

1.e) Como implementaría un sensor inteligente de altitud?

¿QUÉ ES UN BMP180?

El BMP180 es un termómetro y barómetro digital. Funcionando como barómetro, además, tenemos un **dispositivo que mide la presión del aire y que puede usarse como altímetro**. Podemos conectar este sensor a un autómatas 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.

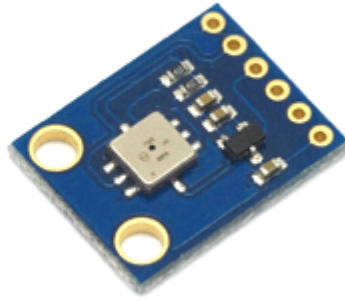
Dado que la presión barométrica depende de la altura de la columna de aire ubicada sobre el sensor también **puede ser empleada para estimar la altitud** a la que está ubicado el sensor respecto al nivel del mar.

Debemos tener en cuenta que la presión barométrica varía continuamente debido a las condiciones climatológicas por lo que **no proporciona una medición absoluta de la altitud** con precisión. Sin embargo, sí puede ser de utilidad en **mediciones diferenciales de altitud**, es decir, diferencias de cota en un desplazamiento vertical.

El BMP180 es un termómetro y barométrico digital, que emplea su 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 sí tanto en hardware como software.

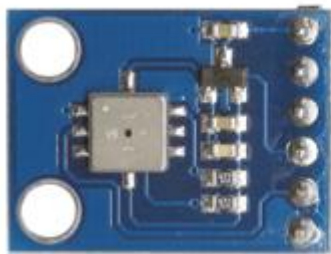
El BMP180 es un sensor de **alta precisión y baja potencia**. El rango de medición es de 300hPa a 1110 hPa, equivalente a una altitud de -500m a 9000m sobre el nivel del mar. La precisión absoluta es de 1.0 hPa, y la relativa de 0.12 hPa, equivalente a una precisión en altitud de aproximadamente 1m.

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



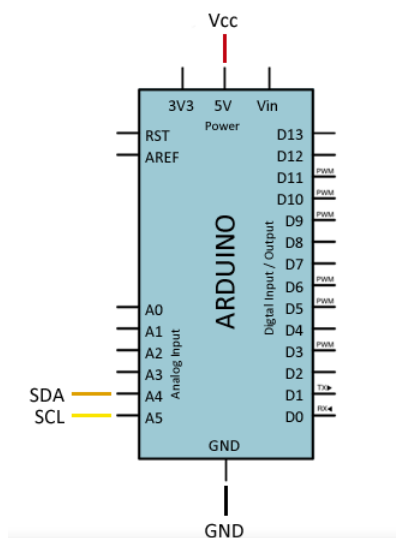
ESQUEMA MONTAJE

La conexión es sencilla, simplemente alimentamos el módulo desde Arduino mediante GND y 5V y conectamos el pin SDA y SCL de Arduino con los pines correspondientes del barómetro BMP180.



GND	—	GND
Vcc	—	5V
SDA	—	SDA (A4)
SCL	—	SCL (A5)

Mientras que la conexión vista desde el lado de Arduino quedaría así.



En Arduino Uno, Nano y Mini Pro, SDA es el pin A4 y el SCL el pin A5. Para otros modelos de Arduino consultar el [esquema patillaje](#) correspondiente.

Verificar que la placa que utilice sea compatible con 5V antes de conectarla a Arduino. Si no, tendréis que usar un adaptador de nivel lógico.

EJEMPLOS DE CÓDIGO

Para realizar la lectura del BMP180 usaremos la librería desarrollada por Sparkfun, disponible en [este enlace](#).

La librería proporciona ejemplos de código, que resulta aconsejable revisar. Los siguientes ejemplos, por ejemplo, son modificaciones están basados a partir de los disponibles en la librería.

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 estándar para sobre el nivel del mar.

```
#include <SFE_BMP180.h>

#include <Wire.h>

SFE_BMP180 bmp180;

double PresionNivelMar = 1013.25; //presion 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);

}

```

ESTIMAR LA DIFERENCIA DE ALTURA ENTRE DOS PUNTOS

En el último ejemplo, realizamos la medición de la diferencia de altura entre dos puntos. Simplemente medimos la diferencia de cotas entre dos puntos de presión, pero en lugar de tomar la referencia a nivel del mar tomamos como referencia el primer punto.

```

#include <SFE_BMP180.h>

#include <Wire.h>

SFE_BMP180 bmp180;

double Po; //presion del punto inicial para h=0;

char status;

double T,P,A;

void setup()

```

```
{

    Serial.begin(9600);

    if (bmp180.begin())

    {

        Serial.println("BMP180 iniciado");

        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)
```

```

        {

            Po=P; //Asignamos el valor de presión como punto de
referencia

            Serial.println("Punto de referencia establecido: h=0");

        }

    }

}

}

}

}

else

{

    Serial.println("Error al iniciar el BMP180");

    while(1);

}

}

void loop()

{

    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)

            {

                A= bmp180.altitude(P,Po); //Calcular altura con respecto al
                punto de referencia

                Serial.print("h=");

                Serial.print(A);

                Serial.println(" m");

            }

        }

    }

}

```

```
    }  
  
    }  
  
    delay(1000);  
  
}
```