

Sensores y Actuadores

TRABAJO PRÁCTICO N° 4

Profesores: Gonzalo Vera y Jorge Morales

Grupo Nro 3

Alumno: Dario Arriola

Ejercicio 1 e)

Como implementaría un sensor inteligente de altitud?

Barómetro digital.

Un barómetro digital es un dispositivo que mide la presión del aire y que puede usarse como altímetro. Podemos conectar este sensor a un autómata o procesador como Arduino para registrar la medición de la presión del aire o estimar la altitud del sensor respecto al nivel del mar.

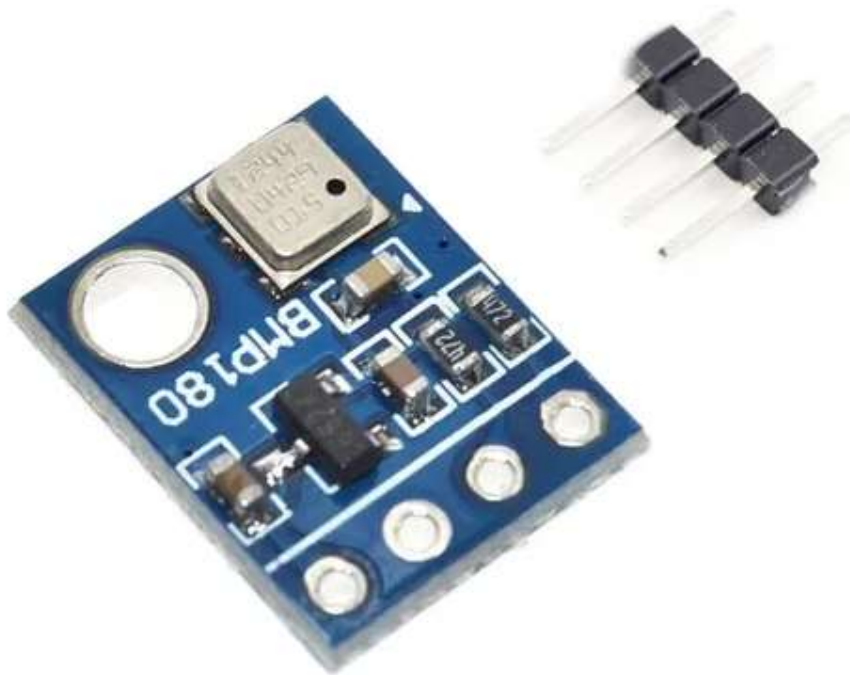
La presión barométrica está originada por el peso de la columna del aire de la atmósfera. La presión barométrica depende de diversos factores, especialmente de la temperatura, ya que esta influye en la densidad del aire y, por tanto, en el peso de la columna de aire. Otros factores que afectan a la medición son la humedad y el viento. El sensor a utilizar es, por sus siglas BaroMetric Pression, es el BMP180, que es un sensor barométrico digital, que incorpora un sensor de temperatura para compensar sus efectos en la medición de la presión barométrica. A su vez, el BMP180 es una versión mejorada del BMP085, siendo ambos modelos compatibles entre si tanto en hardware como software.

El BMP180 es un sensor de alta precisión y baja potencia. El rango de medición es de 300hPa a 1110 hPa, equivalente a una altitud de -500m a 9000m sobre el nivel del mar. La precisión es configurable, desde 0.06hPa (0.5 metros) en el modo de bajo consumo, a 0.02hPa (0.17 metros) en el modo de alta precisión. La comunicación se realiza a través del bus I2C, por lo que es sencillo obtener los datos medidos. La tensión de alimentación es de bajo voltaje entre 1.8 a 3.6V. Frecuentemente se encuentran integrados en módulos como la GY-68 que incorporan la electrónica necesaria para conectarla de forma sencilla a un Arduino. En la mayoría de los módulos, estos incluye un regulador de voltaje que permite alimentar directamente a 5V.

El consumo promedio es de $0.1\mu\text{A}$ en stand-by, y $650\mu\text{A}$ durante la medición, lo que supone un consumo promedio de $5\mu\text{A}$ tomando 1 muestra por segundo en precisión estándar. El tiempo de respuesta es de unos 5ms en resolución estándar, y 17ms en alta resolución.

