BMP180 Sensor Complete Manual

Overview

The BMP180 is a **digital barometric pressure and temperature sensor** from Bosch. It's used in **weather stations, altimeters, drones, and IoT devices**.

Key Features:

- Measures pressure: 300–1100 hPa
- Measures temperature: -40°C to +85°C
- I2C digital interface
- Low power consumption
- Small and compact

Pinout

Pin	Function	Description
VIN	Power Supply	3.3V or 5V (depending on breakout)
GND	Ground	0V reference
SCL	Serial Clock	I2C clock line
SDA	Serial Data	I2C data line
EOC	End of Conversion	Optional interrupt (on some versions)

Wiring with Arduino

Example (3.3V or 5V logic tolerant breakout):

BMP180	Arduino UNO
VIN	3.3V / 5V
GND	GND
SCL	A5
SDA	A4

Note: For STM32, use the appropriate I2C pins (SCL/SDA) and ensure 3.3V logic compatibility.

Software Libraries

- **Arduino:** `Adafruit_BMP085` (compatible with BMP180)
- **STM32 (HAL/LL):** Use I2C read/write functions or libraries like `BMP180.h`.

Arduino Example Code

```
#include <Wire.h>
#include <Adafruit_BMP085.h>  // or BMP180 library

Adafruit_BMP085 bmp;

void setup() {
  Serial.begin(9600);
  if (!bmp.begin()) {
    Serial.println("BMP180 sensor not found!");
    while(1);
  }
}

void loop() {
  Serial.print("Temperature: ");
  Serial.print(bmp.readTemperature());
  Serial.println(" C");

  Serial.print("Pressure: ");
  Serial.print(bmp.readPressure());
  Serial.println(" Pa");

  float seaLevelPressure = 100400; // replace with local value in Pa
  float altitude = bmp.readAltitude(seaLevelPressure);

  Serial.print("Altitude: ");
  Serial.print(altitude);
  Serial.println(" m");

  delay(1000);
}
```

Understanding Altitude Calculation

Formula:

$$\text{Altitude} = 44330 \times (1 - (\frac{P}{P_0})^{0.1903})$$

- **P** = measured pressure
- **P0** = reference sea-level pressure (default: 101325 Pa)

Tips:

- Use **local sea-level pressure** for accurate altitude.
- For relative altitude (e.g., drones), default P0 is fine.
- BMP180 is sensitive to weather changes—expect ±1–2 m variation.

Calibrating BMP180 Altitude

Step 1: Measure local pressure

- Check a weather website for current sea-level pressure.

Step 2: Adjust P0

```
float localSeaLevel = 100400; // Example value in Pa
float altitude = bmp.readAltitude(localSeaLevel);
```

Step 3: Verify

- Stand at known altitude and adjust until reading matches.

Calculating Relative Altitude

- Useful for detecting **up/down movement**
- Formula:

$$\Delta h = 44330 \times (1 - (\frac{P_{\text{current}}}{P_{\text{initial}}})^{0.1903})$$

- Gives **change in altitude** without knowing exact P0.

Tips for Accurate Measurement

1. Avoid **rapid temperature changes**; let sensor stabilize.
2. Keep sensor **away from heat sources** (motors, sun).
3. Use **shielded wires** for I2C to avoid noise.
4. Ensure **proper pull-up resistors** (4.7 kΩ–10 kΩ) on SDA/SCL lines.

Applications

- Weather stations
- Drones & UAVs (altimeter)
- Indoor navigation systems
- Smartphones & wearables
- IoT environmental monitoring

Common Issues

Issue	Cause	Solution
Wrong altitude	Default P0 used	Input local sea-level pressure
I2C not detected	Wiring / Pull-ups	Check SDA/SCL, add pull-ups
Fluctuating readings	Environmental changes	Add filter / average readings
Negative altitude	Measured pressure > P0	Check P0 value

Summary

- BMP180 is **simple, accurate, and small**.
- Absolute altitude requires **sea-level calibration**.
- Relative altitude and pressure monitoring are **very reliable**.
- Compatible with **Arduino, STM32, ESP32, and other MCUs**.
