|  |
| --- |
| **Step-1: Test DHT-22 and BMP-280 Sensor** |
| **ชื่อ-สกุล: B6304577 นายภานุพงศ์ แคนอินทร์** |

1. DHT-22 🡪 <https://www.cybertice.com/p/1045>

* Read Temperature 🡪 DHT\_Tempp
* Read Humidity 🡪 DHT\_Humid

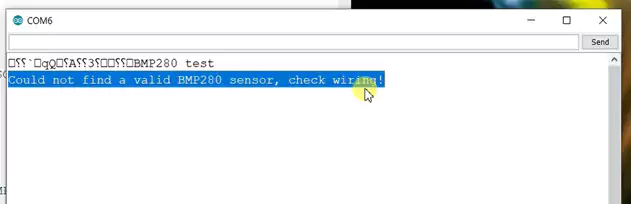
|  |
| --- |
| **Code ที่ใช้**  #include "DHT.h"  #define DHTPIN 4 // Digital pin connected to the DHT sensor  #define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321  DHT dht(DHTPIN, DHTTYPE);  void setup() {  Serial.begin(115200);  Serial.println(F("DHT22 test!"));  dht.begin();  }  void loop() {  // Wait a few seconds between measurements.  delay(2000);  // Reading temperature or humidity takes about 250 milliseconds!  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)  float h = dht.readHumidity();  // Read temperature as Celsius (the default)  float t = dht.readTemperature();  // Read temperature as Fahrenheit (isFahrenheit = true)  float f = dht.readTemperature(true);  // Check if any reads failed and exit early (to try again).  if (isnan(h) || isnan(t) || isnan(f)) {  Serial.println(F("Failed to read from DHT sensor!"));  return;  }  // Compute heat index in Fahrenheit (the default)  float hif = dht.computeHeatIndex(f, h);  // Compute heat index in Celsius (isFahreheit = false)  float hic = dht.computeHeatIndex(t, h, false);  Serial.print(F("Humidity: "));  Serial.print(h);  Serial.print(F("% Temperature: "));  Serial.print(t);  Serial.print(F("°C "));  Serial.print(f);  Serial.print(F("°F Heat index: "));  Serial.print(hic);  Serial.print(F("°C "));  Serial.print(hif);  Serial.println(F("°F"));  } |
| ผลการทดสอบ 1  A picture containing text, software, screenshot, web page  Description automatically generated |
| ผลการทดสอบ 2  A picture containing electronics, electronic engineering, circuit component, text  Description automatically generated |

1. How to use BMP-280 🡪 https://www.circuitschools.com/interfacing-bmp280-with-esp-32-on-i2c-with-errors-and-solutions/

* Read Temperature 🡪 BMP\_Tempp
* Read Pressure 🡪 BMP\_Press
* Read Altitude 🡪 BMP\_Altit

|  |
| --- |
| Code ที่ใช้  #include <Adafruit\_BMP280.h>  #define BMP\_SCK (13)  #define BMP\_MISO (12)  #define BMP\_MOSI (11)  #define BMP\_CS (10)  Adafruit\_BMP280 bmp; // I2C  //Adafruit\_BMP280 bmp(BMP\_CS); // hardware SPI  //Adafruit\_BMP280 bmp(BMP\_CS, BMP\_MOSI, BMP\_MISO, BMP\_SCK);  void setup() {  Serial.begin(9600);  Serial.println(F("BMP280 Forced Mode Test."));  //if (!bmp.begin(BMP280\_ADDRESS\_ALT, BMP280\_CHIPID)) {  if (!bmp.begin(0x76)) {  Serial.println(F("Could not find a valid BMP280 sensor, check wiring or "  "try a different address!"));  while (1) delay(10);  }  /\* Default settings from datasheet. \*/  bmp.setSampling(Adafruit\_BMP280::MODE\_FORCED, /\* Operating Mode. \*/  Adafruit\_BMP280::SAMPLING\_X2, /\* Temp. oversampling \*/  Adafruit\_BMP280::SAMPLING\_X16, /\* Pressure oversampling \*/  Adafruit\_BMP280::FILTER\_X16, /\* Filtering. \*/  Adafruit\_BMP280::STANDBY\_MS\_500); /\* Standby time. \*/  }  void loop() {  // must call this to wake sensor up and get new measurement data  // it blocks until measurement is complete  if (bmp.takeForcedMeasurement()) {  // can now print out the new measurements  Serial.print(F("Temperature = "));  Serial.print(bmp.readTemperature());  Serial.println(" \*C");  Serial.print(F("Pressure = "));  Serial.print(bmp.readPressure());  Serial.println(" Pa");  Serial.print(F("Approx altitude = "));  Serial.print(bmp.readAltitude(1013.25)); /\* Adjusted to local forecast! \*/  Serial.println(" m");  Serial.println();  delay(2000);  } else {  Serial.println("Forced measurement failed!");  }  } |
| ผลการทดสอบ 1  A screenshot of a computer  Description automatically generated |
| ผลการทดสอบ 2  A picture containing text, electronics, electronic engineering, circuit component  Description automatically generated |

