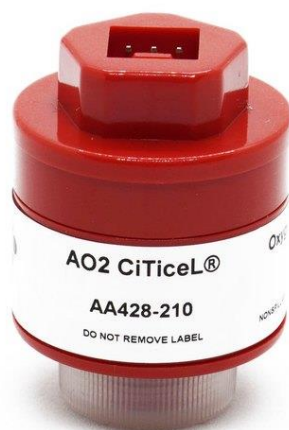


## Introdução

O sensor de oxigênio **CiticeL AO2** é um exemplo de sensor eletroquímico amplamente utilizado em sistemas embarcados para análise de gases. Uma característica marcante deste sensor é que ele não possui **VCC**, ou seja, não requer uma tensão externa para seu funcionamento. Isso ocorre porque o sensor gera uma alimentação interna de **7mV** devido às reações químicas em seu interior.

Por ser eletroquímico, o AO2 possui uma vida útil limitada a aproximadamente **2 anos** em condições normais de ar fresco. Seu tempo de resposta para atingir até 90% de precisão em uma leitura é de apenas **5 segundos**.

*Imagem ilustrativa do sensor AO2:*



## Amplificação e Conversão

Para aumentar a confiabilidade das leituras do AO2, ele foi acoplado a um conversor **ADS1115**, que se comunica com o Raspberry Pi via **I<sup>2</sup>C** utilizando o endereço **0x48**. O ADS1115 amplifica o sinal analógico gerado pelo AO2, que é muito baixo e instável, para um sinal mais alto e estável. Além disso, o conversor realiza a digitalização do sinal, facilitando a leitura pelo microcontrolador.

*Imagem ilustrativa do conversor ADS1115:*



No código, o ADS1115 é configurado para operar com o ganho máximo de **16x**, o que permite trabalhar em uma faixa de tensão de **0,256V**. Essa configuração é ideal para amplificar a tensão interna do AO2, que é de apenas **10mV**.

## Instalação de Dependências

Antes de utilizar o sensor no Raspberry Pi, é necessário configurar o ambiente Python, instalar as bibliotecas necessárias e ativar a comunicação I<sup>2</sup>C:

- Clique no ícone do Raspberry Pi (imagem de uma framboesa).
- Navegue até **Preferências** → **Configuração do Raspberry Pi**.
- Na aba **Interface**, ative a opção **I<sup>2</sup>C**.
- Reinicie o Raspberry Pi para aplicar as configurações.

Em seguida abra o terminal e execute os seguintes comandos;

### 1. Atualize o sistema:

```
sudo apt update  
sudo apt install python3-pip
```

### 2. Instale a biblioteca do ADS1115:

```
pip3 install adafruit-circuitpython-ads1x15
```

## Exemplo de Código

O código abaixo demonstra como configurar o ADS1115 e ler os valores do sensor AO2 em python:

```
import busio
import board
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn

# Inicializa I2C
i2c = busio.I2C(board.SCL, board.SDA)

# Cria o objeto ADC usando o bus I2C
ads = ADS.ADS1115(i2c)

# Configura o ganho máximo (16x)
ads.gain = 16

# Cria o objeto AnalogIn no canal 0 (single-ended)
chan = AnalogIn(ads, ADS.P0)

# Função de leitura do sensor de oxigenio
def le_sensor_o2():
    valor_o2 = (0.0167 * chan.value)
    return valor_o2
```

Quando a função `le_sensor_o2()` for chamada, ela retornará o valor do sensor de oxigênio em porcentagem.

## Calibração e Compensação

### Calibração

- Um sensor novo geralmente retorna valores entre **20%** e **23%** de O<sub>2</sub>.
- O valor típico do ar fresco é de **20,90%** de O<sub>2</sub>.
- Para obter leituras precisas, é necessário calibrar o sensor em uma (calibração em um ponto) ou duas (calibração em dois pontos) das seguintes condições:
  - **Ar fresco:** Ajuste para refletir o nível de O<sub>2</sub> na atmosfera padrão.

- **Zero (Nitrogênio):** Calibração em ambiente sem oxigênio para determinar o offset.
- **Gás conhecido:** Calibração com uma mistura de gás com concentração definida de  $O_2$ .

### ***Compensação de Pressão***

O sensor AO2 não realiza compensação de pressão internamente. De acordo com o fabricante:

- Um aumento de **10% na pressão** resulta em uma leitura **10% maior** de  $O_2$ .
- Uma redução de **10% na pressão** resulta em uma leitura **10% menor** de  $O_2$ .

Portanto, fórmulas e lógicas devem ser implementadas para ajustar os valores de  $O_2$  com base na pressão medida por um sensor adicional, como o **BME280**.

Com essas configurações e ajustes, o sensor AO2 pode fornecer leituras precisas e confiáveis de oxigênio, sendo ideal para aplicações em sistemas embarcados de análise de gases.