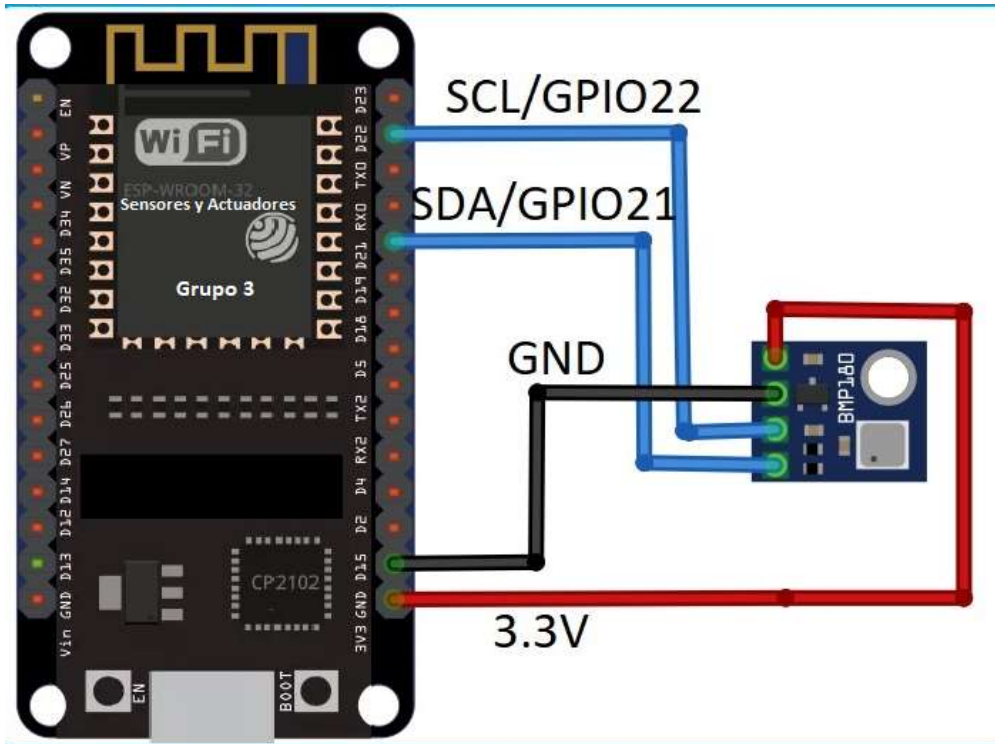
Los barómetros BMP180 y BMP085 son ampliamente empleados en aplicaciones meteorológicas, como estaciones registradoras, relojes que muestran el clima, o para predecir cuándo va a llover, entre otras. También pueden ser empleadas en aplicaciones de climatización o control de ventilación. La capacidad para estimar alturas diferenciales también hace que sean usados en vehículos aéreos como aviones o cuadricopteros, o en aplicaciones que requieran medir la velocidad de desplazamiento vertical, por ejemplo, el ascenso de una máquina o el tiempo de caída de un dispositivo.

El precio del sensor BMP180 ronda los ar\$600 (usd 4.-)



ESQUEMA MONTAJE

La conexión es sencilla, simplemente alimentamos el módulo desde el ESP32 mediante GND y 3V y conectamos el pin SDA y SCL con los pines correspondientes del barómetro MP180.



