

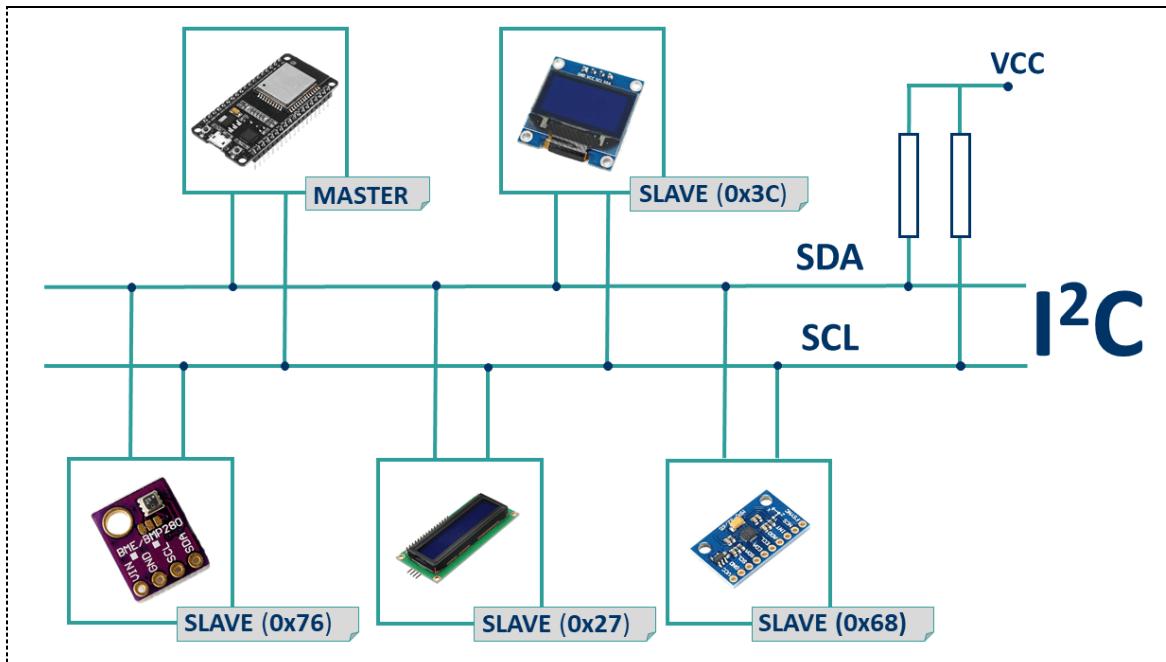
Getting Start ESP32: ESP32 GPIO + ESP32 Interface

Mission 9/12 – ESP32 + AHT20: I2C Temperature and Humidity Sensor Module

1. Read <https://www.glab.co.th/post/interface-the-series-ep-1-การใช้งาน-i2c-และตัวอย่างการอ่านค่าจากเซ็นเซอร์-i2c>

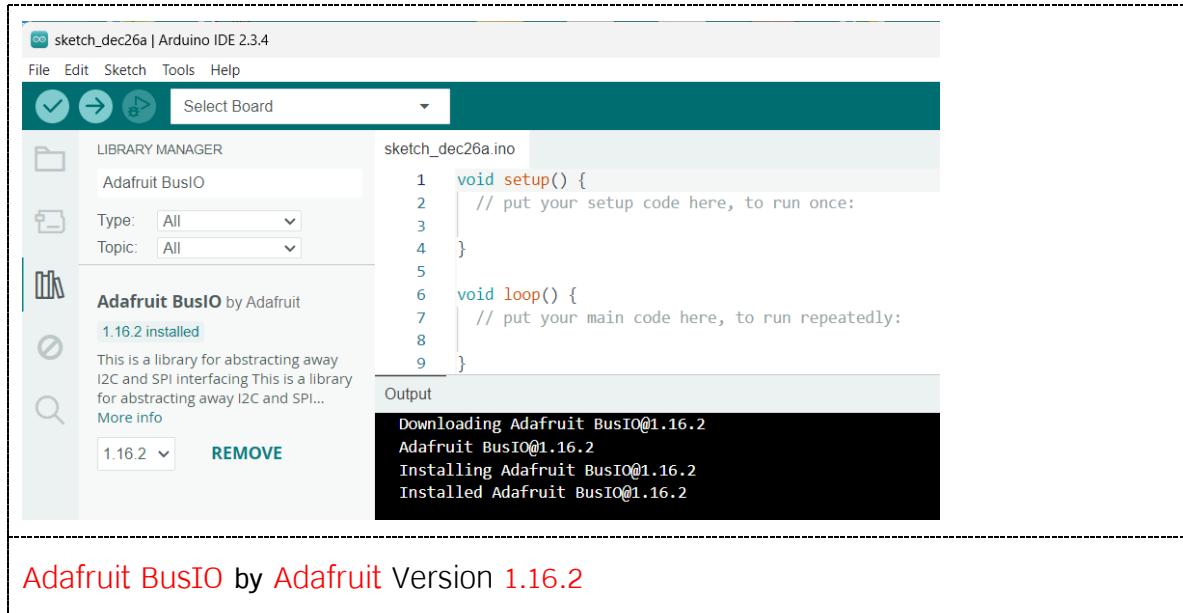
I2C เป็นรูปแบบการเชื่อมต่อ (interface) ที่นิยมใช้กันมากในฝั่งเทคโนโลยีสมองกลฝังตัว (Embedded System) และระบบ IoT โดยเซ็นเซอร์ต่าง ๆ เช่น เซ็นเซอร์อุณหภูมิและความชื้น เซ็นเซอร์แสง เซ็นเซอร์วัดผู้คน และเซ็นเซอร์อื่น ๆ ล้วนใช้ I2C ในการเชื่อมต่อกับไมโครคอนโทรลเลอร์

