

**Chili Board – Plus**

## Overview

บอร์ด Chili เป็นบอร์ดที่ถูกพัฒนาจาก บอร์ด Arduino Nano V3.1 โดยมีการเพิ่มฟีเจอร์ต่างๆ ดังนี้

- รองรับการเขียน 2 รูปแบบ ทั้ง Arduino IDE และ Circuit Python
- มีเซ็นเซอร์พื้นฐาน Temperature Humidity Sensor, 3-Axis Accelerometer, 3-Axis Gyroscope, 3-Axis Magnetometer และ Barometer
- ไฟแสดงผล Neo Pixel บนบอร์ด
- หน่วยความจำ SPI Flash ขนาด 8 MB
- มีขนาดเล็กกระทัดรัด เหมาะแก่การประยุกต์ใช้งาน
- ง่ายต่อการ Debug

บอร์ด Chili รองรับการเขียนทั้ง 2 รูปแบบ ซึ่งเป็นภาษาที่ได้รับความนิยม ทำให้สะดวกต่อการใช้งานมากขึ้น บนบอร์ดมีเซ็นเซอร์มาให้ เช่น ใจโร บารอ วัดอุณหภูมิและความชื้น เหมาะสำหรับนำไปทำโดรน หุ่นยนต์ต่างๆ มีขนาดเล็กต่อใช้งานร่วมกับโปรโตบอร์ดได้ เพียงแค่จ่ายไฟผ่านหัว Micro USB ที่หาได้ง่าย ใช้ตัวประมวลผล ATSAM21G18 ARM Cortex M0+ ทำงานด้วยความเร็ว 48 MHz ที่ 3.3V พร้อมหน่วยความจำ 256K FLASH และ 32K RAM ภายในชิปมี USB-to-Serial program & debug ได้โดยตรง ไม่ต้องต่อวงจรเพิ่ม อีกทั้งถูกออกแบบมาให้มีไฟแสดงผล Neo Pixel และหน่วยความจำ SPI Flash บนบอร์ด พร้อมขา GPIO มากถึง 24 ขา และรองรับ I2C เหมาะกับการนำไปต่อฮาร์ดแวร์ในการใช้งาน

## Features

- It is a smallest, complete, and breadboard friendly.
- USB native support, comes with USB bootloader and serial port debugging
- Measures 44mm x 19mm x 7mm without headers soldered in

## Specification

### MCU

- ATSAM21G18 @ 48MHz with 3.3V logic/power
- 256KB of FLASH + 32KB of RAM
- 32.768 KHz crystal for clock generation & RTC

### Input Power

- USB, 5V-PIN
- 3.3V 800mA Regulator
- PTC Fuse on USB Voltage

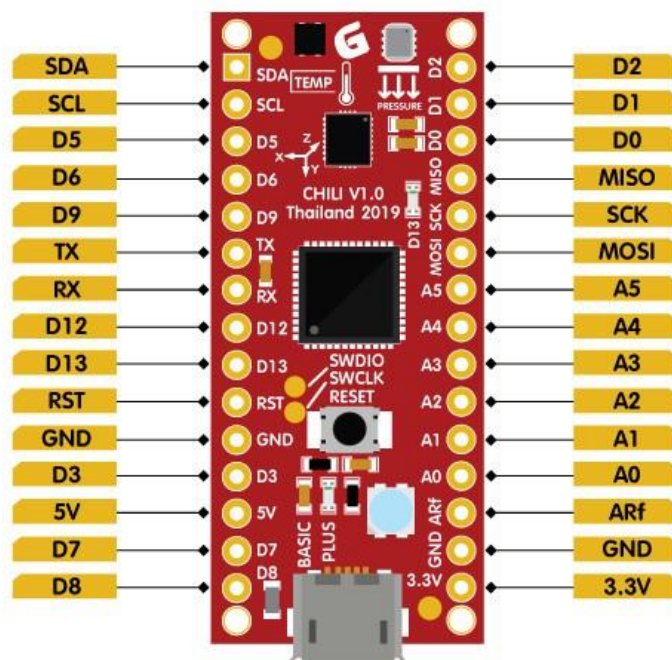
### I/O pins

- 24 GPIO pins (PWM outputs on all pins)
- 6 x 12-bit analog inputs
- 1 x 10-bit analog output (DAC)
- Hardware support Serial, I2C, SPI
- Pin #13 red LED for general purpose blinking
- SPI Flash and NeoPixel
- Reset button

## Sensor

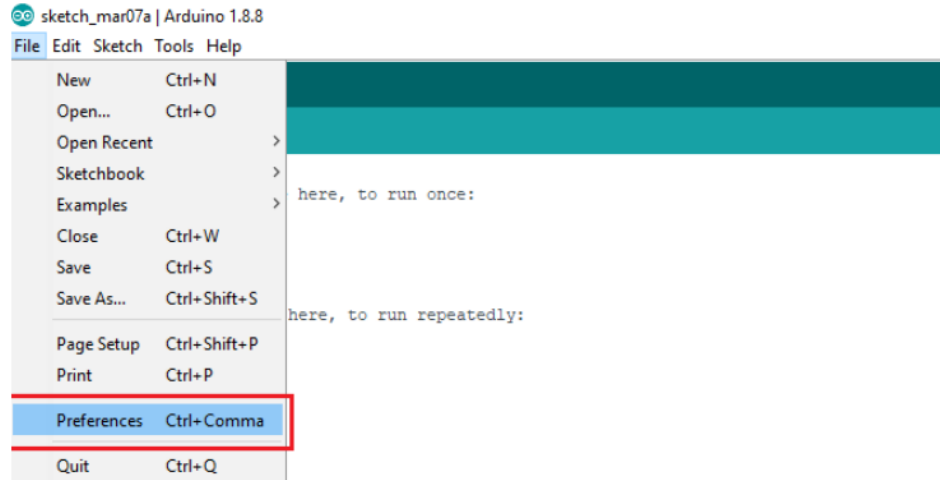
- Temperature and Humidity Sensor
- 3-Axis Accelerometer
- 3-Axis Gyroscope
- 3-Axis Magnetometer
- Barometer

## Pinout



## Arduino IDE setup

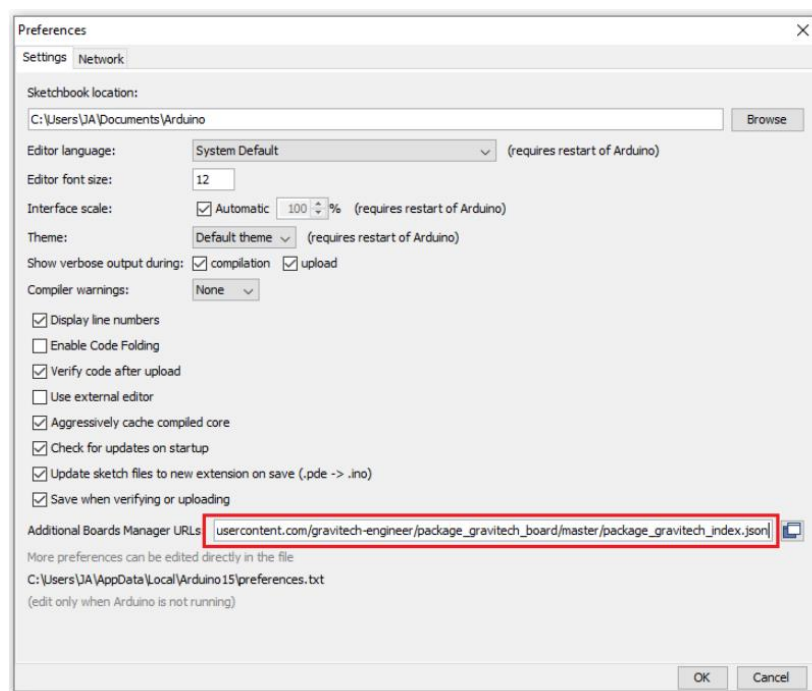
เปิด Preferences โดยไปที่ File > Preferences



