Downloads

Arduino IDE 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the **Getting Started** page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is **hosted by GitHub**. See the instructions for **building the code**. Latest release source code archives are available **here**. The archives are PGP-signed so they can be verified using **this** gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

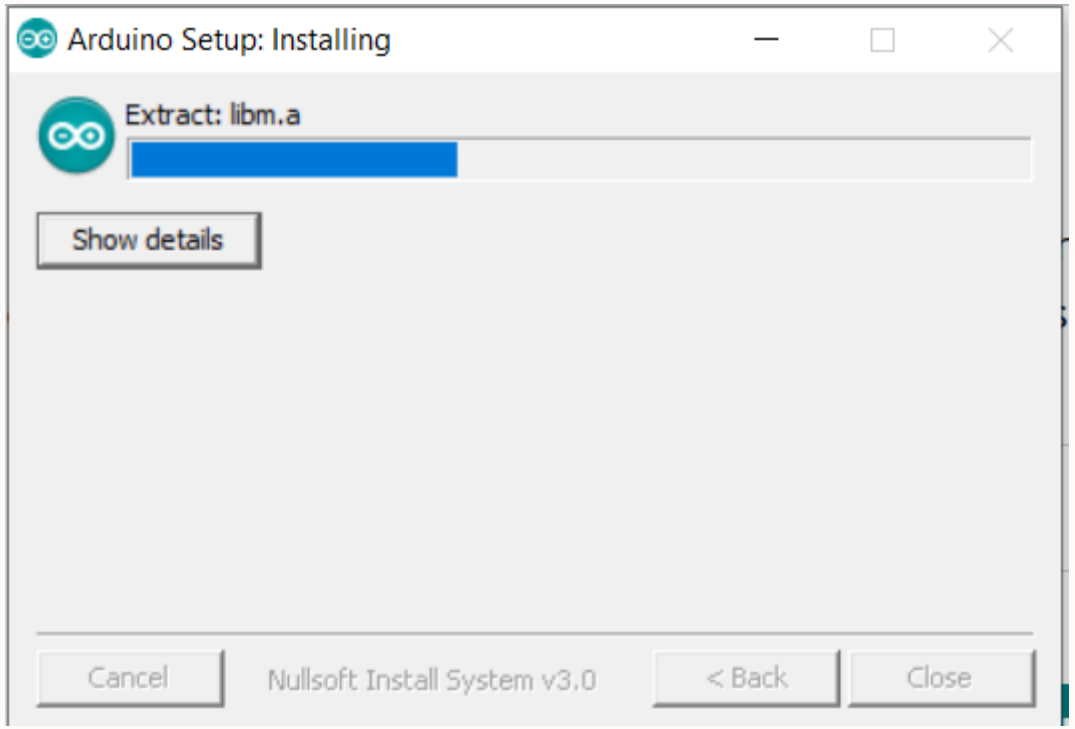
Windows app Win 8.1 or 10 Get

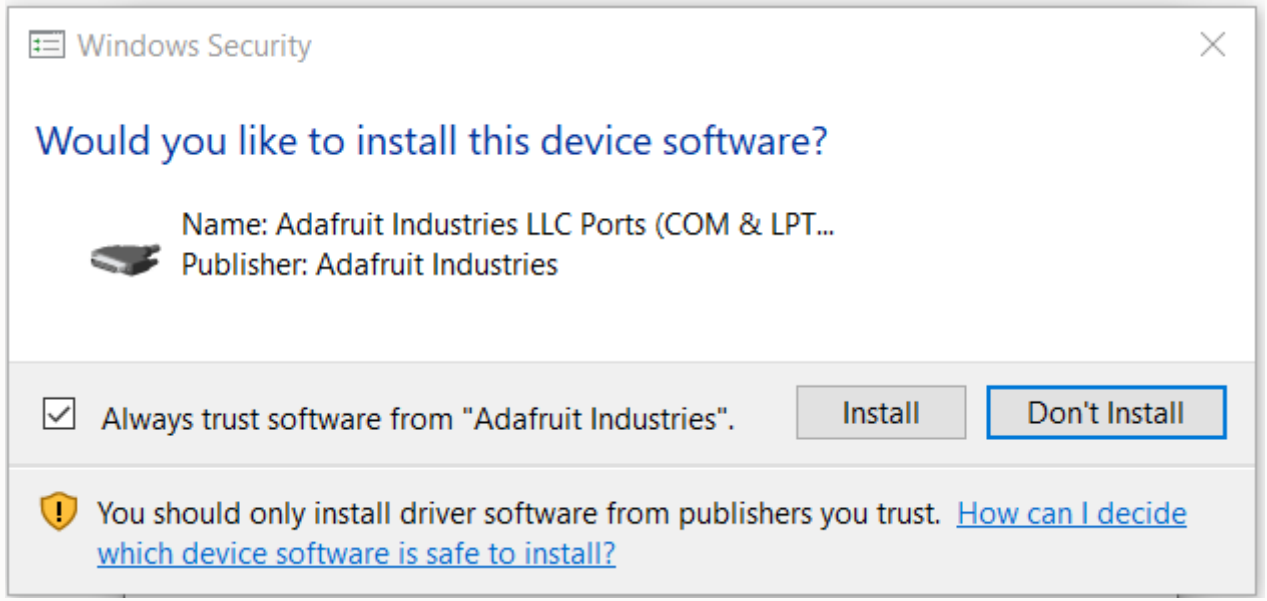
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