I2C (อ่านว่า ไอสแควร์ หรือ ไอทูซี) หรือ IIC ถูกพัฒนาโดยบริษัทฯ Philips Semiconductors หรือ NXP ในปัจจุบัน เป็นทั้งรูปแบบการเชื่อมต่อสาย และรูปแบบการสื่อสาร (Protocol) ทำงานแบบซิงโครไนซ์ (Synchrony) เชื่อมต่อแบบอนุกรม นิยมใช้กันมากในไอซีที่เป็นอุปกรณ์ต่อร่วมกับไมโครคอนโทรลเลอร์ เช่น ไอซี ADC, DAC, เซ็นเซอร์ต่าง ๆ, Digital Potentiometer ในเชิง Physical (การเชื่อมต่อสาย) ประกอบด้วยสายสัญญาณ 2 เส้น คือ SCL (Serial Clock) และ SDA (Serial Data) ทั้งผี้งไมโครคอนโทรลเลอร์ และผี้งอุปกรณ์ I2C โดยการเชื่อมต่อจะต้องเชื่อมต่อสายให้ชื่อตรงกัน คือ SCL เชื่อมต่อกับ SCL, SDA เชื่อมต่อกับ SDA โดยมีตัวต้านทานค่าตั้งแต่ 10k โอห์ม ถึง 1k โอห์ม ต่อ Pull-up ทั้ง SDA และ SCL ไว้