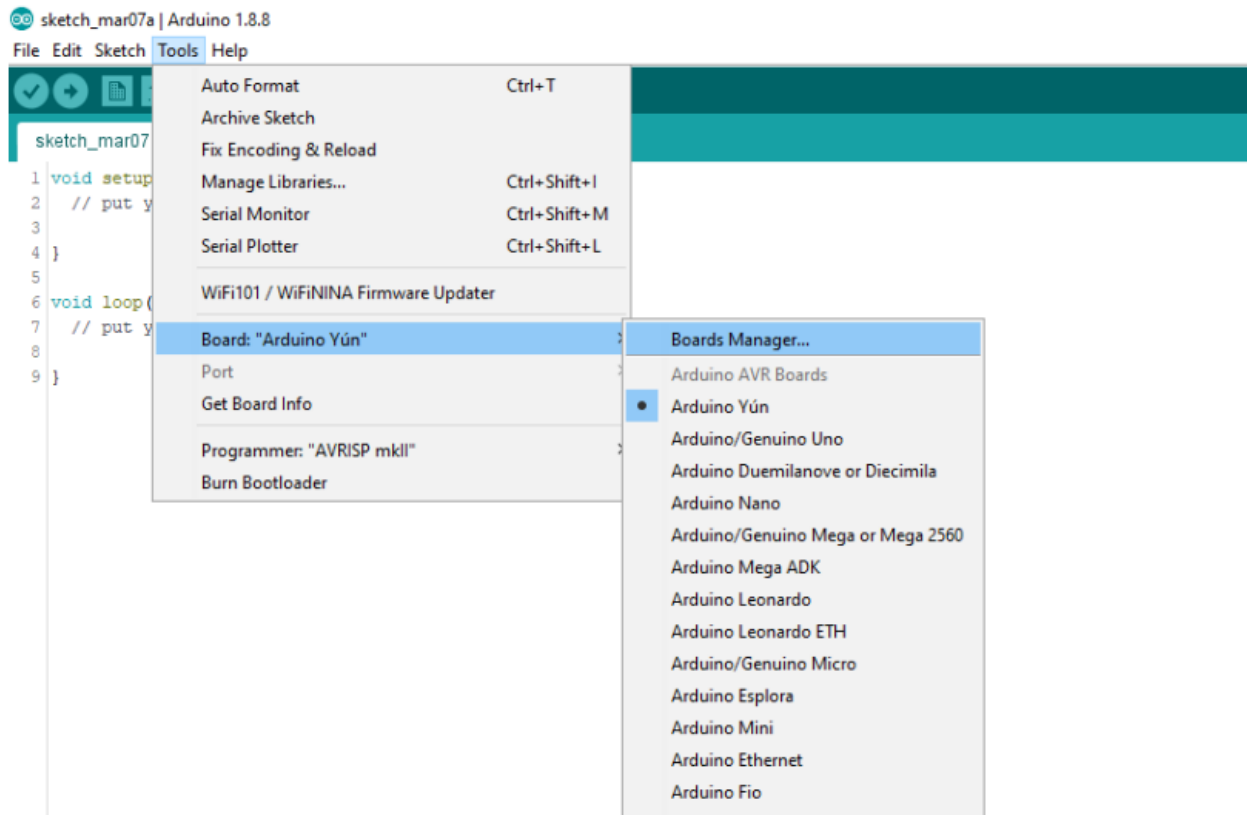
ใส่ URLs ลงไปในช่อง Additional Boards Manager URLs:

[https://raw.githubusercontent.com/gravitechengineer/package\\_gravitech\\_board/master/package\\_gravitech\\_index.json](https://raw.githubusercontent.com/gravitechengineer/package_gravitech_board/master/package_gravitech_index.json)

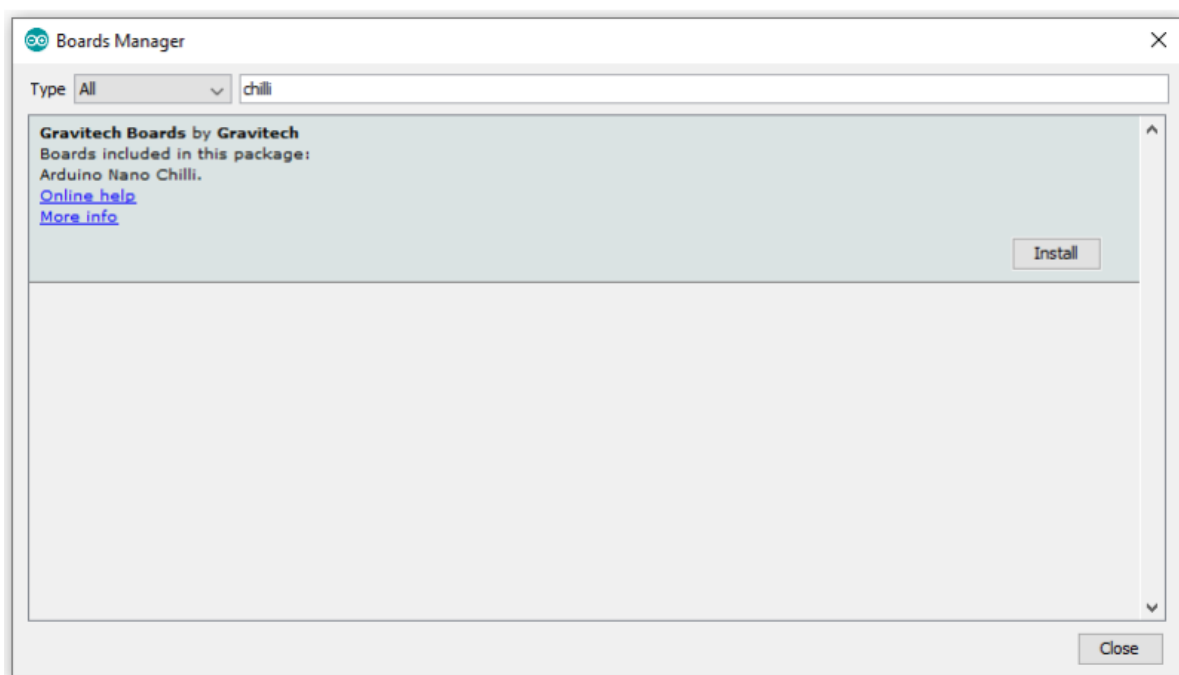
จากนั้นกด OK



เปิด Board Manager โดยเข้าไปที่ Tools > Board: "..." > Boards Manager..