OBTENER LOS VALORES DE PRESIÓN Y TEMPERATURA

El primer ejemplo, obtiene la medición de los valores en bruto (RAW) de presión y temperatura, y los muestra en la pantalla. Estos valores son de utilidad, por ejemplo, para hacer una estación meteorológica.

```
#include <SFE_BMP180.h>

#include <Wire.h>

SFE_BMP180 bmp180;

void setup()
{
  Serial.begin(9600);
  if (bmp180.begin())
    Serial.println("BMP180 iniciado");
  else
```

```
{  
  Serial.println("Error al iniciar BMP180");  
  while(1); // bucle infinito  
}  
}  
  
void loop()  
{  
  char status;  
  double T,P;  
  status = bmp180.startTemperature(); //Inicio de lectura de temperatura  
  if (status != 0)  
  {  
    delay(status); //Pausa para que finalice la lectura  
    status = bmp180.getTemperature(T); //Obtener la temperatura  
    if (status != 0)  
    {  
      status = bmp180.startPressure(3); //Inicio lectura de presión  
      if (status != 0)  
      {  
        delay(status); //Pausa para que finalice la lectura  
        status = bmp180.getPressure(P,T); //Obtenemos la presión  
        if (status != 0)  
        {  
          Serial.print("Temperatura: ");  
          Serial.print(T,2);  
          Serial.print(" *C , ");  
          Serial.print("Presion: ");
```

```
Serial.print(P,2);

Serial.println(" mb");

}

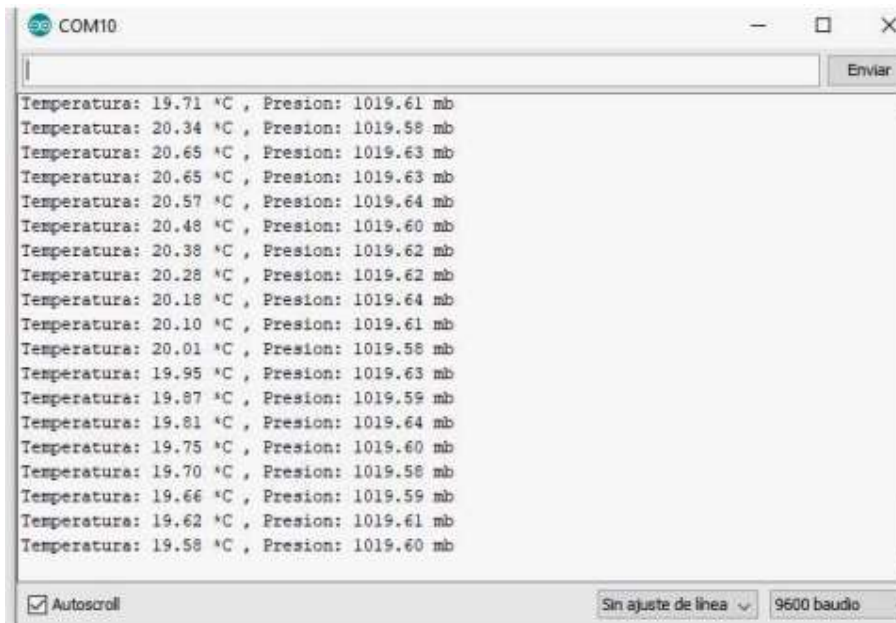
}

}

}

delay(1000);

}
```



ESTIMAR LA ALTITUD RESPECTO AL MAR

En el siguiente ejemplo estimamos la altitud respecto del mar. Para ello tomamos como valor de referencia la presión estandar para sobre el nivel del mar.

```
#include <SFE_BMP180.h>
```

```
#include <Wire.h>
```

```
SFE_BMP180 bmp180;
```

```
double PresionNivelMar = 1013.25; //presión sobre el nivel del mar en mbar
```

```
void setup()
{
  Serial.begin(9600);
  if (bmp180.begin())
    Serial.println("BMP180 iniciado");
  else
  {
    Serial.println("Error al iniciar el BMP180");
    while(1);
  }
}

void loop()
{
  char status;
  double T,P,A;
  status = bmp180.startTemperature(); //Inicio de lectura de temperatura
  if (status != 0)
  {
    delay(status); //Pausa para que finalice la lectura
    status = bmp180.getTemperature(T); //Obtener la temperatura
    if (status != 0)
    {
      status = bmp180.startPressure(3); //Inicio lectura de presión
      if (status != 0)
      {
        delay(status); //Pausa para que finalice la lectura
```

```
status = bmp180.getPressure(P,T); //Obtener la presión
if (status != 0)
{
  Serial.print("Temperatura: ");
  Serial.print(T);
  Serial.print(" *C , ");
  Serial.print("Presion: ");
  Serial.print(P);
  Serial.print(" mb , ");
  A= bmp180.altitude(P,PresionNivelMar); //Calcular altura
  Serial.print("Altitud: ");
  Serial.print(A);
  Serial.println(" m");
}
}
}
}
delay(1000);
}
```