

Tecnicatura Superior en Telecomunicaciones

Materia: Práctica sensores y actuadores

Profesor: C. GONZALO VERA

Profesor: JORGE E. MORALES

Tema: Práctica Semana 4

Ciclo lectivo: 2022

Alumnos : Grupo 6

- Guzmán, Lilén <https://github.com/lilenguzman01>
- López, Maximiliano <https://github.com/Maxilopez28>
- Moyano, Emilio <https://github.com/TerraWolf>
- Muguruza, Sergio <https://github.com/sergiomuguruza>
- Gonzalez, Mario <https://github.com/mariogonzalezispc>
- Ripoli, Enrique <https://github.com/enriqueripoli>

Fotos armado del prototipo :

[Fotos](#)

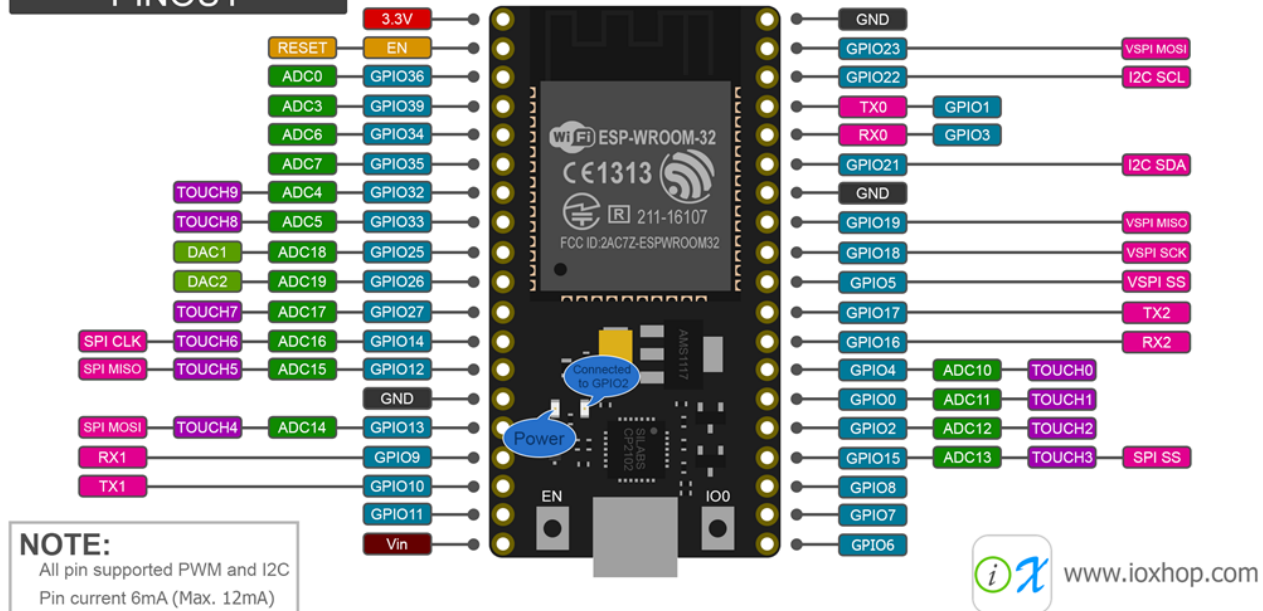
<https://photos.app.goo.gl/sVMHbST6cKu5yyVJ6>

Ejercicio 1-e

Implementación de una placa nodeMCU esp32-S con dos sensores de presión , el BMP180 y el BMP280. Conexión por I2C. y publicación por MQTT broker mosquito dashboard NODE-RED hosting de IBM BLueMIX