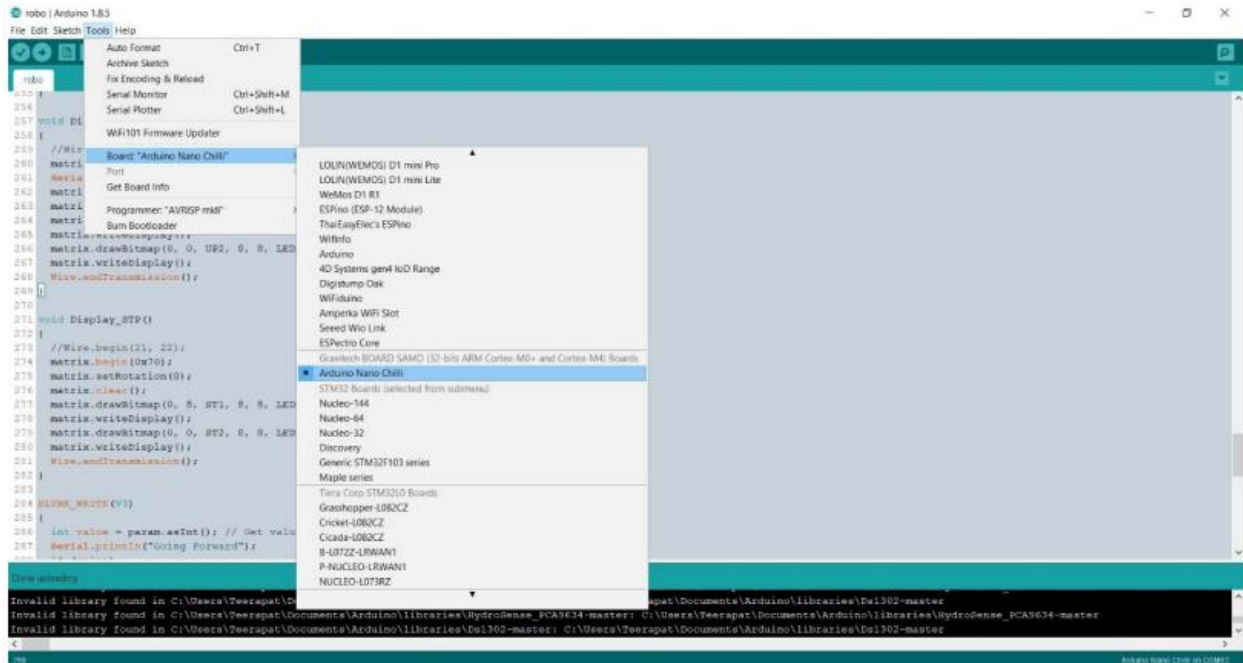


ค้นหาคำว่า chilli > แล้วกด Install



ในการใช้งานบอร์ด ให้เลือกบอร์ดชื่อว่า “ Arduino nano chili ”

โดยเข้าไปที่ Tools > Board: “...” > Arduino nano chili



จากนั้นสามารถใช้งานบอร์ด chili ได้

## Python set up

- เริ่มจากกดที่ลิงค์นี้เพื่อดาวน์โหลด [https://github.com/gravitech-engineer/chili\\_circuitpython](https://github.com/gravitech-engineer/chili_circuitpython)
- เชื่อมต่อ chili Board เข้ากับคอมพิวเตอร์โดยใช้สาย USB
- กดปุ่ม reset 2 ครั้ง เพื่อตรวจสอบไฟ NeoPixel ถ้าขึ้นไฟสีเขียวคือสามารถใช้งานได้ แต่หากขึ้นเป็นสีแดงต้องมีการตรวจเช็คสาย USB ที่ใช้ หากตรวจสอบสาย USB ที่ใช้แล้วยังขึ้นเป็นสีแดงควรมีการเปลี่ยนสาย USB
- เมื่อเข้า This PC จะพบ disk drive ใหม่ที่ชื่อว่า CHILIBOOT
- เข้าไปใน Folder ที่ดาวน์โหลดมาจากลิงค์ข้างต้นชื่อว่า chili circuitpython-master > uf2 แล้ว copy ไฟล์ที่ชื่อว่า chili\_circuitpython.uf2 มาวางไว้ที่ disk drive ที่ชื่อว่า CHILIBOOT



- จากนั้นไฟ LED จะกระพริบ และ CHILIBOOT จะหายไป แต่จะมี disk drive ใหม่ขึ้นมาแทน ชื่อว่า

“CIRCUITPY”

- จากนั้นเข้า Folder Chili\_circuitpython-master แล้ว copy ข้อมูลทั้งหมดใน Folder ลงใน disk drive

CIRCUITPY

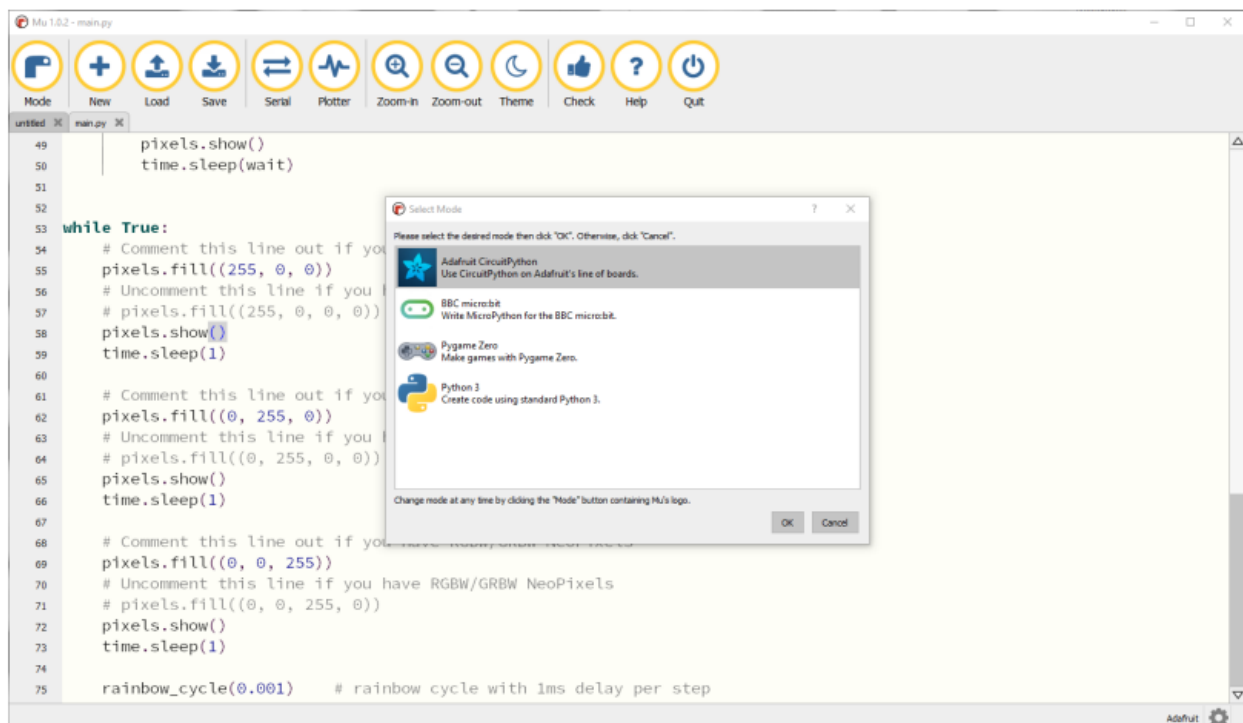
### การติดตั้ง Mu Editor

Mu เป็นโปรแกรมแก้ไข Code อย่างง่าย สามารถทำงานได้ทั้งบน Windows, MacOS, Linux และ Raspberry Pi

