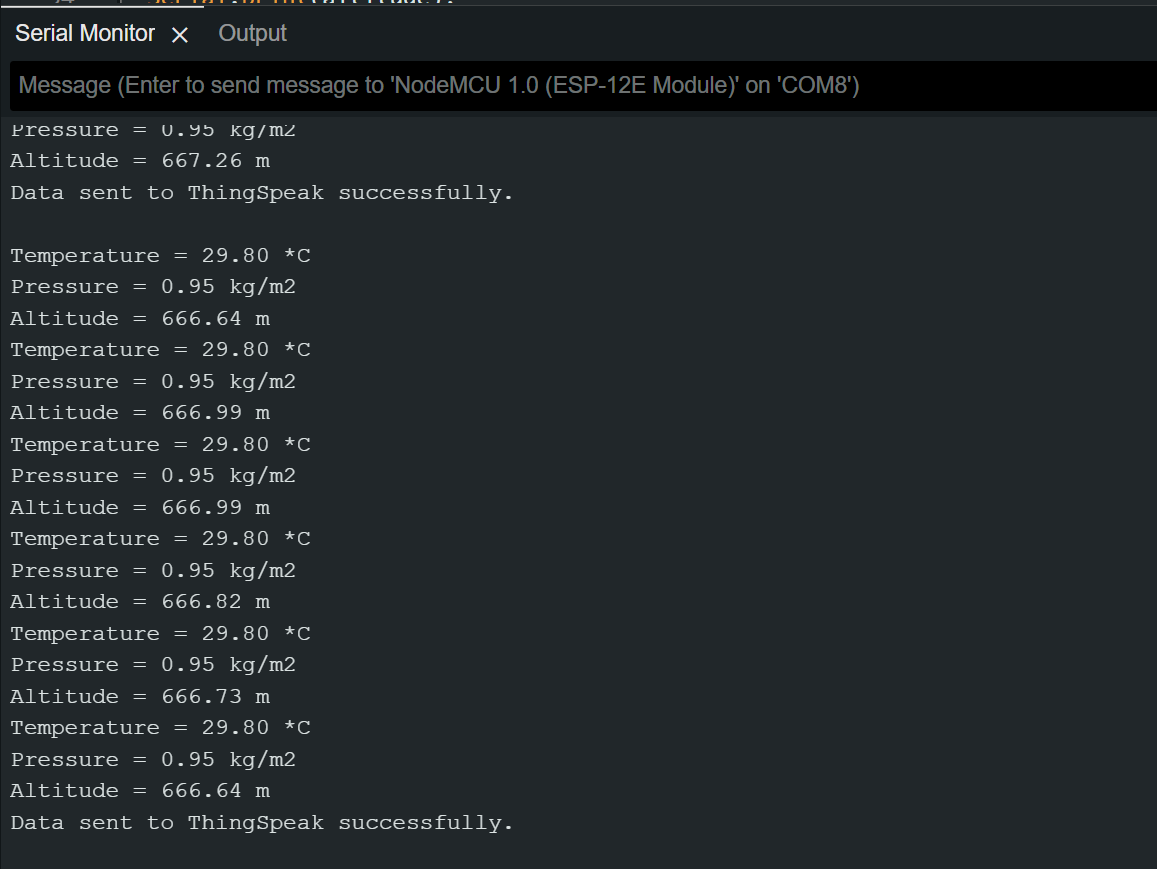
**Circuit Diagram –**

****

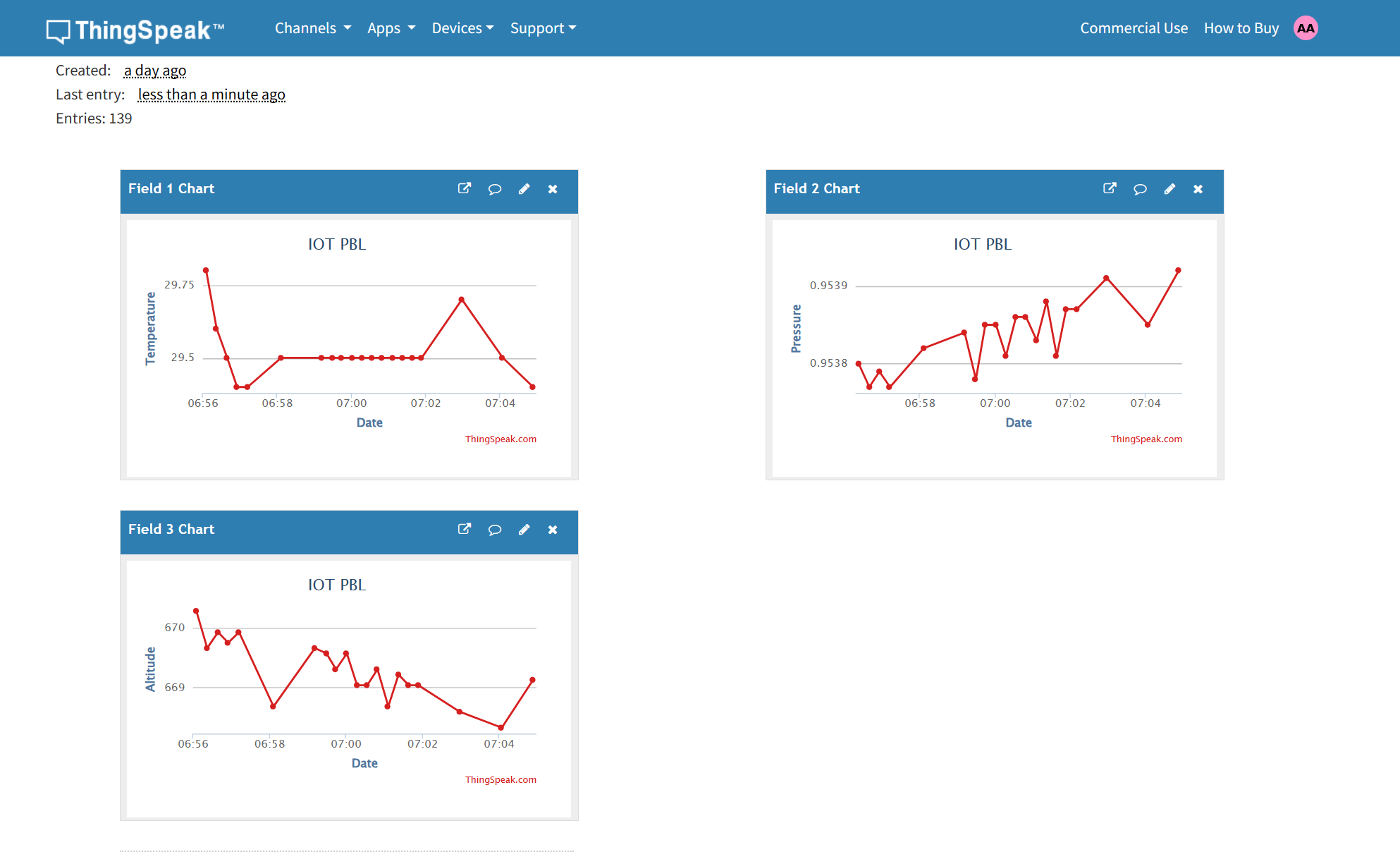
****

**Output –**

**Data displayed on Srieal Monitor –**

****

**Data Logged on ThingSpeak –**

****

**Applications –**

The BMP180 sensor, a barometric pressure and temperature sensor, finds numerous applications across various industries and domains due to its ability to accurately measure atmospheric pressure and temperature. Here's a detailed explanation of a real-life application of the BMP180 sensor:

**1. Weather Monitoring and Forecasting:**

**2. Altitude Measurement in Aviation:**

**3. Indoor and Outdoor Navigation:**

**4. Environmental Monitoring and Agriculture:**

**5. Industrial Applications:**

**BMP Sensor in Airplanes:**

In airplanes, the BMP180 sensor serves several essential functions:

* **Altitude Measurement:** The BMP180 sensor accurately measures barometric pressure, which is used to determine the altitude of the aircraft above sea level. This information is crucial for navigation, flight planning, and maintaining safe flight operations.
* **Airspeed Calculation:** By comparing the pressure readings from the BMP180 sensor with the static pressure outside the aircraft, the airspeed of the aircraft can be calculated. This is particularly useful for determining the aircraft's speed relative to the surrounding air during flight.
* **Flight Control Systems:** BMP180 sensor data is often integrated into flight control systems to assist in altitude control, autopilot functions, and instrument landing systems. It helps the aircraft maintain a stable flight path and altitude, especially during automated flight modes.
* **Emergency Situations:** In the event of a loss of cabin pressure or depressurization, the BMP180 sensor can detect changes in atmospheric pressure inside the aircraft. This information triggers warning systems and emergency protocols to ensure passenger safety and initiate emergency descent procedures.