!!! But unfortunately, we got an error, saying, “Could not find a valid bmp 280 sensor, check wiring.”



To solve this, first we need to find the, i2c address of the bmp sensor, with i2c scanner, like our previous tutorials, link for the code here. upload the code, and open serial monitor, to check the i2c address, where it was 0x76. note it down, and get back to our previous sample code, and type the address, in the sensor initialization line, as shown.

|  |
| --- |
| Code ที่ใช้  #include <Wire.h> //include Wire.h library    void setup()  {  Wire.begin(); // Wire communication begin  Serial.begin(9600); // The baudrate of Serial monitor is set in 9600  while (!Serial); // Waiting for Serial Monitor  Serial.println("\nI2C address Scanner");  }    void loop()  {  byte error, address; //variable for error and I2C address  int devicecount;    Serial.println("Scanning...");    devicecount = 0;  for (address = 1; address < 127; address++ )  {  // The i2c\_scanner uses the return value of  // the Write.endTransmisstion to see if  // a device did acknowledge to the address.  Wire.beginTransmission(address);  error = Wire.endTransmission();  if (error == 0)  {  Serial.print("I2C device found at address 0x");  if (address < 16)  Serial.print("0");  Serial.print(address, HEX);  Serial.println(" !");  devicecount++;  }  else if (error == 4)  {  Serial.print("Unknown error at address 0x");  if (address < 16)  Serial.print("0");  Serial.println(address, HEX);  }  }  if (devicecount == 0)  Serial.println("No I2C devices found\n");  else  Serial.println("done\n");  delay(5000); // wait 5 seconds for the next I2C scan  } |
| ผลการทดสอบ 1  A screenshot of a computer  Description automatically generated |

|  |
| --- |
| **Step-2: InfluxDB 🡪 Bucket Name = “B6304577”** |

1. <https://www.influxdata.com/get-influxdb/> 🡪 Login to influxDB Clound 2.0

|  |
| --- |
| A screenshot of a website  Description automatically generated with low confidence |

1. Register InfluxDB

|  |
| --- |
| A screenshot of a computer  Description automatically generated with medium confidenceA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Create Bucket

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Create Tokenkey

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

|  |
| --- |
| **Step-3: ESP32 Code** |

1. Add Lib

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Code write data to InfluxDB

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Monitor

|  |
| --- |
| A screenshot of a computer  Description automatically generated |

|  |
| --- |
| **Step-4: Grafana 🡺 Dashboard Name = “01”** |

1. <https://grafana.com/> 🡪 Create free account (or Sign in)

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Import Data

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated with medium confidenceA screenshot of a computer  Description automatically generated with medium confidenceA screenshot of a computer  Description automatically generated with medium confidenceA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Create Dashboard

|  |
| --- |
| A screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated with medium confidenceA screenshot of a computer  Description automatically generatedA screenshot of a computer  Description automatically generated |

1. Show 🡪 BMP280{Temperature, Pressure, Altitude} + DHT-11{Temperature, Humidity}

|  |
| --- |
|  |