ดาวน์โหลดโปรแกรม MU ได้จากลิงค์ Download Mu (<https://adafru.it/BI8>) จากนั้นทำการติดตั้ง

การใช้งาน Mu Editor ร่วมกับ Chili Board

- เปิดโปรแกรม Mu เลือก Adafruit Circuitpython และกด OK

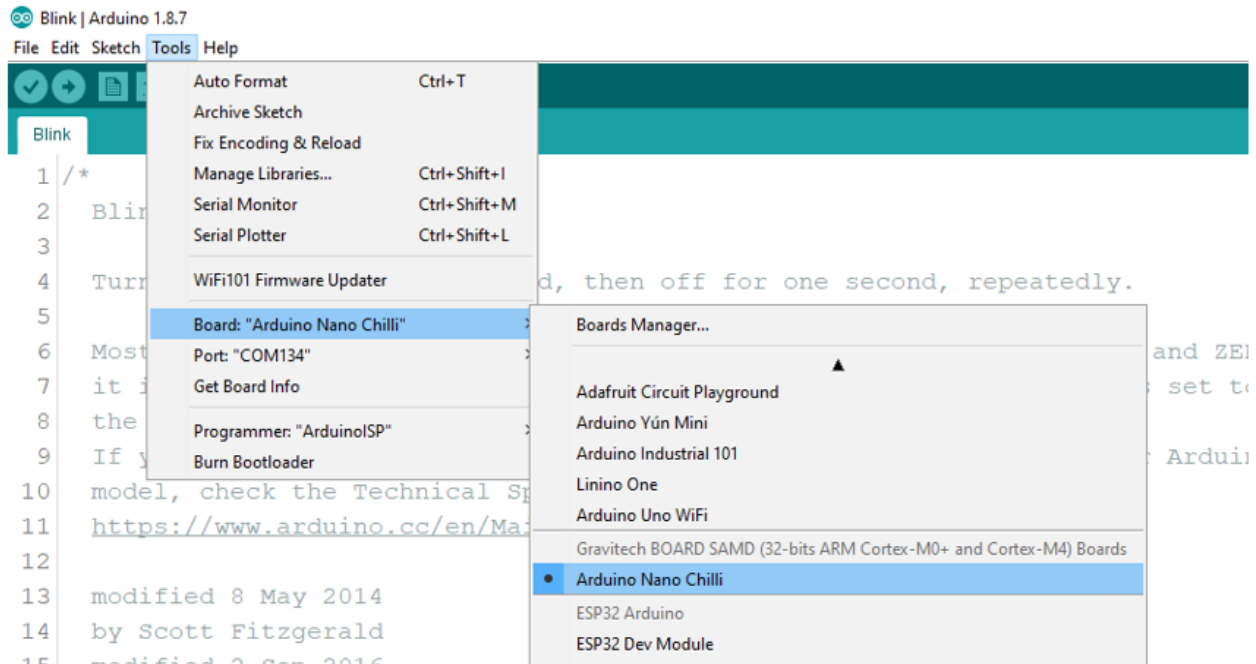


- จากนั้นสามารถแก้ไขโปรแกรมได้ตามต้องการ

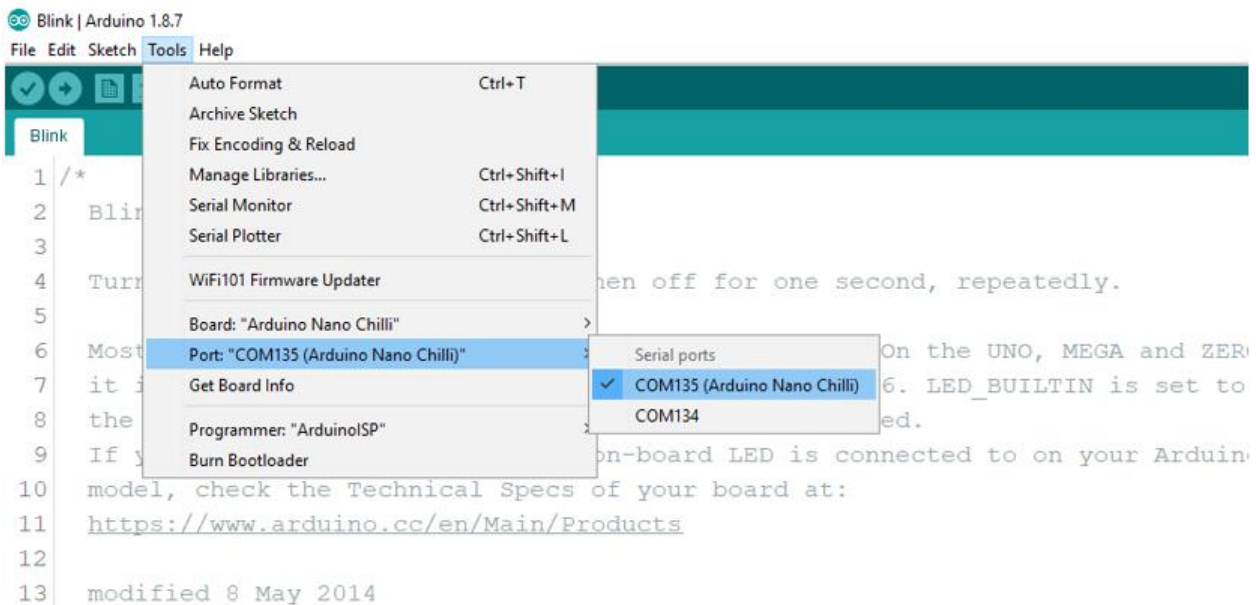


## Arduino IDE: Blink

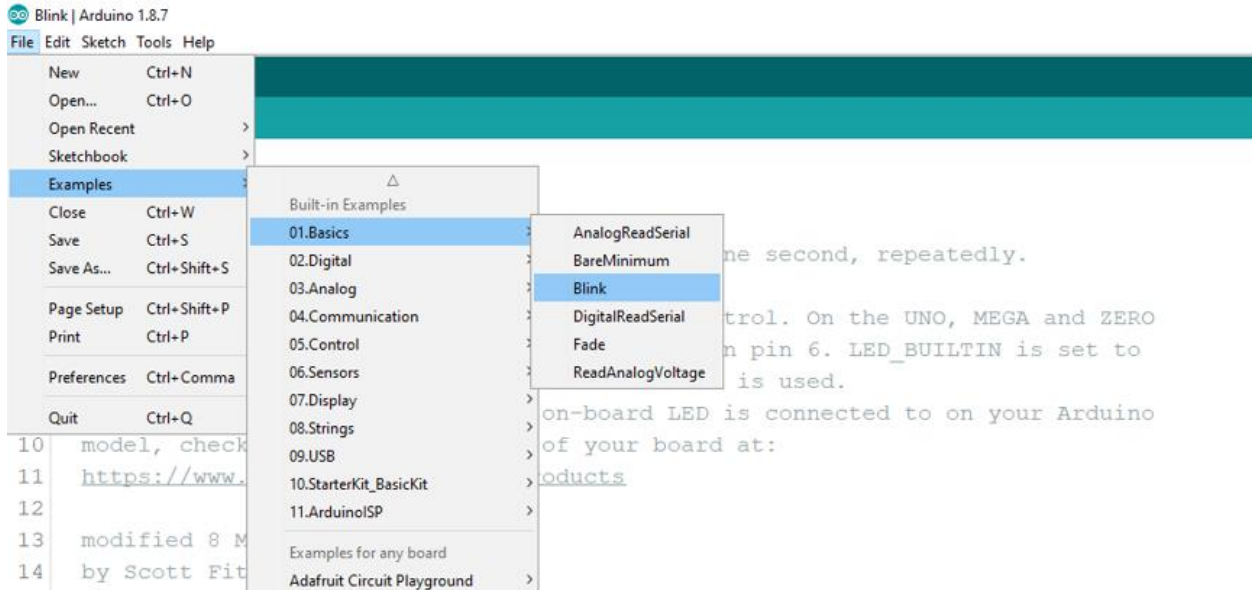
1. กดปุ่ม reset 2 ครั้ง Neopixel บนบอร์ดจะขึ้นสีเขียว
2. เลือกบอร์ด Tools-> Board-> Arduino Nano Chili