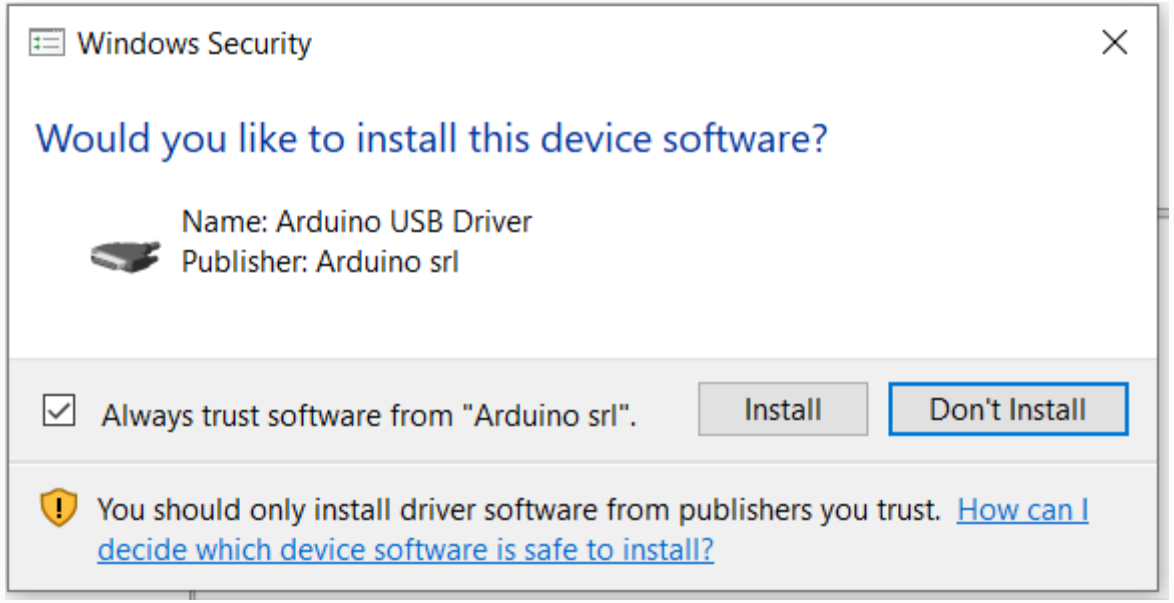
Mac OS X 10.10 or newer

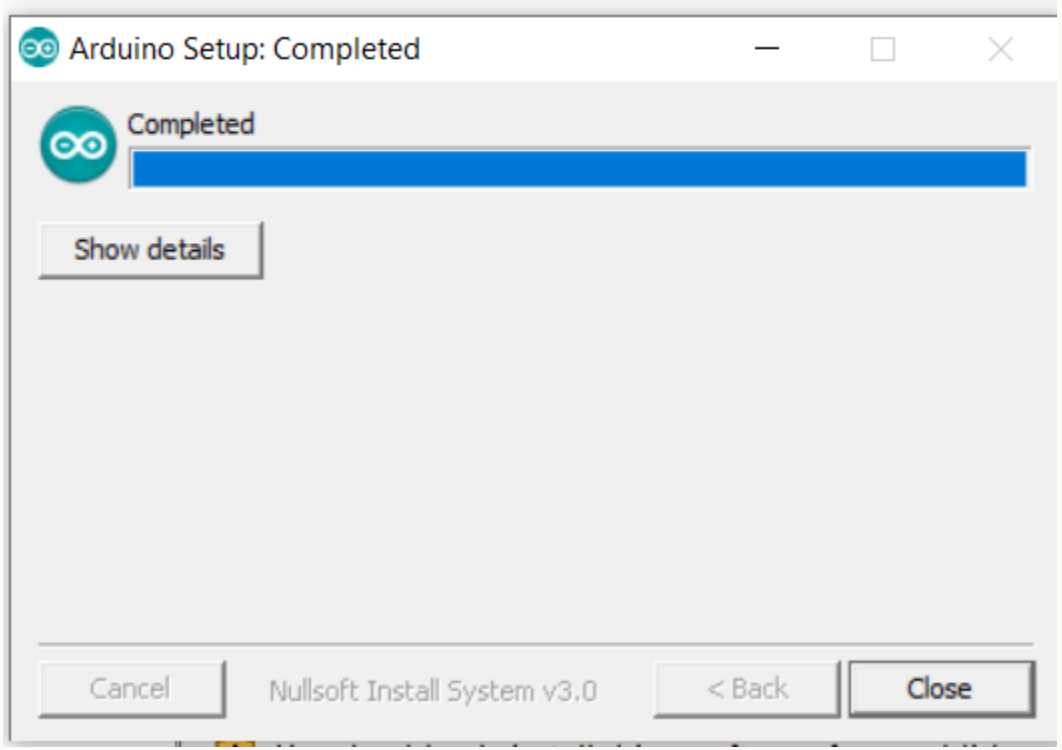
Release Notes Checksums (sha512)

Help

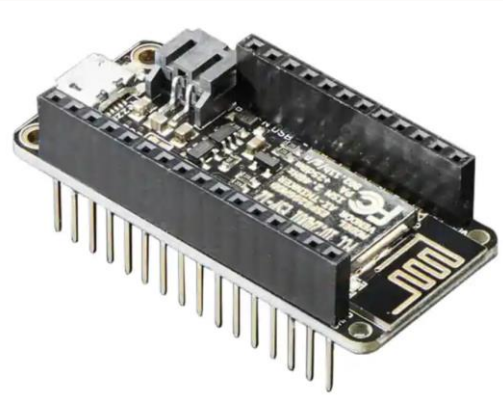

Arduino Setup: Installing
Extract: libm.a
Show details
Cancel
Nullsoft Install System v3.0
< Back
Close


Windows Security
Would you like to install this device software?
Name: Adafruit Industries LLC Ports (COM & LPT...
Publisher: Adafruit Industries
Always trust software from "Adafruit Industries".
Install
Don't Install
You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

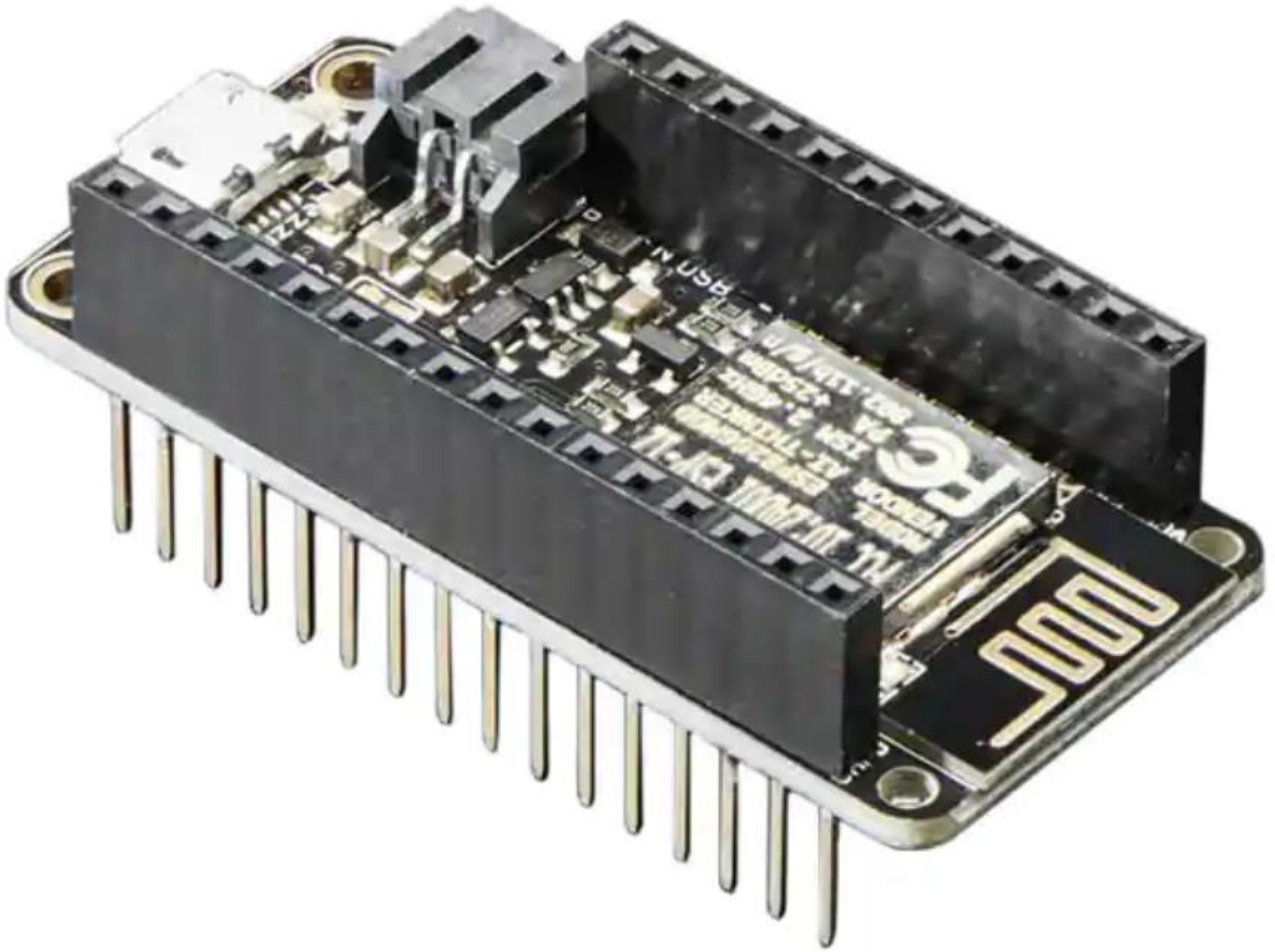

Windows Security
Would you like to install this device software?
Name: Arduino USB Driver
Publisher: Arduino srl
Always trust software from "Arduino srl".
Install
Don't Install
You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