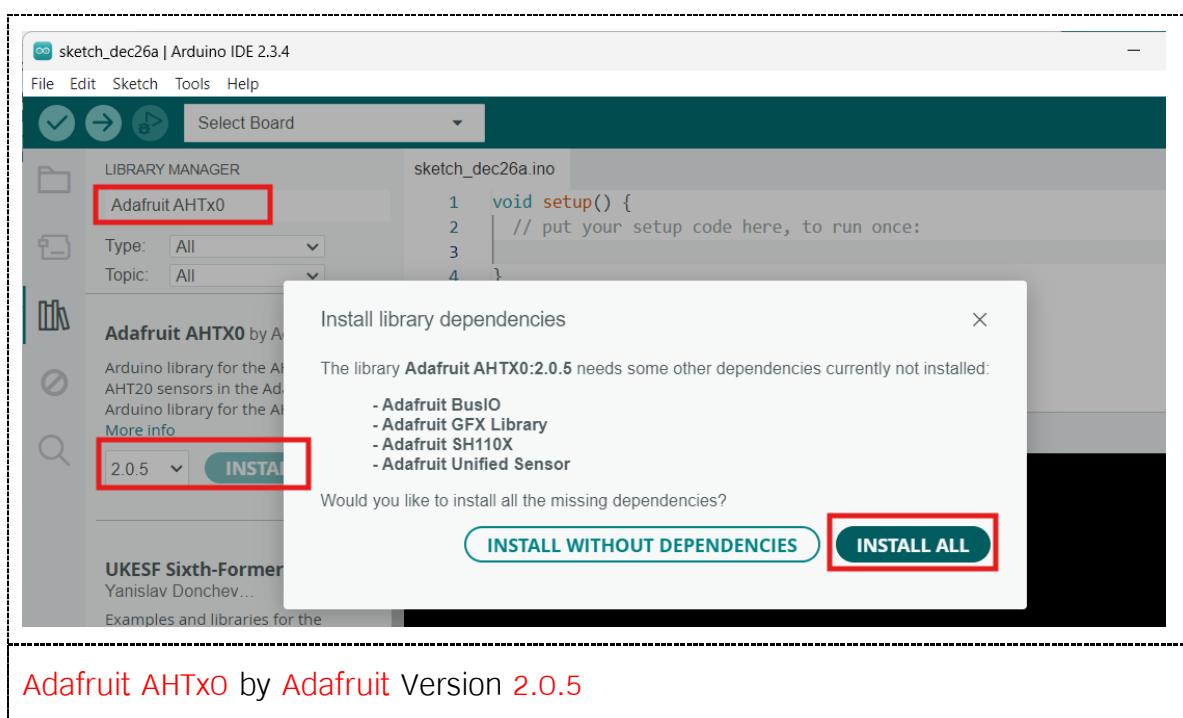


2. Read <https://www.cybertice.com/p/4451>
3. Read <https://www.allnewstep.com/p/4432>
4. Read <https://www.cybertice.com/b/423>

5. Add Library: Sketch → Include Library → Manage
6. Filter with “Adafruit BusIO”, Select Adafruit BusIO by Adafruit Version 1.16.2



7. Filter with “Adafruit AHTx0”, Select Adafruit AHTx0 by Adafruit Version 2.0.5



8. Test Code “Test01-Hello AHT20”

```
#include <Adafruit_AHTX0.h>

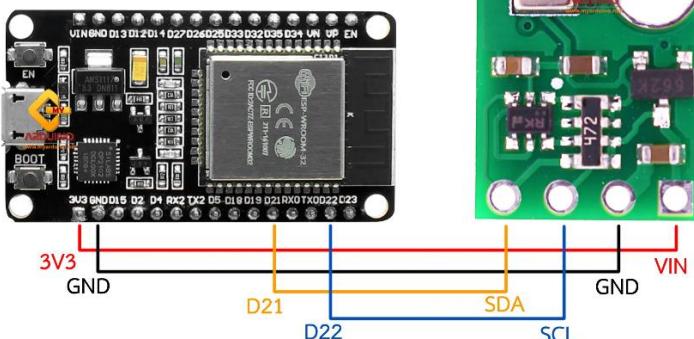
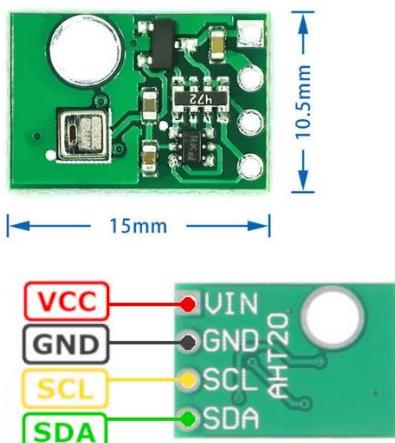
Adafruit_AHTX0 aht;

void setup() {
  Serial.begin(115200);
  Serial.println("Adafruit AHT10/AHT20 demo!");

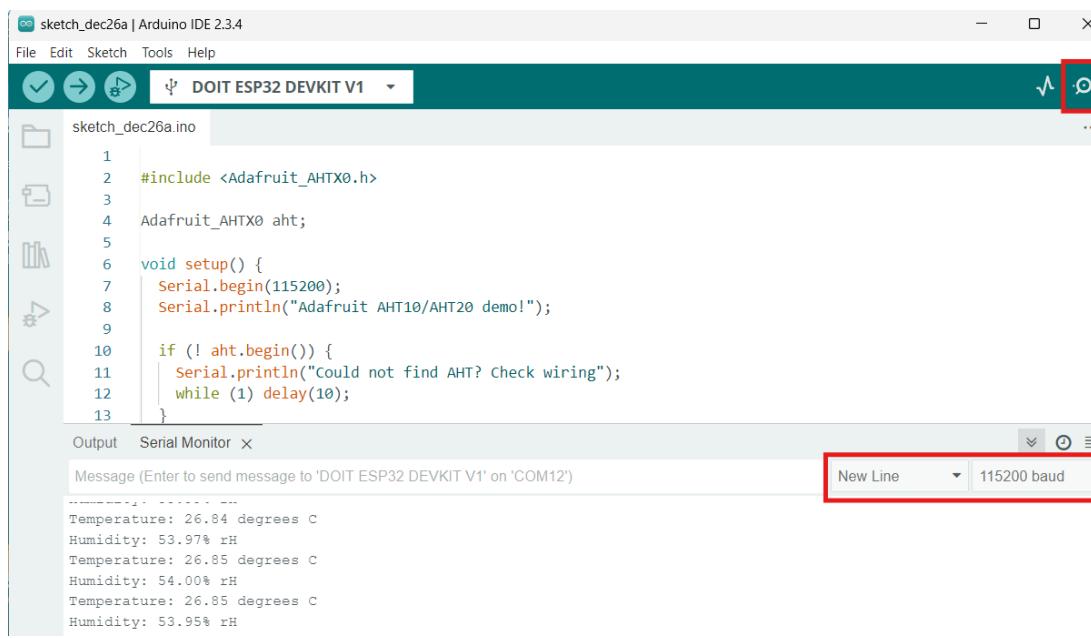
  if (! aht.begin()) {
    Serial.println("Could not find AHT? Check wiring");
    while (1) delay(10);
  }
  Serial.println("AHT10 or AHT20 found");
}

void loop() {
  sensors_event_t humidity, temp;
  aht.getEvent(&humidity, &temp); // populate temp and humidity objects with fresh data
  Serial.print("Temperature: "); Serial.print(temp.temperature); Serial.println(" degrees C");
  Serial.print("Humidity: "); Serial.print(humidity.relative_humidity); Serial.println("% rH");

  delay(500);
}
```