3. เลือก Tools-> Port-> COM... (Arduino Nano Chili)



#### 4. เลือก File-> Examples-> 0.1Basics-> Blink



#### Example Blink

// the setup function runs once when you press reset or power the board

```
void setup() {
```

// initialize digital pin 13 as an output.

```
pinMode(13, OUTPUT);
```

```
}
```

// the loop function runs over and over again forever

```
void loop() {
```

```
digitalWrite(13, HIGH);    // turn the LED on (HIGH is the voltage level)
```

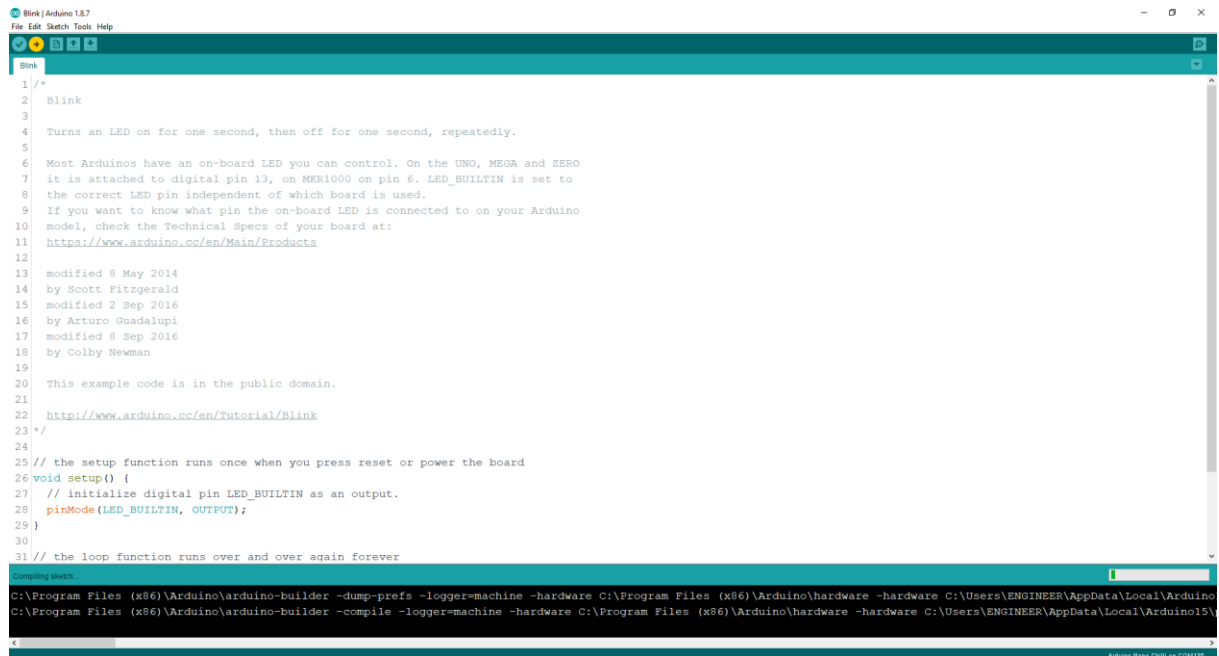
```
delay(1000);               // wait for a second
```

```
digitalWrite(13, LOW);     // turn the LED off by making the voltage LOW
```

```
delay(1000);              // wait for a second
```

```
}
```

## 5. กด upload (การ upload ครั้งแรกต้องเลือก port ใหม่อีกครั้ง) และรอ Done Uploading



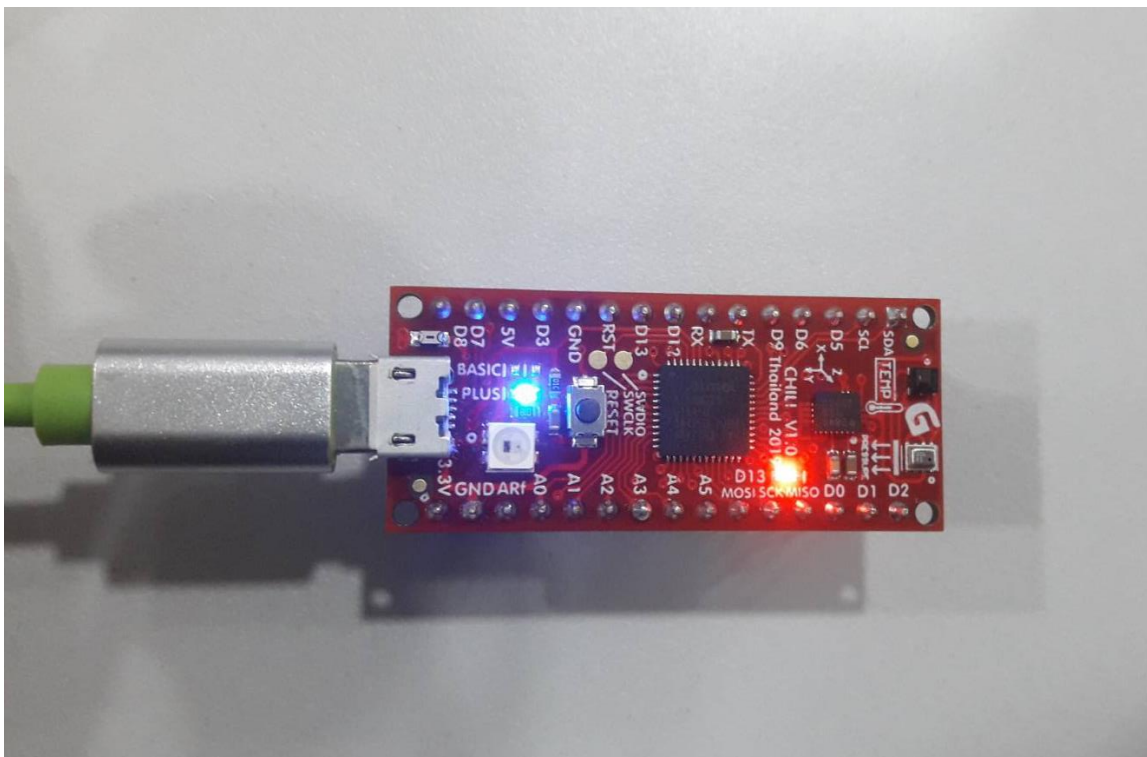
```

1 /*
2  * Blink
3  *
4  * Turns an LED on for one second, then off for one second, repeatedly.
5  *
6  * Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
7  * it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
8  * the correct LED pin independent of which board is used.
9  * If you want to know what pin the on-board LED is connected to on your Arduino
10  * model, check the Technical Specs of your board at:
11  * https://www.arduino.cc/en/Main/Products
12  *
13  * modified 8 May 2014
14  * by Scott Fitzgerald
15  * modified 2 Sep 2016
16  * by Arturo Guadalupi
17  * modified 8 Sep 2016
18  * by Colby Newman
19  *
20  * This example code is in the public domain.
21  *
22  * http://www.arduino.cc/en/Tutorial/Blink
23  */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
  