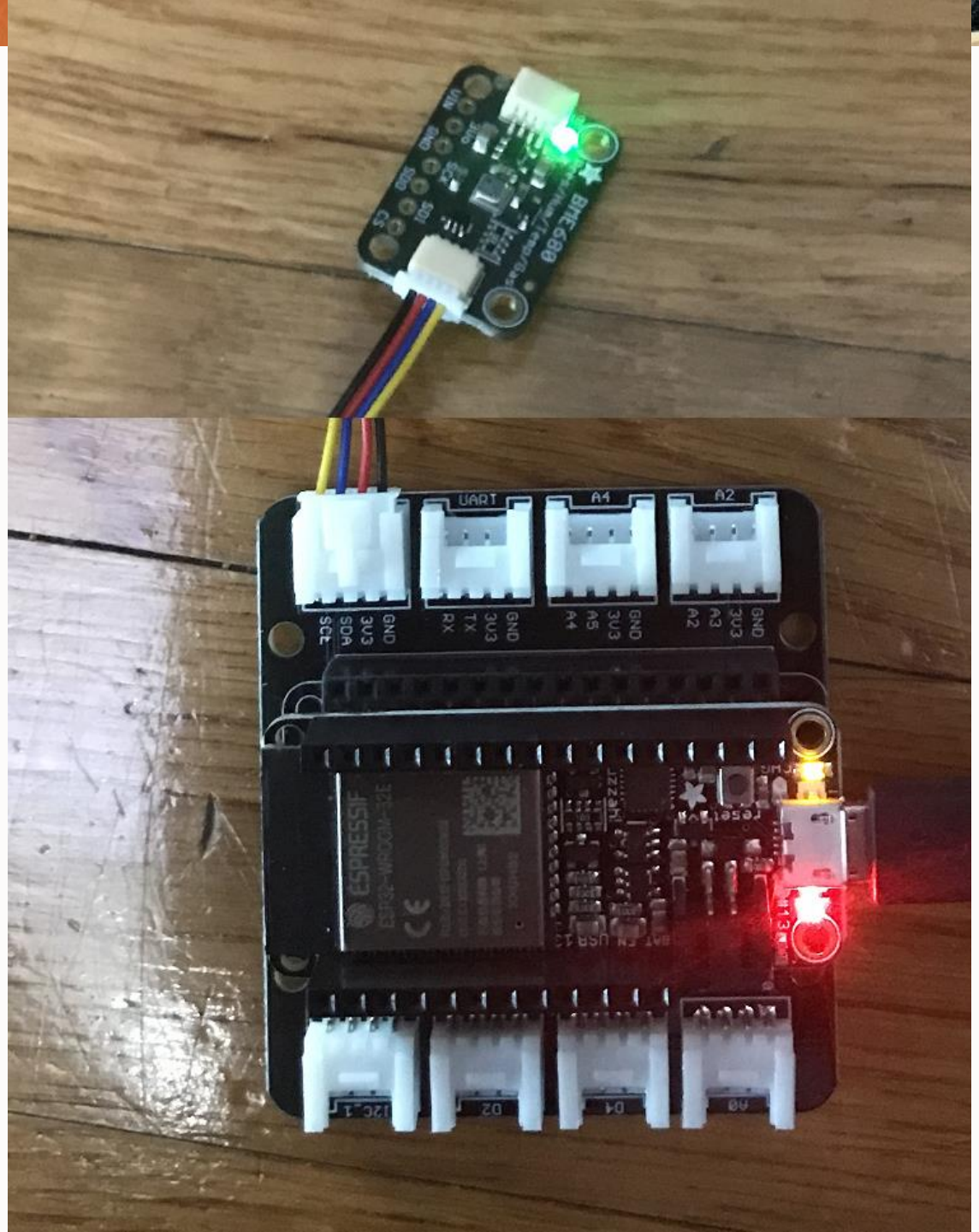

Arduino Setup: Completed
Completed
Show details
Cancel
Nullsoft Install System v3.0
< Back
Close

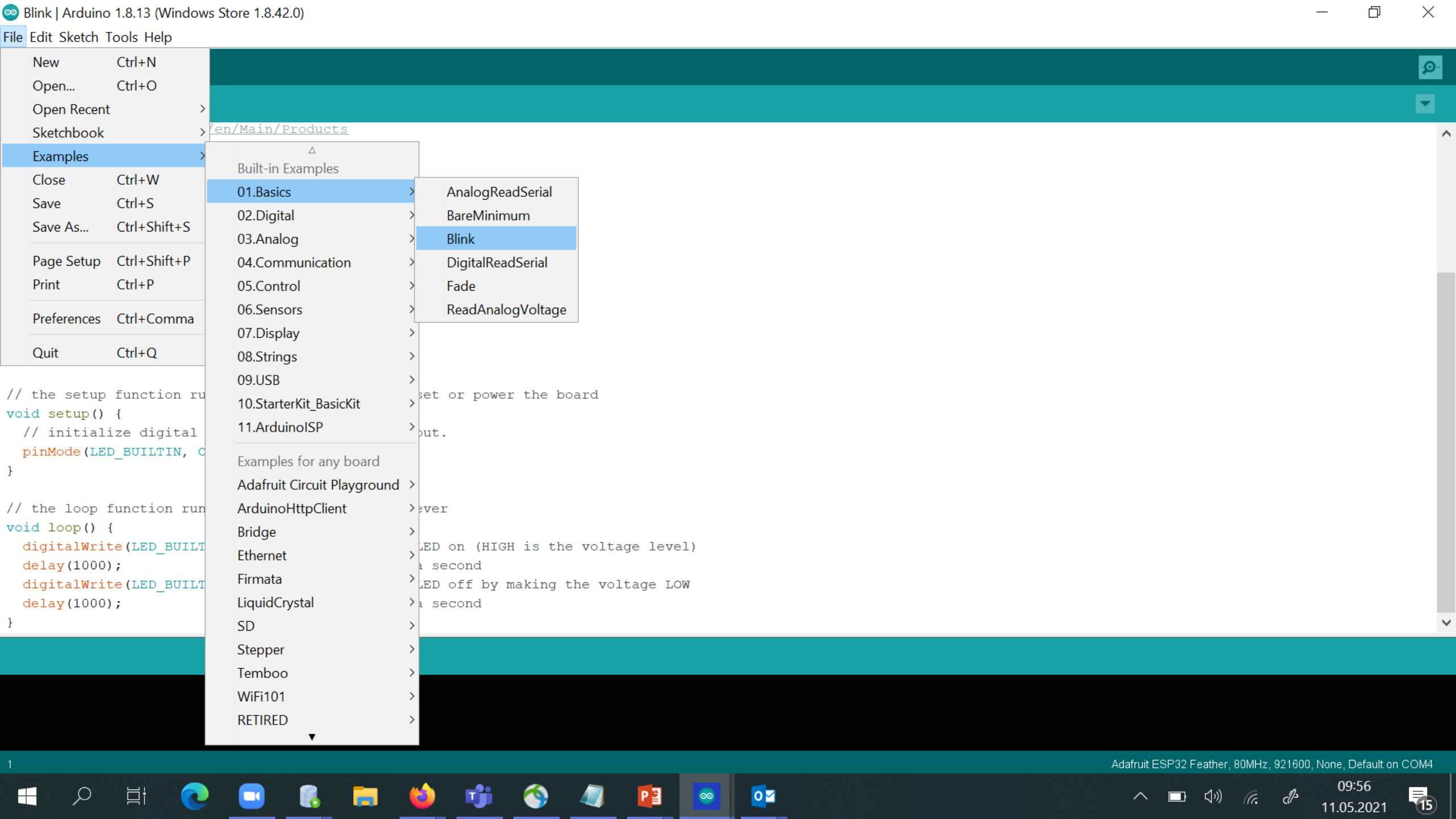
- https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
  https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json



