Industrial Solar Monitoring System

Y.Mallikarjuna Rao M.Tech(Ph.D) (Assistant professor), P.Akash ,N.Pranadeep,V.Narendra,Sd.Mohammed Taheer
Department of Electrical and Electronics Engineering
RVR&JC College of Engineering Chowdavaram,Guntur,Andhra Pradesh, India

Abstract:

This project presents an industrial-grade solar panel monitoring system built around a single ESP32 microcontroller. It is designed to monitor a 2S2P solar array, consisting of four panels connected as two series strings in parallel, to measure both voltage and current output. In addition to electrical parameters, the system also tracks environmental conditions using multiple sensors, including a DHT11 sensor for temperature and humidity, an MQ2 sensor for gas and smoke detection, an MQ135 sensor for air quality monitoring, and a BMP180 sensor for atmospheric pressure measurement. All collected data is displayed in real time on a 16x2 I2C LCD for local monitoring and is also transmitted to the Blynk IoT cloud platform for remote access via a mobile application. The system includes a safety feature for gas detection, providing clear status messages such as “Gas Detected” or “Safe” on both the display and the app, along with automatic alert notifications to ensure timely user awareness.

Keywords:

1. Microcontroller
2. Solar Monitoring
3. Environmental Sensors
4. Display & Communication

1 Introduction

Industrial solar panel systems require continuous monitoring of electrical output and environmental conditions for efficient operation. Existing monitoring methods rely heavily on manual inspection, which is time-consuming and prone to human error.

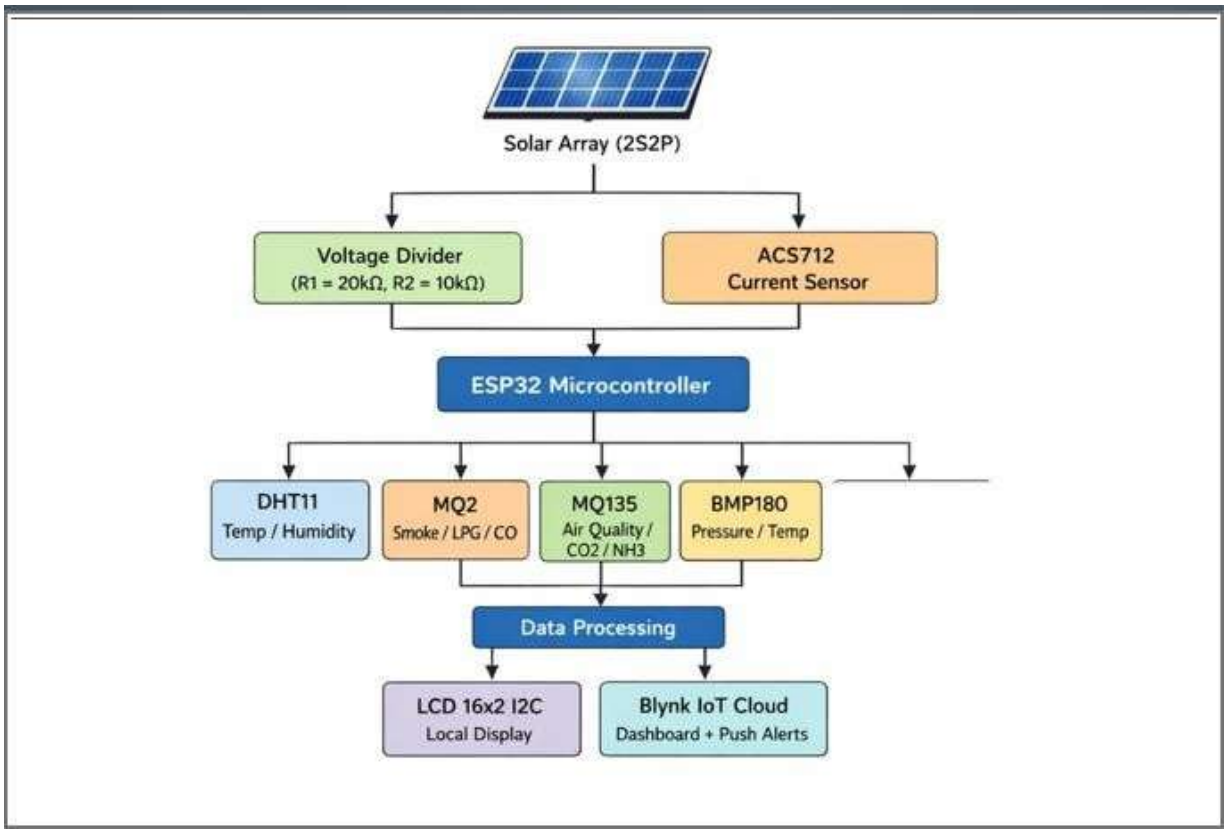
There is no reliable way to obtain real-time data and instant alerts using traditional approaches. Hazardous gas leaks in industrial environments pose serious safety risks, with limited or no automated detection systems in place.Lack of an integrated system results in separate monitoring of performance and safety parameters, reducing overall effectiveness.