```

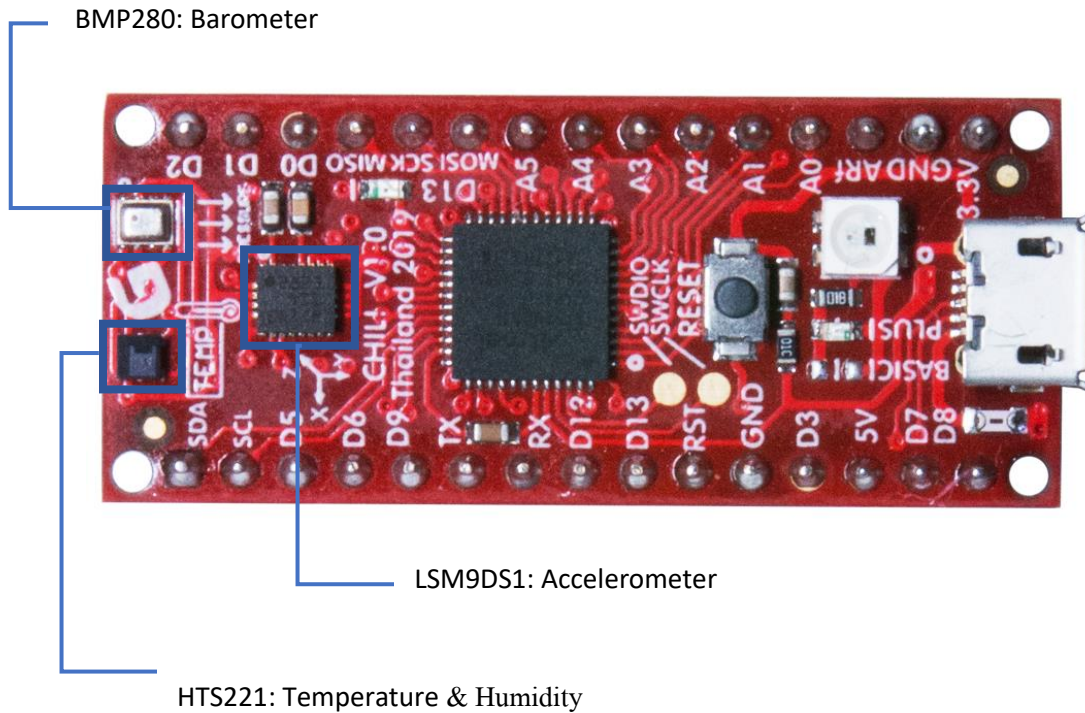
Compiling sketch...

C:\Program Files (x86)\Arduino\arduino-builder -dump-preps -logger-machine -hardware C:\Program Files (x86)\Arduino\hardware -hardware C:\Users\ENGINEER\AppData\Local\Arduino  
 C:\Program Files (x86)\Arduino\arduino-builder -compile -logger-machine -hardware C:\Program Files (x86)\Arduino\hardware -hardware C:\Users\ENGINEER\AppData\Local\Arduino15\

## 6. LED D13 จะกะพริบ



## วิธีการใช้งานเซนเซอร์บนบอร์ด CHILI



บนบอร์ดมีเซนเซอร์ 3 ตัว ได้แก่ Temperature Humidity Sensor, 3-Axis Accelerometer, 3-Axis Gyroscope, 3-Axis Magnetometer และ Barometer โดยมีวิธีการใช้งานดังนี้

### 1.Barometer Sensor: BMP280

สามารถวัดค่าอุณหภูมิ ความดัน และ ความสูงเหนือระดับน้ำทะเล (altitude)

#### Code

```
#include <Wire.h>
#include <SPI.h>
#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 test"));
  if (!bmp.begin()) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring!"));
    while (1);
  }

  /* Default settings from datasheet. */
  bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* 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() {
  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);
}
```

bmp280test | Arduino 1.8.7

File Edit Sketch Tools Help

```
16 *****/
17
18 #include <Wire.h>
19 #include <SPI.h>
20 #include <Adafruit_BMP280.h>
21
22 #define BMP_SCK (13)
23 #define BMP_MISO (12)
24 #define BMP_MOSI (11)
25 #define BMP_CS (10)
26
27 Adafruit_BMP280 bmp; // I2C
28 //Adafruit_BMP280 bmp(BMP_CS); // hardware SPI
29 //Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO, BMP_SCK);
30
31 void setup() {
32   Serial.begin(9600);
33   Serial.println(F("BMP280 test"));
34
35   if (!bmp.begin()) {
36     Serial.println(F("Could not find a valid BMP280 sensor, check wiring!"));
37     while (1);
38   }
39 }
```

COM127 (Arduino Nano Chilli)

Send

```
Temperature = 25.08 *C
Pressure = 101000.01 Pa
Approx altitude = 27.09 m

Temperature = 25.08 *C
Pressure = 100999.59 Pa
Approx altitude = 27.13 m

Temperature = 25.09 *C
Pressure = 101004.78 Pa
Approx altitude = 26.69 m

Temperature = 25.08 *C
Pressure = 100997.86 Pa
Approx altitude = 27.27 m
```

☐ Autoscroll ☐ Show timestamp

No line ending 9600 baud Clear output



## 2. Temperature & Humidity: HTS221

สามารถวัดค่าอุณหภูมิ และความชื้น

### Code

```
#include <Wire.h>

#include <Arduino.h>

#include <HTS221.h>

// the setup function runs once when you press reset or power the board

void setup() {

  //Initiate the Wire library and join the I2C bus

  Wire.begin();

  pinMode(PIN_LED_13, OUTPUT);

  smeHumidity.begin();

  Serial.begin(115200);

}

// the loop function runs over and over again forever

void loop() {

  double data = 0;

  data = smeHumidity.readHumidity();

  Serial.print("Humidity  : ");

  Serial.print(data);

  Serial.println(" %");


  data = smeHumidity.readTemperature();

  Serial.print("Temperature: ");

  Serial.print(data);

  Serial.println(" celsius");


  digitalWrite(PIN_LED_13, LOW);

  delay(300);

  digitalWrite(PIN_LED_13, HIGH);    // turn the LED on

  delay(700);           // wait for a second

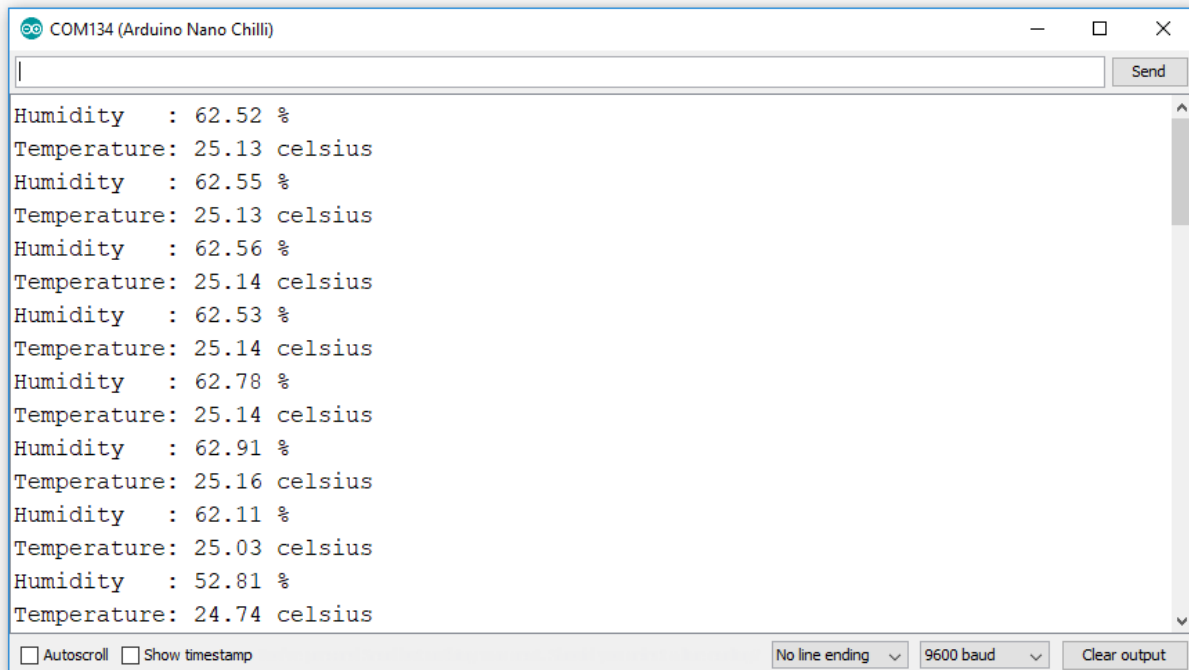
}
```