- https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
  https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json




BME680
UART
A4
A2
GND
3V3
SDA
SCL
TX
RX
A5
A3
A1
A0
D2
D4
I2C_1
ESPRESSIF


Blink | Arduino 1.8.13 (Windows Store 1.8.42.0)
File Edit Sketch Tools Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q
Built-in Examples
01.Basics
02.Digital
03.Analog
04.Communication
05.Control
06.Sensors
07.Display
08.Strings
09.USB
10.StarterKit_BasicKit
11.ArduinoISP
Examples for any board
Adafruit Circuit Playground
ArduinoHttpClient
Bridge
Ethernet
Firmata
LiquidCrystal
SD
Stepper
Temboo
WiFi101
RETIRED
AnalogReadSerial
BareMinimum
Blink
DigitalReadSerial
Fade
ReadAnalogVoltage
/en/Main/Products
// the setup function ru
void setup() {
// initialize digital
pinMode(LED_BUILTIN, O
}
// the loop function run
void loop() {
digitalWrite(LED_BUILT
delay(1000);
digitalWrite(LED_BUILT
delay(1000);
}
set or power the board
ever
LED on (HIGH is the voltage level)
second
LED off by making the voltage LOW
second
Adafruit ESP32 Feather, 80MHz, 921600, None, Default on COM4
09:56
11.05.2021