There is a need for a low-cost, automated, and remotely accessible solution to monitor both electrical and environmental conditions in a single system

Objective:

The proposed IoT-Based Battery Monitoring System for Electric Vehicles enhances conventional methods by integrating embedded control with IoT connectivity for real-time monitoring and intelligent protection. The system utilizes an Arduino to continuously measure voltage, current, and temperature parameters, displaying them locally on an LCD while transmitting live data to the ThingSpeak cloud platform through NodeMCU. Automated protection mechanisms are implemented using relays to control charging and load conditions when abnormal parameters are detected. Visual (LED) and audible (buzzer) alerts are activated during faults, and instant notifications are sent to users via a GSM module. The system also simulates EV load using a DC motor and manages auxiliary charging operations. By combining real-time data acquisition, remote monitoring, automated control, and alert systems, the proposed model provides a smart, reliable, and scalable battery management solution for electric vehicles.

2. Block diagram:



Components:

ESP32

ACS712

DHT11 Sensor

MQ2 Sensor

MQ135 Sensor

BMP180 Sensor

16x2 LCD Display

Blynk IoT Cloud

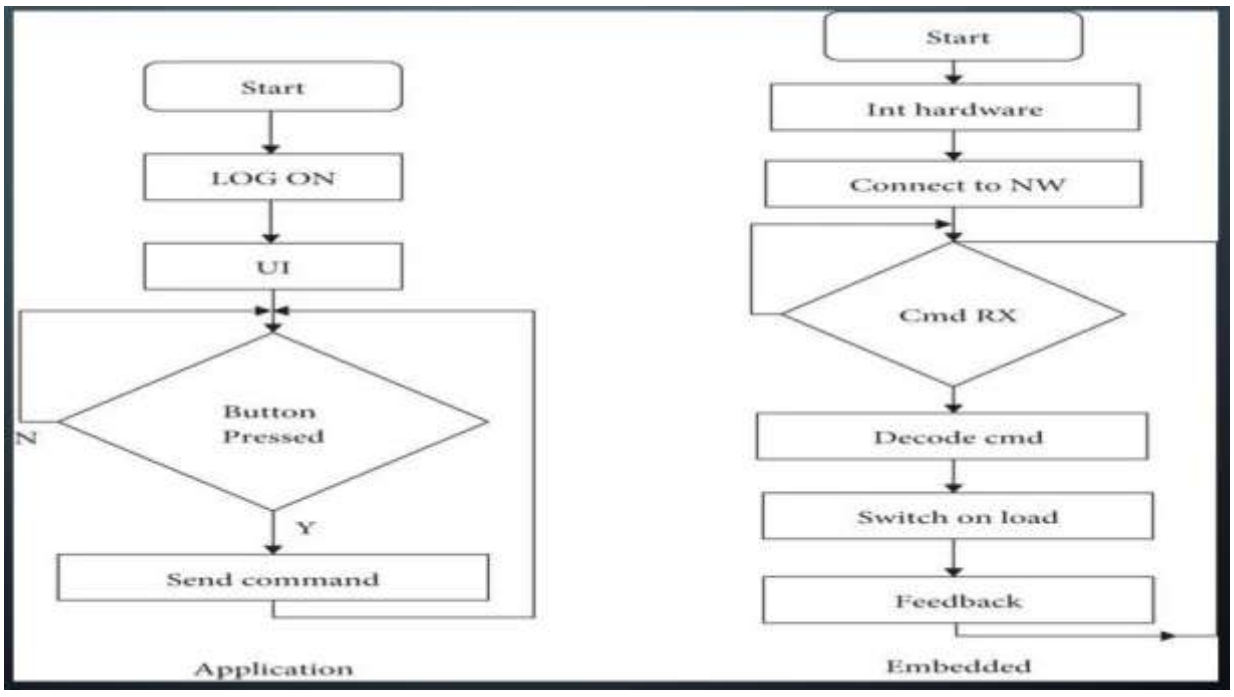
Solar Panels

Resistors

Jumper wires

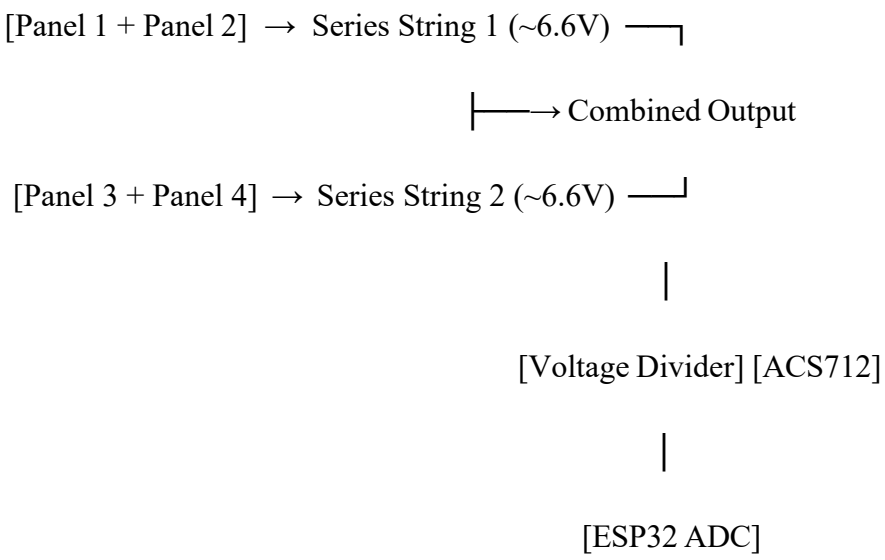
BreadBoard

3. System Flowchart:



4. Solar Panel Configuration — 2S2P

- **Configuration :**



- **Series Connection Benefits:**

  - Voltage adds up: 3.3V + 3.3V = ~6.6V per string

  - Current remains the same as a single panel

- **Parallel Connection Benefits:**

  - Voltage stays the same (~6.6V)

  - Current doubles (both strings combined)

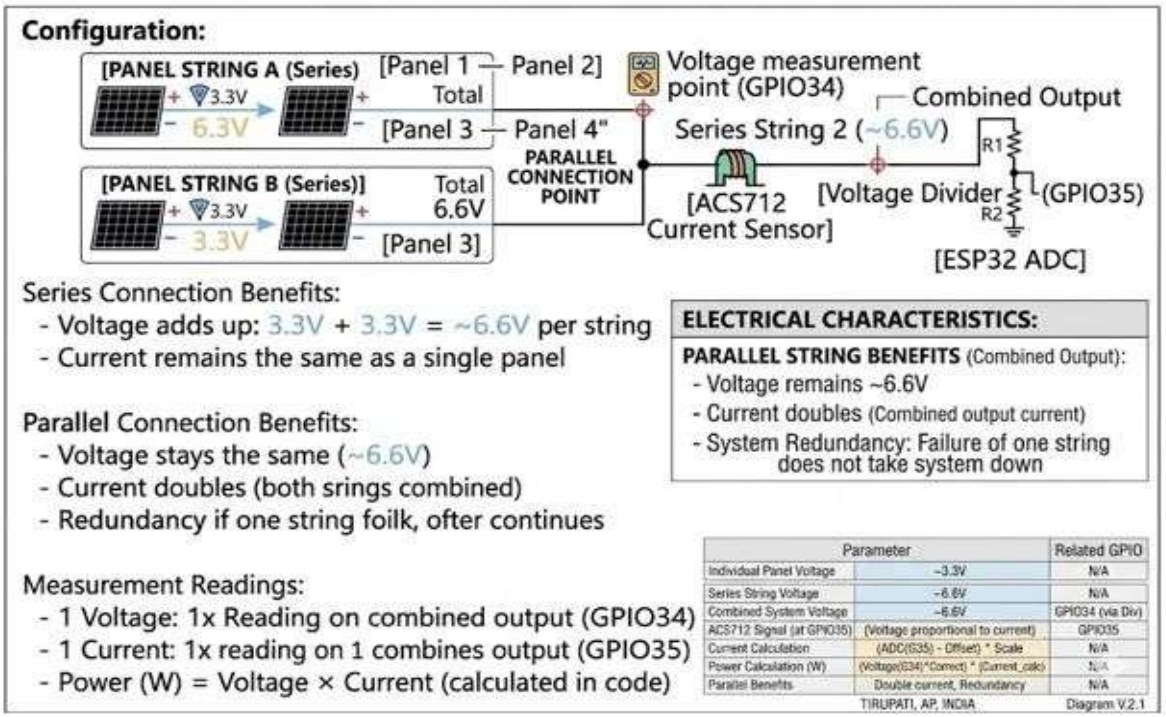
  - Redundancy: if one string fails, other continues