HumidityTemperature | Arduino 1.8.7

File Edit Sketch Tools Help

```
HumidityTemperature $
1 #include <Wire.h>
2 #include <Arduino.h>
3
4 #include <HTS221.h>
5
6 // the setup function runs once when you press reset or power the board
7 void setup() {
8   //Initiate the Wire library and join the I2C bus
9   Wire.begin();
10  pinMode(PIN_LED_13, OUTPUT);
11
12  smeHumidity.begin();
13  Serial.begin(115200);
14 }
15
16 // the loop function runs over and over again forever
17 void loop() {
18
19   double data = 0;
20
21   data = smeHumidity.readHumidity();
22   Serial.print("Humidity  : ");
23   Serial.print(data);
24   Serial.println(" %");
25
26   data = smeHumidity.readTemperature();
27   Serial.print("Temperature: ");
28   Serial.print(data);
29   Serial.println(" celsius");
```



COM134 (Arduino Nano Chilli)

Humidity : 62.52 %  
Temperature: 25.13 celsius  
Humidity : 62.55 %  
Temperature: 25.13 celsius  
Humidity : 62.56 %  
Temperature: 25.14 celsius  
Humidity : 62.53 %  
Temperature: 25.14 celsius  
Humidity : 62.78 %  
Temperature: 25.14 celsius  
Humidity : 62.91 %  
Temperature: 25.16 celsius  
Humidity : 62.11 %  
Temperature: 25.03 celsius  
Humidity : 52.81 %  
Temperature: 24.74 celsius