SDA	D21
SCL	D22
GND	GND
VIN	3V3



9. Test Code “Test02 – Display AHT20 on TM1638”

```
#include <Adafruit_AHTX0.h>
#include <TM1638plus.h>
#define pin_STB 5
#define pin_CLK 17
#define pin_DIO 16

Adafruit_AHTX0 aht;
TM1638plus tm(pin_STB, pin_CLK, pin_DIO, true);

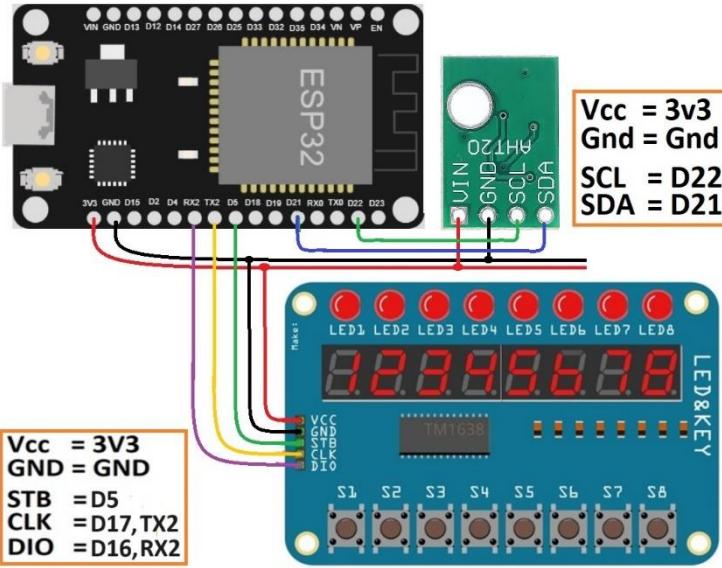
void setup() {
  Serial.begin(115200);
  tm.displayBegin();
  tm.brightness(6);
  Serial.println("Adafruit AHT10/AHT20 demo!");
  if (!aht.begin()) {
    Serial.println("Could not find AHT? Check wiring");
    while (1) delay(10);
  }
  Serial.println("AHT10 or AHT20 found");
}

void loop() {
  delay(2000);
  sensors_event_t humidity, temp;
  aht.getEvent(&humidity, &temp); // populate temp and
  humidity objects with fresh data
  Serial.print("Temperature: ");
  Serial.print(temp.temperature);
  Serial.print(" C, Humidity: ");
  Serial.print(humidity.relative_humidity);
  Serial.print("% rH");

  int intData, singleData;
  float Temp = temp.temperature;
  float Humid = humidity.relative_humidity;
  Serial.print(" >> Temp("C),Humid(%) = ");
  Serial.print(Temp, 1);
  Serial.print(", ");
  Serial.print(Humid, 1);
  Serial.println();

  intData = (int)(Temp * 10);
  tm.displayASCII(0, 't');
  singleData = intData % 10;
  intData /= 10;
  tm.displayHex(3, singleData);
  singleData = intData % 10;
  intData /= 10;
  tm.displayASCIIwDot(2, 0x30 + singleData);
  singleData = intData % 10;
  intData /= 10;
  tm.displayHex(1, singleData);

  intData = (int)(Humid * 10);
  tm.displayASCII(4, 'h');
  singleData = intData % 10;
  intData /= 10;
  tm.displayHex(7, singleData);
  singleData = intData % 10;
  intData /= 10;
  tm.displayASCIIwDot(6, 0x30 + singleData);
  singleData = intData % 10;
  intData /= 10;
  tm.displayHex(5, singleData);
}
```



- How display on MAX7219 7Segment Board?