- **Measurement Point:**

  - 1 Voltage reading on final combined output (GPIO34)

  - 1 Current reading on final combined output (GPIO35)

  - Power (W) = Voltage × Current (calculated in code)



5. ACS712 5A Current Sensor:

Working Principle:

Based on Hall Effect magnetic field from current flow

-induces a proportional voltage at the output pin.

-At 0A: dutput VCC/2~2.5V (midpoint)

-Sensitivity: 185 mV per Ampere (5A module)

Formula Used in Code:

Current (A) = (V_out-V_midpoint) /Sensitivity

Where:

V-midpoint (2048/4095) x 3.3V × 1000 mV

Key Parameters:

- Module version: ACS712-05B (5A)

- Sensitivity: 185 mV/A

- Supply voltage: 5V (module), output read by ESP32

- ESP32 ADC pin: GPIO35

- Averaging: 20 samples for noise reduction

-Noise floor: Readings < 0.05A treated as OA

Wiring:

IP+/IP: In-series with solar output line

-VIOUT: GPIO35 (ESP32 ADC)

- VCC: 5V

-GND: GND

6. Voltage Measurement — Voltage Divider

Why needed?

- ESP32 ADC accepts maximum 3.3V
- Solar array output (~6.6V) exceeds this direct connection will damage ESP32

Voltage Divider Formula:

$V_{out} = V_{in} \times R2 / (R1 + R2)$

$V_{actual} = V_{adc} \times (R1 + R2) / R2$

Resistor Values:

- R1 = 20,000 Ω (20kΩ)
- R2 = 10,000 Ω (10kΩ)
- Multiplier = (20000 + 10000) / 10000 = 3.0
- Max safe input = 3.3V × 3.0 = 9.9V ✓ (covers 6.6V range)

Circuit:

Solar(+)
──[R1=20kΩ]──┬──[R2=10kΩ]── GND
│
GPIO34 (ESP32 ADC)

Key Details:

ADC Resolution : 12-bit (0 to 4095)
ADC Attenuation : 11dB (full 0–3.3V range)
Averaging : 20 samples per reading
Noise floor : Readings < 0.05V treated as 0V

7. Temperature & Humidity Sensor

Overview:

- Measures ambient temperature and relative humidity
- Single-wire digital communication protocol
- Connected to GPIO4 on ESP32

Specifications:

- Temperature range : 0°C to 50°C (±2°C accuracy)
- Humidity range : 20% to 90% RH (±5% accuracy)
- Sampling rate : 1 reading per second (max)
- Supply voltage : 3.3V – 5V

Importance in Solar Monitoring:

- High temperature reduces solar panel efficiency
- Humidity affects insulation and wiring integrity
- BMP180 provides a secon temperature reference

LCD Display: Page 3 → "Temp : 34.5°C" / "Hum : 65.0%"
Blynk: V3 → Temperature (°C) V4 → Humidity (%)

8. MQ2 — Smoke, LPG & CO Gas Sensor

Overview:
Electrochemical sensor sensitive to combustible gases
Detects: LPG, Propane, Methane,
Smoke, CO, Hydrogen
Analog output proportional to gas concentration
Connected to GPIO32 on ESP32
Working Principle:
Tin Dioxide (SnO2) sensing element
Resistance decreases as gas concentration increases
ADC reads analog voltage → converted to raw value (0–4095)
Threshold in Code:
MQ2_THRESHOLD = 600 (adjustable after warm-up calibration)
Above threshold → mq2Alert = true → "Gas Detected“
Below threshold → "Safe"
Warm-Up Note:
Sensor requires 24–48 hours initial burn-in for stable readings
Allow 2–3 minutes power-on warm-up before each use
LCD Display:
Page 4 → "MQ2 : Gas Detected" or "MQ2 : Safe“
Blynk:
V5 → "Gas Detected" or "Safe" (Label widget)