**BMP Sensor in Underwater Diving:**

Although the BMP180 sensor is primarily designed for atmospheric pressure measurement, it can also be adapted for use in underwater diving applications:

* **Depth Measurement:** Similar to altitude measurement in aviation, the BMP180 sensor can be used to measure the depth of water during underwater diving activities. By calibrating the sensor to account for the density of water and hydrostatic pressure, it can provide accurate depth readings.
* **Dive Computers:** Some dive computers and underwater monitoring devices utilize BMP180 sensors to measure depth, temperature, and dive profiles. This information is crucial for divers to monitor their ascent and descent rates, track bottom time, and ensure safe decompression during scuba diving activities.
* **Safety Features:** BMP180 sensor data is integrated into dive computers to provide real-time depth information and trigger safety alarms for ascent rate violations, decompression sickness risks, and low-pressure warnings. This helps divers avoid hazardous situations and adhere to safe diving practices.
* **Research and Exploration:** Scientists and researchers may deploy underwater monitoring systems equipped with BMP180 sensors to study underwater ecosystems, monitor environmental conditions, and conduct underwater surveys. The sensor data contributes to research efforts aimed at understanding marine life, ecosystems, and ocean dynamics.

.

**Conclusion –**

The BMP180 sensor is regarded as a versatile and indispensable tool for a wide range of applications across various industries and domains. Through its precise measurement of atmospheric pressure and temperature, critical functions such as weather monitoring, altitude measurement, navigation, environmental monitoring, industrial automation, and more are facilitated.

Throughout this report, the diverse real-life applications of the BMP180 sensor have been explored, ranging from weather forecasting and aviation to underwater diving and industrial automation. In aviation, altitude determination, airspeed calculation, and flight control are aided by the BMP180 sensor, contributing to safe and efficient air travel. Similarly, in underwater diving, accurate depth measurement, safety monitoring, and exploration of marine environments are enabled.

Moreover, applications in weather stations, GPS navigation systems, agricultural monitoring, industrial HVAC systems, and DIY projects highlight the versatility and adaptability of the BMP180 sensor to various scenarios. By providing accurate and reliable sensor data, informed decisions, enhanced safety, optimized processes, and advances in scientific knowledge are empowered.

As technology continues to evolve, the BMP180 sensor remains a vital component in the ever-expanding landscape of sensor technology and IoT (Internet of Things). Its compact size, low power consumption, and ease of integration make it a preferred choice for developers, engineers, and enthusiasts seeking to implement sensor-based solutions for diverse applications.

In conclusion, the BMP180 sensor is exemplified as the convergence of innovation, utility, and reliability, positioning it as an indispensable asset in modern-day technology ecosystems.

**Course Outcome –**

The successful completion of this project has significantly contributed to the fulfillment of the following course outcomes.

**CO-I :** To understand the basic concept and the industrial IoT Paradigm.

**CO-IV :** To design basic IIoT Applications

**References –**

* J Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st Edition, Academic Press, 2014.
* Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st Edition, Academic Press, 2014.
* Sai Yamanoor,SrihariYamanoor ,Python programming with Raspberry Pi , Packt
* Publishing,2017
* A. Gupta, “Microcontroller and Embedded Systems”, S.K. Kataria & Sons (India), 2019.
* <https://how2electronics.com/bmp180-pressure-temperature-thingspeak-esp8266/>
* <https://thingspeak.com/channels/2503051>
* <https://randomnerdtutorials.com/esp8266-nodemcu-bmp388-arduino/>
* <https://diyi0t.com/spi-tutorial-for-arduino-and-esp8266/>
* <https://www.instructables.com/Adding-the-BMP180-to-the-ESP8266/>