Blink | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

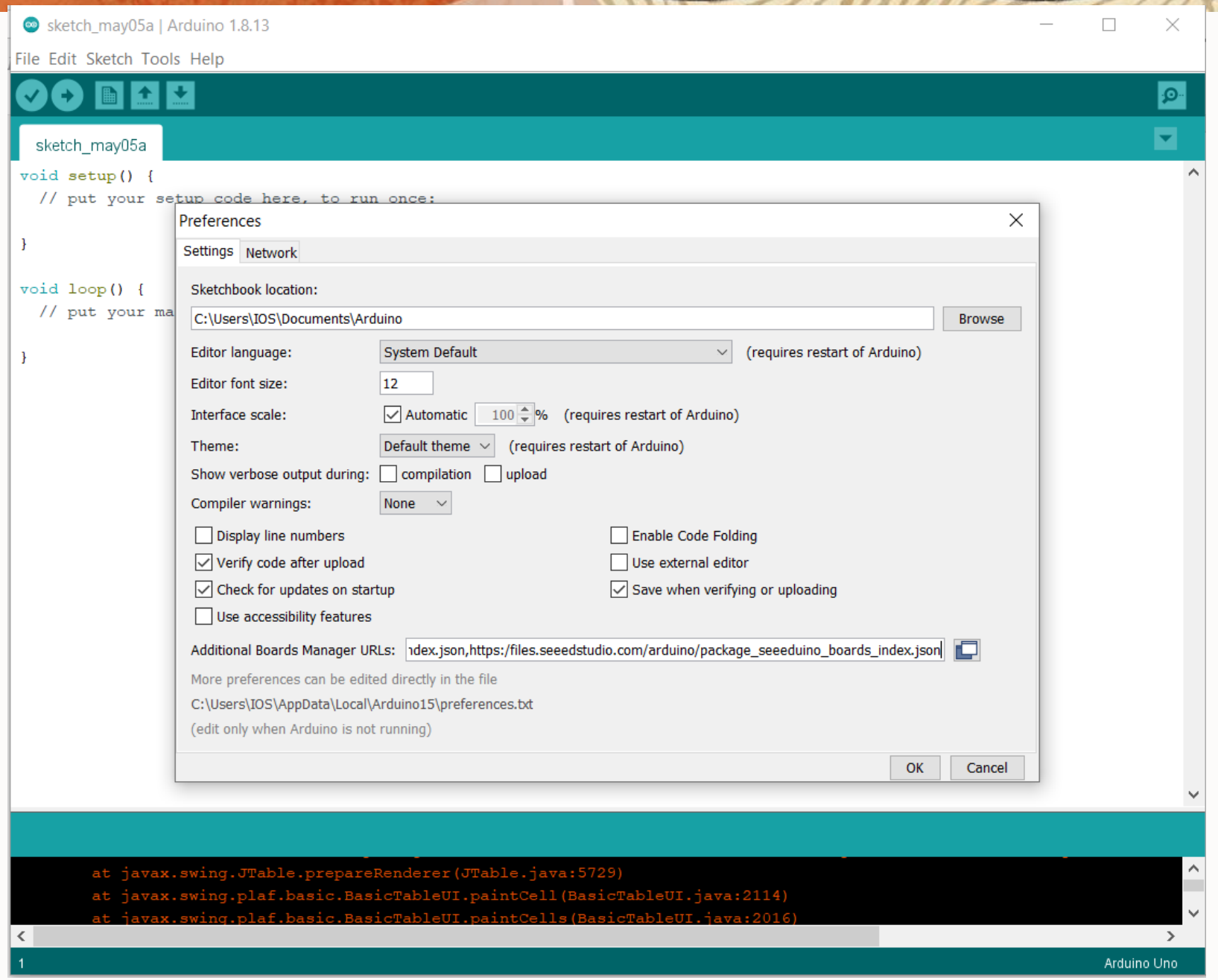
Blink

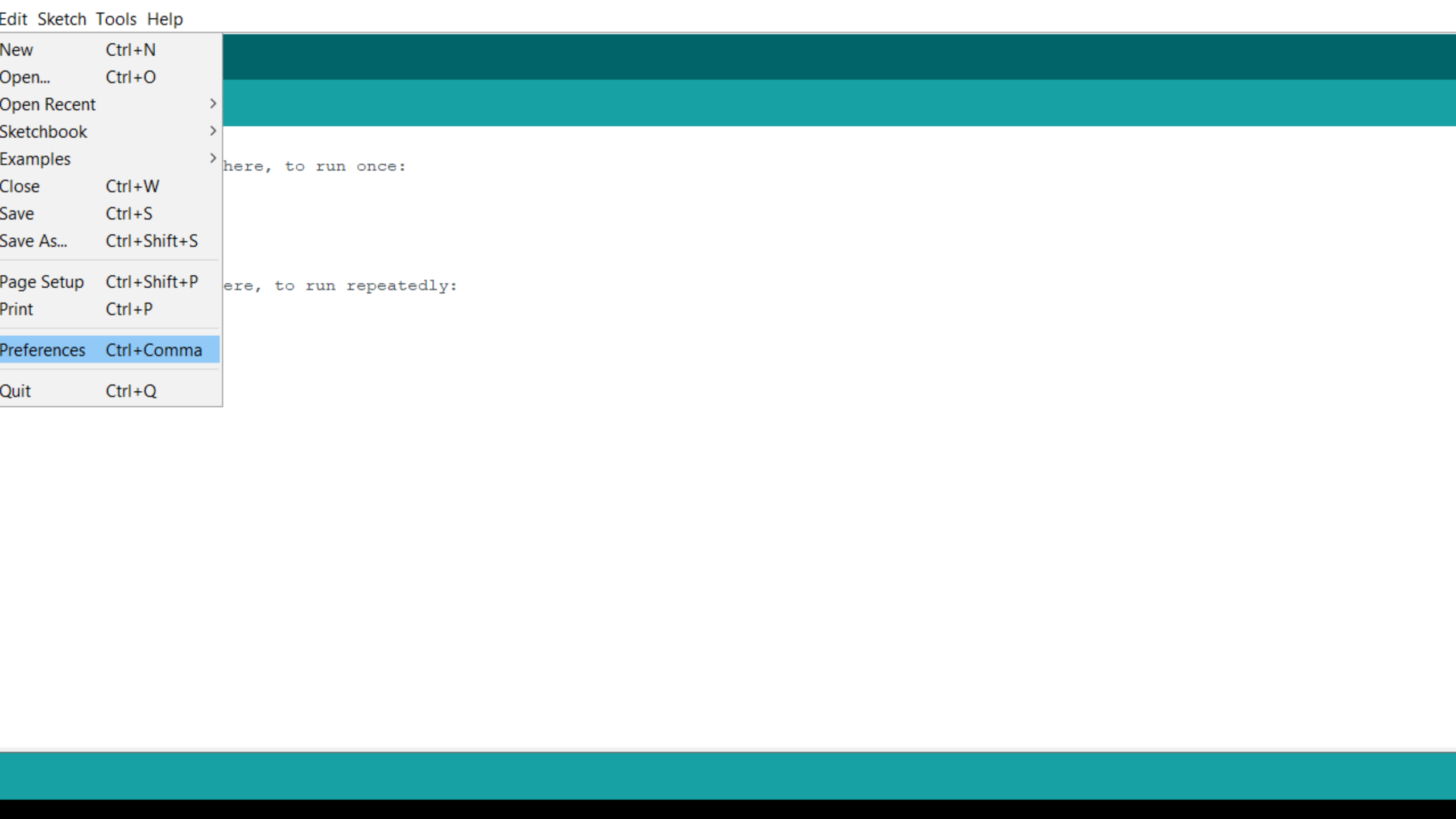
```
  https://www.arduino.cc/en/Main/Products

  modified 8 May 2014
  by Scott Fitzgerald
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

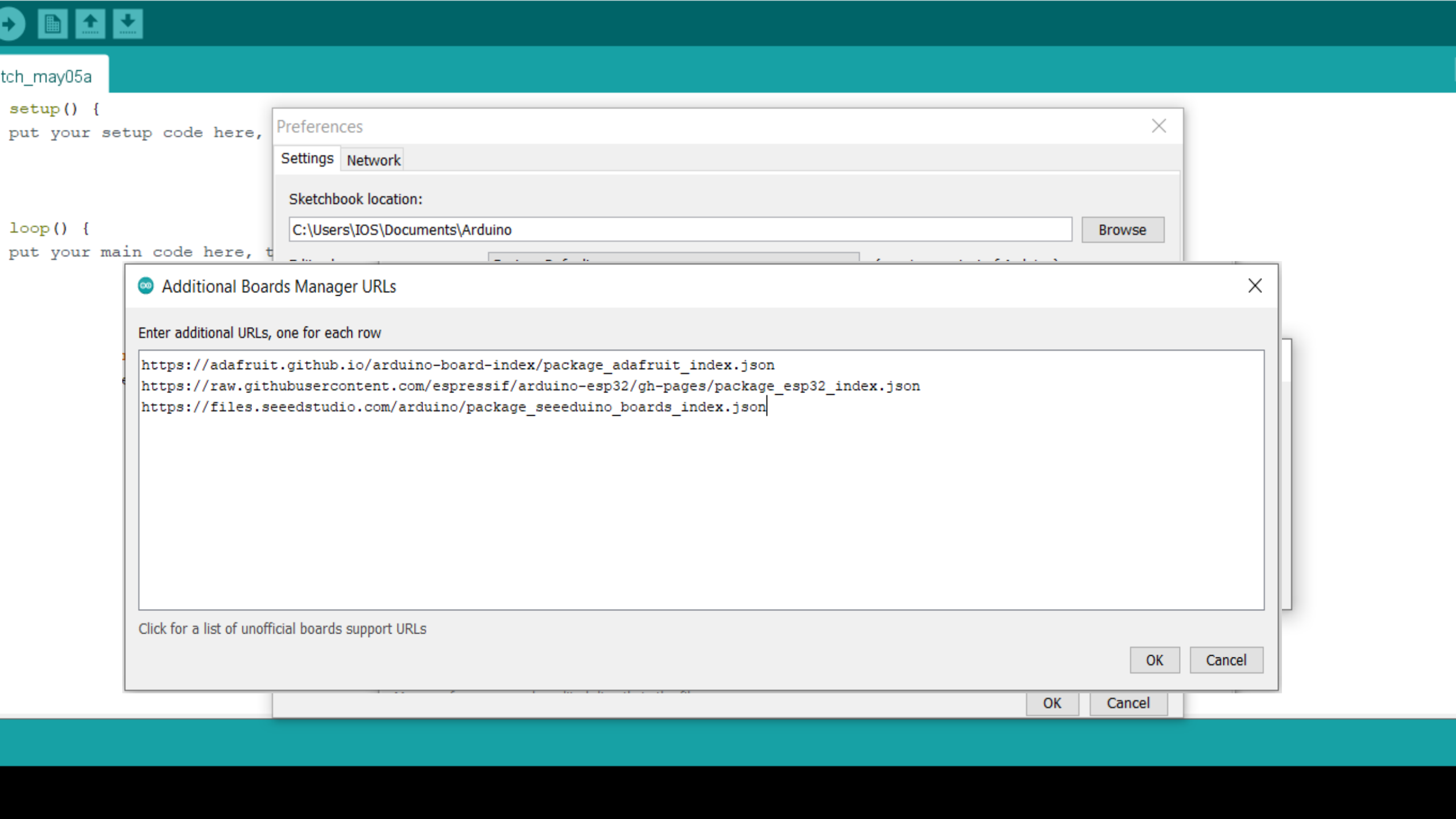
1

Adafruit ESP32 Feather, 80MHz, 921600, None, Default on COM4

09:55
11.05.2021


sketch_may05a | Arduino 1.8.13
File Edit Sketch Tools Help
sketch_may05a
void setup() {
// put your setup code here, to run once:
}
void loop() {
// put your ma
}
Preferences
Settings
Network
Sketchbook location:
C:\Users\IOS\Documents\Arduino
Browse
Editor language:
System Default
(requires restart of Arduino)
Editor font size:
12
Interface scale:
Automatic
100
%
(requires restart of Arduino)
Theme:
Default theme
(requires restart of Arduino)
Show verbose output during:
compilation
upload
Compiler warnings:
None
Display line numbers
Enable Code Folding
Verify code after upload
Use external editor
Check for updates on startup
Save when verifying or uploading
Use accessibility features
Additional Boards Manager URLs:
ıdex.json,https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json
More preferences can be edited directly in the file
C:\Users\IOS\AppData\Local\Arduino15\preferences.txt
(edit only when Arduino is not running)
OK
Cancel
at javax.swing.JTable.prepareRenderer(JTable.java:5729)
at javax.swing.plaf.basic.BasicTableUI.paintCell(BasicTableUI.java:2114)
at javax.swing.plaf.basic.BasicTableUI.paintCells(BasicTableUI.java:2016)
1
Arduino Uno


Edit Sketch Tools Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q
here, to run once:
ere, to run repeatedly:


tch_may05a
setup() {
put your setup code here,
loop() {
put your main code here, t
Preferences
Settings
Network
Sketchbook location:
C:\Users\IOS\Documents\Arduino
Browse
Additional Boards Manager URLs
Enter additional URLs, one for each row
https://adafruit.github.io/arduino-board-index/package_adafruit_index.json
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json
Click for a list of unofficial boards support URLs
OK
Cancel
OK
Cancel

tch_may05a