9. MQ135 — Air Quality Sensor

Overview:

- Detects harmful gases in air: CO2, NH3, Benzene, Alcohol, Smoke
- Used to monitor overall air quality in industrial environments
- Analog output connected to GPIO33 on ESP32

Detectable Gases:

- Ammonia (NH3)
- Nitrogen Oxides (NOx)
- Carbon Dioxide (CO2)
- Benzene / Acetone vapors

- Smoke particles

Threshold in Code:

- MQ135_THRESHOLD = 700 (adjustable)
- Above threshold → mq135Alert = true→ "Pollution Detected"
- Below threshold → "Safe"

Combined Alert Logic:

gasAlert = mq2Alert OR mq135Alert

If BOTH triggered → LCD shows "MQ2 + MQ135 !!"

LCD Display:

Page 4 → "MQ135: Pollution!" or "MQ135: Safe"

Blynk:

V6 → "Pollution Detected" or "Safe" (Label widget)

V9 → "ALERT" or "All Clear" (overall status)

10. BMP180 — Atmospheric Pressure & Temperature Sensor

Overview:

- Digital barometric pressure sensor with I2C interface
- Shares I2C bus with LCD (SDA=GPIO21, SCL=GPIO22)
- I2C Address: 0x77

Specifications:

- Pressure range : 300 to 1100 hPa
- Pressure accuracy: ±0.12 hPa
- Temperature range: -40°C to +85°C
- Temperature accur: ±0.5°C
- Supply voltage : 1.8V – 3.6V (3.3V from ESP32)

Role in This Project:

- Monitors atmospheric conditions around solar installation
- Cross-checks temperature with DHT11
- Detects environmental anomalies (storms, pressure drops)
- Altitude can be derived from pressure if needed

Failsafe:

- If BMP180 not found at boot→ system continues without it
- LCD shows "BMP180: No Sensor / Check I2C wiring"

LCD Display:

Page 5 → "Pres: 1013.2 hPa" / "Temp: 31.5°C BMP"

Blynk:

V7 → Pressure (hPa)

V8 → BMP Temperature (°C)

11. LCD 16x2 I2C — Display Pages

Interface : I2C (SDA=GPIO21, SCL=GPIO22)

I2C Address : 0x27 (or 0x3F)

Page Switch : Every 3 seconds (automatic cycling)

Gas Alert Override (any page):

Line 1: ⚠ MQ2 DETECTED! /

⚠ MQ135 DETECTED! /

⚠ MQ2 + MQ135 !!

Line 2: Smoke/LPG/CO !! / Air Pollution!!

Action: LCD backlight flashes until gas clears

Page	Parameter	Value	Description
Page 1	Solar Voltage	6.54 V	Voltage from solar panel
	Solar Current	0.432 A	Current produced by solar panel
Page 2	Solar Power	2.82 W	Total power generated
	Array Configuration	2S2P	Solar panel configuration (2 series × 2 parallel)
Page 3	Temperature	34.5 °C	Measured by DHT11 sensor
	Humidity	65.0 %	Measured by DHT11 sensor
Page 4	MQ2 Gas Status	Safe	Gas leakage detection status
	MQ135 Air Quality	Safe	Air quality level
Page 5	Pressure	1013.2 hPa	Atmospheric pressure from BMP180
	Temperature (BMP180)	31.5 °C	Temperature from BMP180 sensor

12. Blynk IoT — Remote Monitoring Dashboard

Platform : Blynk IoT (blynk.cloud)

Template : "QuickStart Device"

Connection : WIFI (ESP32 → Router → Blynk Cloud → Mobile App)

Push Notification:

- Event name: "gas_alert"
- Triggered when: mq2Alert OR mq135Alert = true
- Message: "Gas/Smoke Detected! Check area immediately."