☐ Autoscroll ☐ Show timestamp No line ending 9600 baud Clear output

### 3. Accelerometer: LSM9DS1

สามารถวัดค่า 3-Axis Accelerometer, 3-Axis Gyroscope และ 3-Axis Magnetometer

#### Code

```
#include <Wire.h>

#include <SPI.h>

#include <Adafruit_LSM9DS1.h>

#include <Adafruit_Sensor.h> // not used in this demo but required!


// i2c
Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1();


#define LSM9DS1_SCK A5
#define LSM9DS1_MISO 12
#define LSM9DS1_MOSI A4
#define LSM9DS1_XGCS 6
#define LSM9DS1_MCS 5

// You can also use software SPI
//Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1(LSM9DS1_SCK, LSM9DS1_MISO, LSM9DS1_MOSI,
LSM9DS1_XGCS, LSM9DS1_MCS);

// Or hardware SPI! In this case, only CS pins are passed in
//Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1(LSM9DS1_XGCS, LSM9DS1_MCS);


void setupSensor()
{
    // 1.) Set the accelerometer range
    lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_2G);
    //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_4G);
    //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_8G);
    //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_16G);


    // 2.) Set the magnetometer sensitivity
    lsm.setupMag(lsm.LSM9DS1_MAGGAIN_4GAUSS);
```

```
//lsm.setupMag(lsm.LSM9DS1_MAGGAIN_8GAUSS);
//lsm.setupMag(lsm.LSM9DS1_MAGGAIN_12GAUSS);
//lsm.setupMag(lsm.LSM9DS1_MAGGAIN_16GAUSS);
// 3.) Setup the gyroscope
lsm.setupGyro(lsm.LSM9DS1_GYROSCALE_245DPS);
//lsm.setupGyro(lsm.LSM9DS1_GYROSCALE_500DPS);
//lsm.setupGyro(lsm.LSM9DS1_GYROSCALE_2000DPS);
}

void setup()
{
  Serial.begin(115200);
  while (!Serial) {
    delay(1); // will pause Zero, Leonardo, etc until serial console opens
  }
  Serial.println("LSM9DS1 data read demo");
  // Try to initialise and warn if we couldn't detect the chip
  if (!lsm.begin())
  {
    Serial.println("Oops ... unable to initialize the LSM9DS1. Check your wiring!");
    while (1);
  }
  Serial.println("Found LSM9DS1 9DOF");

  // helper to just set the default scaling we want, see above!
  setupSensor();
}

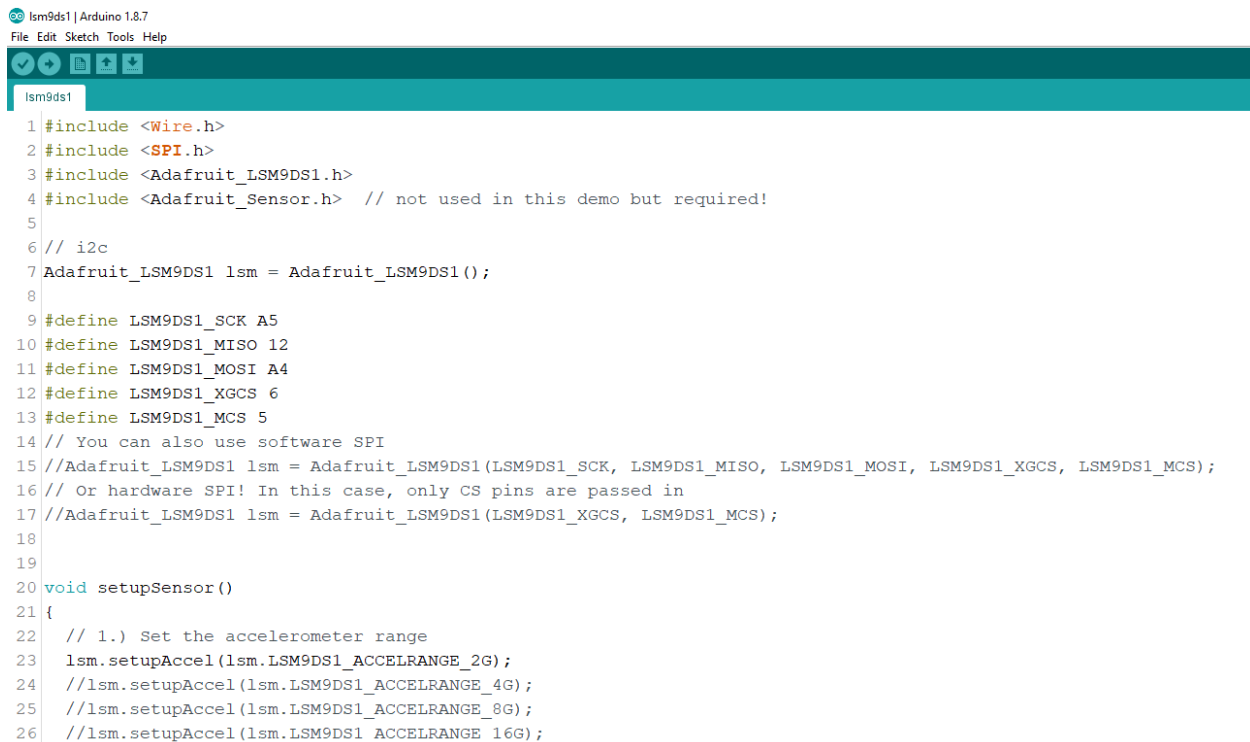
void loop()
{
  lsm.read(); /* ask it to read in the data */
  /* Get a new sensor event */
  sensors_event_t a, m, g, temp;
  lsm.getEvent(&a, &m, &g, &temp);
  Serial.print("Accel X: "); Serial.print(a.acceleration.x); Serial.print(" m/s^2");
```

```
Serial.print("\tY: "); Serial.print(a.acceleration.y);   Serial.print(" m/s^2 ");
Serial.print("\tZ: "); Serial.print(a.acceleration.z);   Serial.println(" m/s^2 ");

Serial.print("Mag X: "); Serial.print(m.magnetic.x);   Serial.print(" gauss");
Serial.print("\tY: "); Serial.print(m.magnetic.y);   Serial.print(" gauss");
Serial.print("\tZ: "); Serial.print(m.magnetic.z);   Serial.println(" gauss");

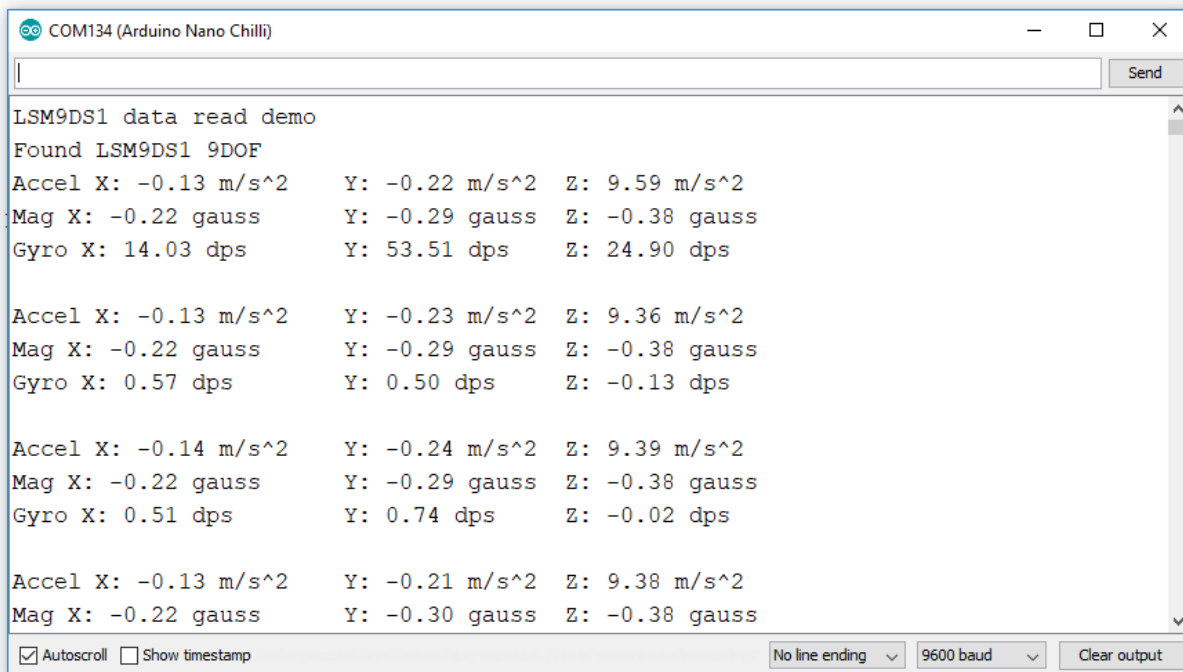
Serial.print("Gyro X: "); Serial.print(g.gyro.x);   Serial.print(" dps");
Serial.print("\tY: "); Serial.print(g.gyro.y);   Serial.print(" dps");
Serial.print("\tZ: "); Serial.print(g.gyro.z);   Serial.println(" dps");

Serial.println();
delay(200);
}
```



```
lsm9ds1 | Arduino 1.8.7
File Edit Sketch Tools Help

lsm9ds1
1 #include <Wire.h>
2 #include <SPI.h>
3 #include <Adafruit_LSM9DS1.h>
4 #include <Adafruit_Sensor.h> // not used in this demo but required!
5
6 // i2c
7 Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1();
8
9 #define LSM9DS1_SCK A5
10 #define LSM9DS1_MISO 12
11 #define LSM9DS1_MOSI A4
12 #define LSM9DS1_XGCS 6
13 #define LSM9DS1_MCS 5
14 // You can also use software SPI
15 //Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1(LSM9DS1_SCK, LSM9DS1_MISO, LSM9DS1_MOSI, LSM9DS1_XGCS, LSM9DS1_MCS);
16 // Or hardware SPI! In this case, only CS pins are passed in
17 //Adafruit_LSM9DS1 lsm = Adafruit_LSM9DS1(LSM9DS1_XGCS, LSM9DS1_MCS);
18
19
20 void setupSensor()
21 {
22   // 1.) Set the accelerometer range
23   lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_2G);
24   //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_4G);
25   //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_8G);
26   //lsm.setupAccel(lsm.LSM9DS1_ACCEL_RANGE_16G);
```



COM134 (Arduino Nano Chilli)

LSM9DS1 data read demo  
Found LSM9DS1 9DOF

Accel X: -0.13 m/s^2	Y: -0.22 m/s^2	Z: 9.59 m/s^2
Mag X: -0.22 gauss	Y: -0.29 gauss	Z: -0.38 gauss
Gyro X: 14.03 dps	Y: 53.51 dps	Z: 24.90 dps
Accel X: -0.13 m/s^2	Y: -0.23 m/s^2	Z: 9.36 m/s^2
Mag X: -0.22 gauss	Y: -0.29 gauss	Z: -0.38 gauss
Gyro X: 0.57 dps	Y: 0.50 dps	Z: -0.13 dps
Accel X: -0.14 m/s^2	Y: -0.24 m/s^2	Z: 9.39 m/s^2
Mag X: -0.22 gauss	Y: -0.29 gauss	Z: -0.38 gauss
Gyro X: 0.51 dps	Y: 0.74 dps	Z: -0.02 dps
Accel X: -0.13 m/s^2	Y: -0.21 m/s^2	Z: 9.38 m/s^2
Mag X: -0.22 gauss	Y: -0.30 gauss	Z: -0.38 gauss

☒ Autoscroll ☐ Show timestamp

No line ending 9600 baud Clear output