```
setup() {
put your setup c

loop() {
put your main co
```

Boards Manager

Type All

by **Arduino**
Boards included in this package:
Arduino Uno WiFi Rev2, Arduino Nano Every.
Online Help
More Info

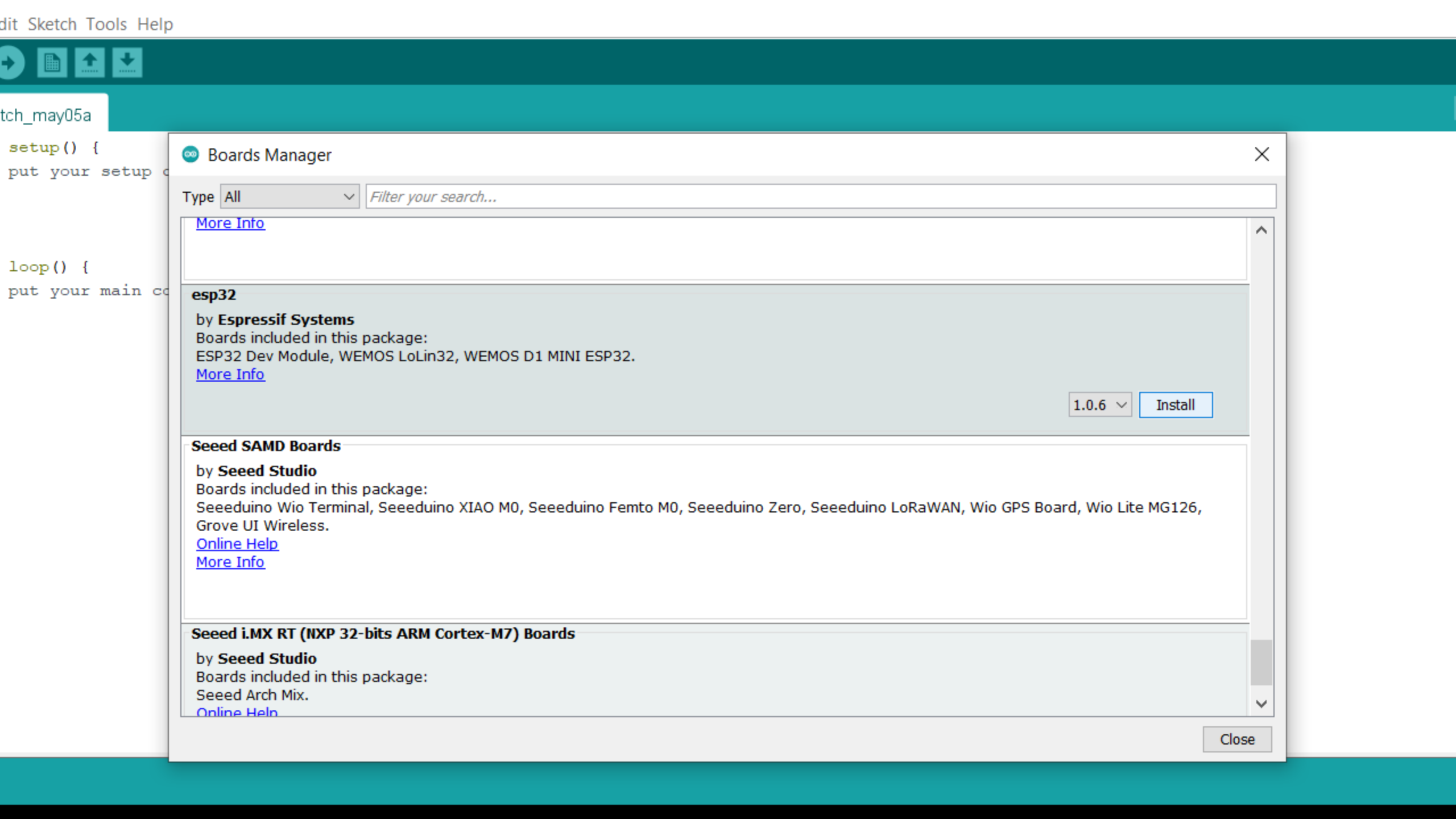
Arduino SAM Boards (32-bits ARM Cortex-M3)

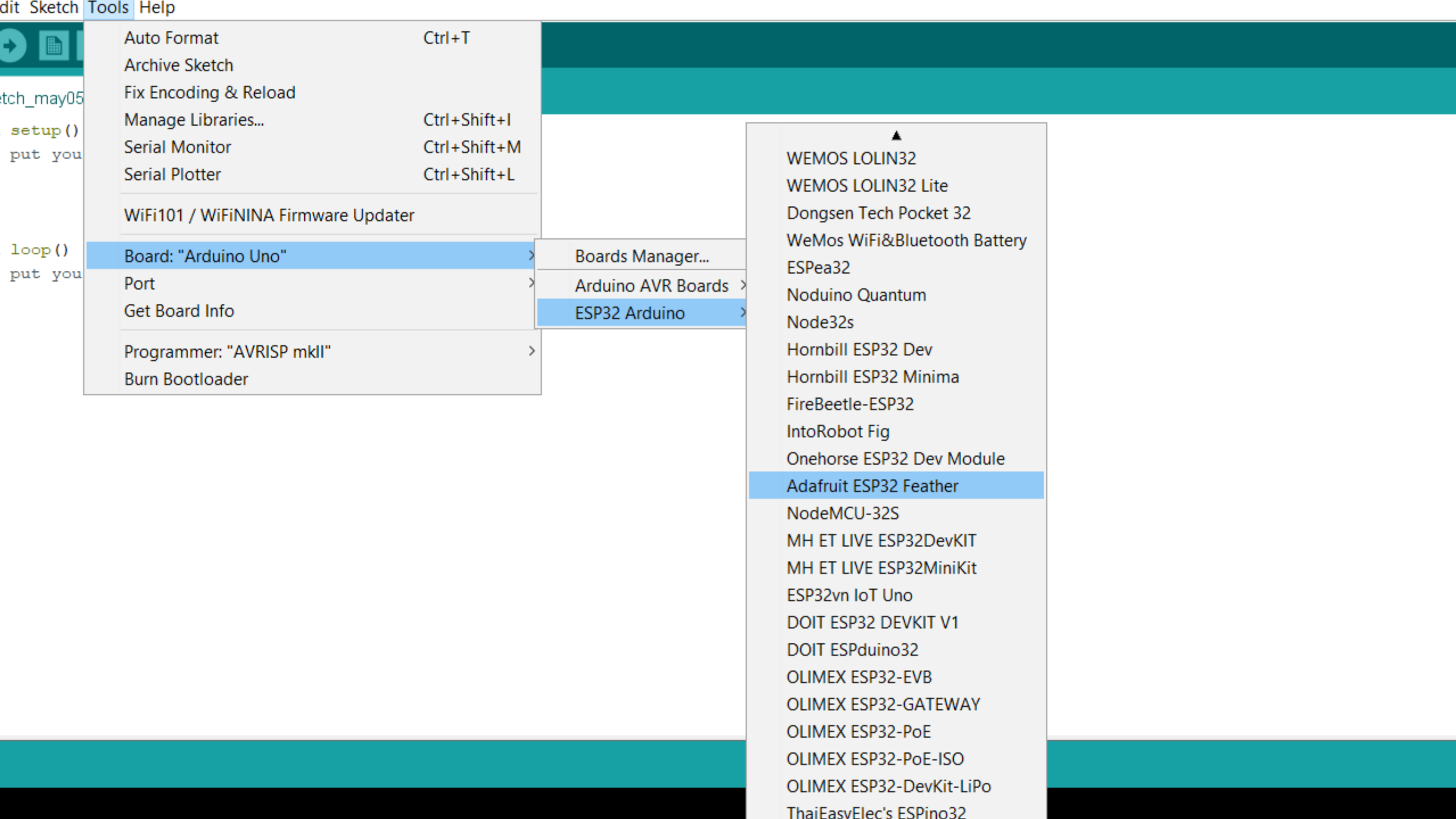
by **Arduino**
Boards included in this package:
Arduino Due.
Online Help
More Info

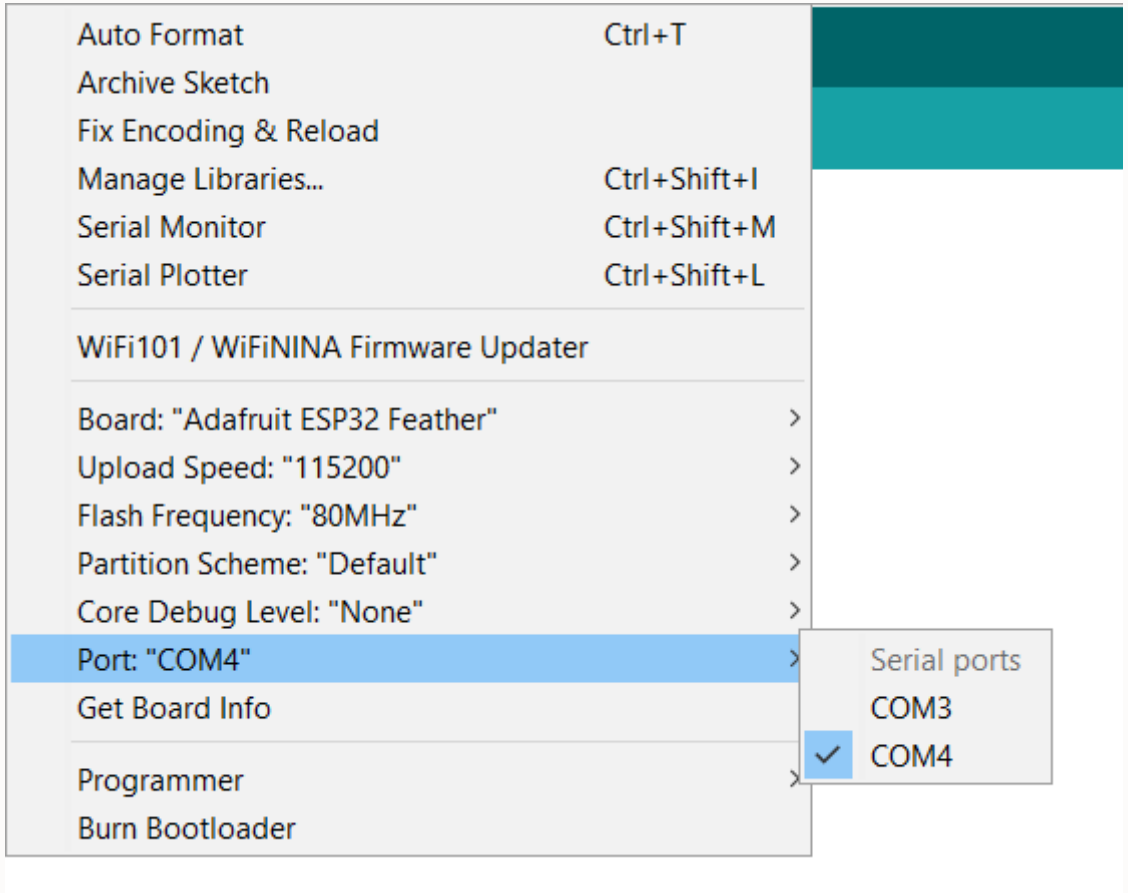
Arduino SAMD Boards (32-bits ARM Cortex-M0+)

by **Arduino**
Boards included in this package:
Arduino MKR WiFi 1010, Arduino Zero, Arduino MKR1000, Arduino MKRZERO, Arduino MKR FOX 1200, Arduino MKR WAN 1300, Arduino MKR WAN 1310, Arduino MKR GSM 1400, Arduino MKR NB 1500, Arduino MKR Vidor 4000, Arduino Nano 33 IoT, Arduino M0 Pro, Arduino M0, Arduino Tian, Adafruit Circuit Playground Express.
Online Help
More Info

Close


dit Sketch Tools Help
tch_may05a
setup() {
put your setup c
loop() {
put your main co
Boards Manager
Type All
Filter your search...
More Info
esp32
by Espressif Systems
Boards included in this package:
ESP32 Dev Module, WEMOS LoLin32, WEMOS D1 MINI ESP32.
More Info
1.0.6
Install
Seeed SAMD Boards
by Seeed Studio
Boards included in this package:
Seeeduino Wio Terminal, Seeeduino XIAO M0, Seeeduino Femto M0, Seeeduino Zero, Seeeduino LoRaWAN, Wio GPS Board, Wio Lite MG126, Grove UI Wireless.
Online Help
More Info
Seeed i.MX RT (NXP 32-bits ARM Cortex-M7) Boards
by Seeed Studio
Boards included in this package:
Seeed Arch Mix.
Online Help
Close


dit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Uno"
Port
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Boards Manager...
Arduino AVR Boards
ESP32 Arduino
WEMOS LOLIN32
WEMOS LOLIN32 Lite
Dongsen Tech Pocket 32
WeMos WiFi&Bluetooth Battery
ESPea32
Noduino Quantum
Node32s
Hornbill ESP32 Dev
Hornbill ESP32 Minima
FireBeetle-ESP32
IntoRobot Fig
Onehorse ESP32 Dev Module
Adafruit ESP32 Feather
NodeMCU-32S
MH ET LIVE ESP32DevKIT
MH ET LIVE ESP32MiniKit
ESP32vn IoT Uno
DOIT ESP32 DEVKIT V1
DOIT ESPduino32
OLIMEX ESP32-EVB
OLIMEX ESP32-GATEWAY
OLIMEX ESP32-PoE
OLIMEX ESP32-PoE-ISO
OLIMEX ESP32-DevKit-LiPo
ThaiEasyElec's ESPino32


Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Adafruit ESP32 Feather"
Upload Speed: "115200"
Flash Frequency: "80MHz"
Partition Scheme: "Default"
Core Debug Level: "None"
Port: "COM4"
Get Board Info
Programmer
Burn Bootloader
Serial ports
COM3
COM4

Blink | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

Blink