Data Send Rate : Every 5 seconds

Reconnect : Auto-retry every 30 seconds if disconnected

Offline Mode : System runs locally on LCD if WiFi unavailable

Virtual Pin Map:		
Pin	Data	Widget Type
V0	Solar Voltage (V)	Gauge
V1	Solar Current (A)	Gauge
V2	Solar Power (W)	SuperChart
V3	DHT11 Temperature (°C)	Gauge
V4	DHT11 Humidity (%)	Gauge
V5	MQ2 Status (text)	Label
V6	MQ135 Status (text)	Label
V7	BMP180 Pressure (hPa)	Gauge
V8	BMP180 Temperature (°C)	Gauge
V9	Overall Gas Alert	Label

13. ESP32 Pin Connection Summary

Important Notes:

- GPIO 34, 35, 36, 39 are INPUT ONLY — do not drive them as output
- ADC2 pins (GPIO 0, 2, 4, 12–15, 25–27) conflict with WiFi — avoid for analog
- BMP180 and LCD safely share the same I2C bus (different I2C addresses)
- All analog sensors use ADC1 to avoid WiFi conflicts
- ADC attenuation set to 11dB for full 0–3.3V range

Component	ESP32 GPIO	Notes
Voltage Divider	GPIO 34	ADC1 — input only
ACS712 AOUT	GPIO 35	ADC1 — input only
DHT11 DATA	GPIO 4	Digital, single-wire communication
MQ2 AOUT	GPIO 32	ADC1 analog input
MQ135 AOUT	GPIO 33	ADC1 analog input
BMP180 SDA	GPIO 21	I2C shared with LCD
BMP180 SCL	GPIO 22	I2C shared with LCD
LCD SDA	GPIO 21	I2C shared with BMP180
LCD SCL	GPIO 22	I2C shared with BMP180

-

14. Software & Libraries Used

Libraries (install via Arduino Library Manager):

Development Environment:

- Arduino IDE (version 2.x recommended)
- ESP32 Board Package by Express if Systems

Key Code Features:

- Non-blocking WiFi connection with timeout (no infinite hang)
- WiFi network scan before connecting (shows exact error reason)
- Auto-reconnect if WiFi drops (retries every 30 seconds)
- 20-sample averaging for stable ADC readings
- BMP180 graceful skip if not detected at boot
- Gas alert overrides LCD with flashing backlight warning

Library	Purpose
Blynk (by Volodymyr Shymanskyy)	IoT cloud communication
LiquidCrystal_I2C (by de Brabander)	LCD 16×2 I2C display control
DHT sensor library (Adafruit)	DHT11 temperature and humidity measurement
Adafruit BMP085 (Adafruit)	Interface for BMP180 pressure sensor
Adafruit Unified Sensor	Dependency library required for BMP085
WiFi.h (built-in ESP32)	WiFi connectivity for ESP32
Wire.h (built-in ESP32)	I2C communication protocol

15. Conclusion

This project successfully demonstrates a complete, low-cost industrial solar monitoring system using a single ESP32 microcontroller integrated with six sensor modules and dual display methods — local LCD and remote Blynk cloud dashboard.

Key Achievements:

- Real-time monitoring of 4-panel 2S2P solar array (voltage, current, power)
- Continuous environmental sensing (temperature, humidity, pressure)
- Intelligent gas detection with human-readable string status for MQ2 and MQ135
- Automatic mobile push notifications on gas detection

- Robust WiFi handling with auto-reconnect and full offline fallback
- Clean, structured code with 20-sample ADC averaging for stable readings

The system is deployable in real industrial environments and can be extended with minimal changes to support larger arrays, additional sensors, or multi-node networks using ESP-NOW communication.

16. References

1. Espressif Systems — ESP32 Technical Reference Manual
2. Blynk IoT Documentation https://docs.blynk.io/
3. Allegro Microsystems — ACS712 Datasheet
https://www.allegromicro.com/en/products/sense/current-sensor-ics/zero-to-fifty- amp-integrated-conductor-sensor-ics/acs712
4. Bosch Sensortec — BMP180 Datasheet
https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf
5. Aosong Electronics — DHT11 Datasheet
https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf
6. Hanwei Electronics — MQ2 Sensor Datasheet
https://www.pololu.com/file/0J309/MQ2.pdf
7. Hanwei Electronics — MQ135 Sensor Datasheet
https://www.electronicoscaldas.com/datasheet/MQ-135_Hanwei.pdf
8. Arduino Reference — analogRead(), analogSetAttenuation()
https://www.arduino.cc/reference/en/
9. Adafruit Industries — BMP085/BMP180 Library Documentation
https://github.com/adafruit/Adafruit-BMP085-Library