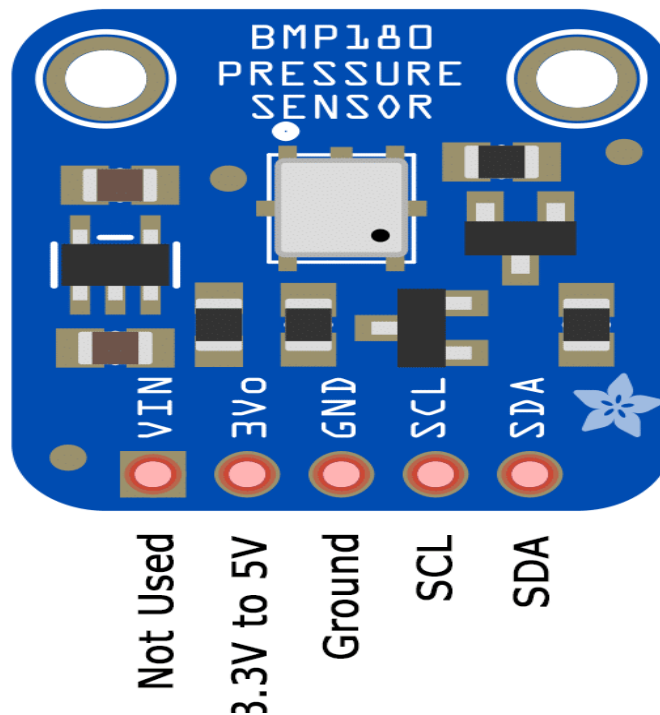
NodeMCU-32S

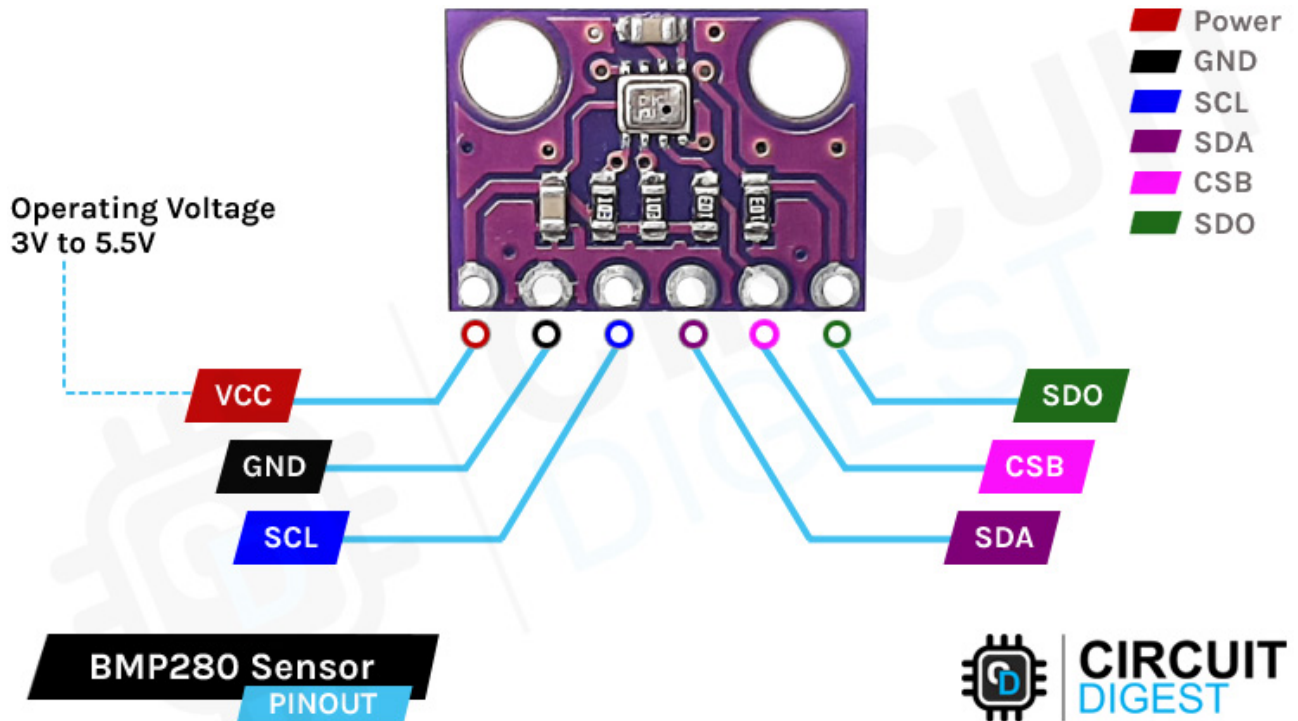
PINOUT



Sensor BMP180

Presión





Código para probar el sensor BMP280 y calcular la altitud aproximada.

```
Ejercicio 1 - Ejercicio_1-e.ino

1  /*
2   * Materia sensores y actuadores
3   * Grupo 6
4   * Alumno : Mario Gonzalez
5   * Ejercicio 1-e
6   */
7  #include <Wire.h>           // incluimos libreria de comunicacion I2C nativa
8  #include <Adafruit_BMP280.h> // incluimos libreria de adafruit para sensor BMP280
9
10 Adafruit_BMP280 bmp; // creamos el objeto
11
12 void setup()
13 {
14   Serial.begin(9600);           // iniciamos el serial nativo
15   bmp.begin(BMP280_ADDRESS_ALT, BMP280_CHIPID); // iniciamos el objeto sensor en la direccion alterna 0x77
16
17   /* configurar de fabrica segun datasheet ADAFRUIT Library */
18   bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* Operating Mode. */
19                  Adafruit_BMP280::SAMPLING_X2, /* Temp. oversampling */
20                  Adafruit_BMP280::SAMPLING_X16, /* Pressure oversampling */
21                  Adafruit_BMP280::FILTER_X16, /* Filtering. */
22                  Adafruit_BMP280::STANDBY_MS_500); /* Standby time. */
23   pinMode(LED_BUILTIN, OUTPUT); // el pin 2 como salida para el led azul de la placa
24 }
25
```

Ejercicio 1 - Ejercicio_1-e.ino

```
1  /*
2   * Materia sensores y actuadores
3   * Grupo 6
4   * Alumno : Mario Gonzalez
5   * Ejercicio 1-e
6   */
7  void loop()
8  {
9      Serial.print("Temperatura = ");           // imprimo por serial la palabra temperatura
10     Serial.print(bmp.readTemperature());       // recupero del objeto el metodo readTemperatura
11     Serial.println(" *C");                     // imprimo C de centigrado al final del renglon y hago CR/LF
12     Serial.print("Presion = ");                // imprimo por serial la palabra Presion
13     Serial.print(bmp.readPressure() / 100);    // recupero del objeto el metodo readPressure y divido para Hectopascal
14     Serial.println(" hPa");                    // imprimo Hecto pascal
15     Serial.print("Altitud = ");                // imprimo por serial la palabra Altitud
16     Serial.print(bmp.readAltitude(1013.25));  // recupero del objeto el metodo readAltitude e instancio
17     Serial.println(" m");
18     Serial.println();                          // imprimo CRyLF
19     digitalWrite(LED_BUILTIN, HIGH); // Enciendo led azul
20     delay(100);
21     digitalWrite(LED_BUILTIN, LOW); // apago led azul
22     delay(100);
23     digitalWrite(LED_BUILTIN, HIGH); // Enciendo led azul
24     delay(100);
25     digitalWrite(LED_BUILTIN, LOW); // apago led azul
26     delay(3000);
27 }
```


Implementación prototipo