```
/*
  Blink

  Turns an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
  it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
  the correct LED pin independent of which board is used.
  If you want to know what pin the on-board LED is connected to on your Arduino
  model, check the Technical Specs of your board at:
  https://www.arduino.cc/en/Main/Products

  modified 8 May 2014
  by Scott Fitzgerald
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
```

Compiling sketch...

1

Adafruit ESP32 Feather on COM4

10:30
11.05.2021

Google

arduino i2c scanner

https://gist.github.com/tfeldmann/5411375

sketch_may11a §

```
// i2c_scanner
// Version 1
//    This program (or code that looks like it)
//    can be found in many places.
//    For example on the Arduino.cc forum.
//    The original author is not known.
// Version 2, Juni 2012, Using Arduino 1.0.1
//     Adapted to be as simple as possible by Arduino.cc user Krodal
// Version 3, Feb 26  2013
//    V3 by louarnold
// Version 4, March 3, 2013, Using Arduino 1.0.3
//    by Arduino.cc user Krodal.
//    Scanning addresses changed from 0...127 to 1...119,
//    according to the i2c scanner by Nick Gammon
//    http://www.gammon.com.au/forum/?id=10896
// Version 5, March 28, 2013
//    As version 4, but address scans now to 127.
//    A sensor seems to use address 120.
// This sketch tests the standard 7-bit addresses
// Devices with higher bit address might not be seen properly.
#include <Wire.h>
void setup()
{
  Wire.begin();
  Serial.begin(9600);
  Serial.println("\nI2C Scanner");
}
```

sketch_may11a | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

sketch_may11a

```
void loop(){
  byte error, address;
  int nDevices;
  Serial.println("Scanning...");
  nDevices = 0;
  for(address = 1; address < 127; 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("  !");
      nDevices++;
    }
    else if (error==4) {
      Serial.print("Unknow error at address 0x");
      if (address<16)
        Serial.print("0");
      Serial.println(address,HEX);
    }
  }
  if (nDevices == 0) Serial.println("No I2C devices found\n");
  else Serial.println("done\n");
  delay(5000);           // wait 5 seconds for next scan
}
```

Compiling sketch...

1

Adafruit ESP32 Feather on COM4

10:52
11.05.2021

sketch_may11a | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

Serial Monitor

sketch_may11a

```
void loop(){
  byte error, address;
  int nDevices;
  Serial.println("Scanning...");
  nDevices = 0;
  for(address = 1; address < 127; 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("  !");
      nDevices++;
    }
    else if (error==4) {
      Serial.print("Unknow error at address 0x");
      if (address<16)
        Serial.print("0");
      Serial.println(address,HEX);
    }
  }
  if (nDevices == 0) Serial.println("No I2C devices found\n");
  else Serial.println("done\n");
  delay(5000);           // wait 5 seconds for next scan
}
```

COM4

Send

```
10:58:33.454 ->
10:58:38.370 -> Scanning...
10:58:38.417 -> No I2C devices found
10:58:38.464 ->
10:58:43.419 -> Scanning...
10:58:43.466 -> No I2C devices found
10:58:43.466 ->
10:58:48.426 -> Scanning...
10:58:48.472 -> No I2C devices found
10:58:48.472 ->
10:58:53.427 -> Scanning...
10:58:53.474 -> No I2C devices found
10:58:53.520 ->
10:58:58.471 -> Scanning...
10:58:58.518 -> I2C device found at address 0x77  !
10:58:58.565 -> done
10:58:58.565 ->
10:59:03.468 -> Scanning...
10:59:03.514 -> I2C device found at address 0x77  !
10:59:03.562 -> done
10:59:03.562 ->
```

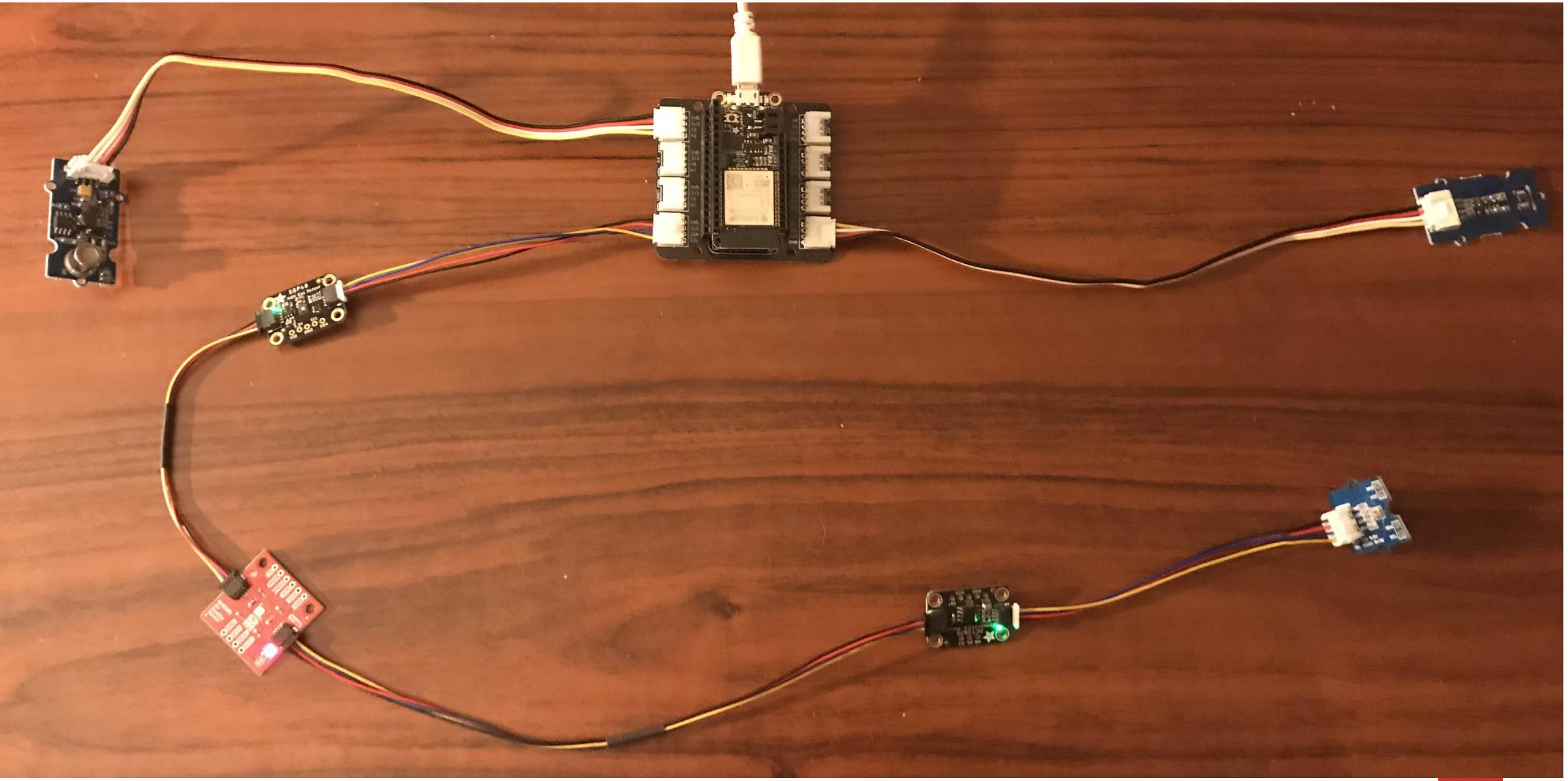
Autoscroll Show timestamp Newline 9600 baud Clear output

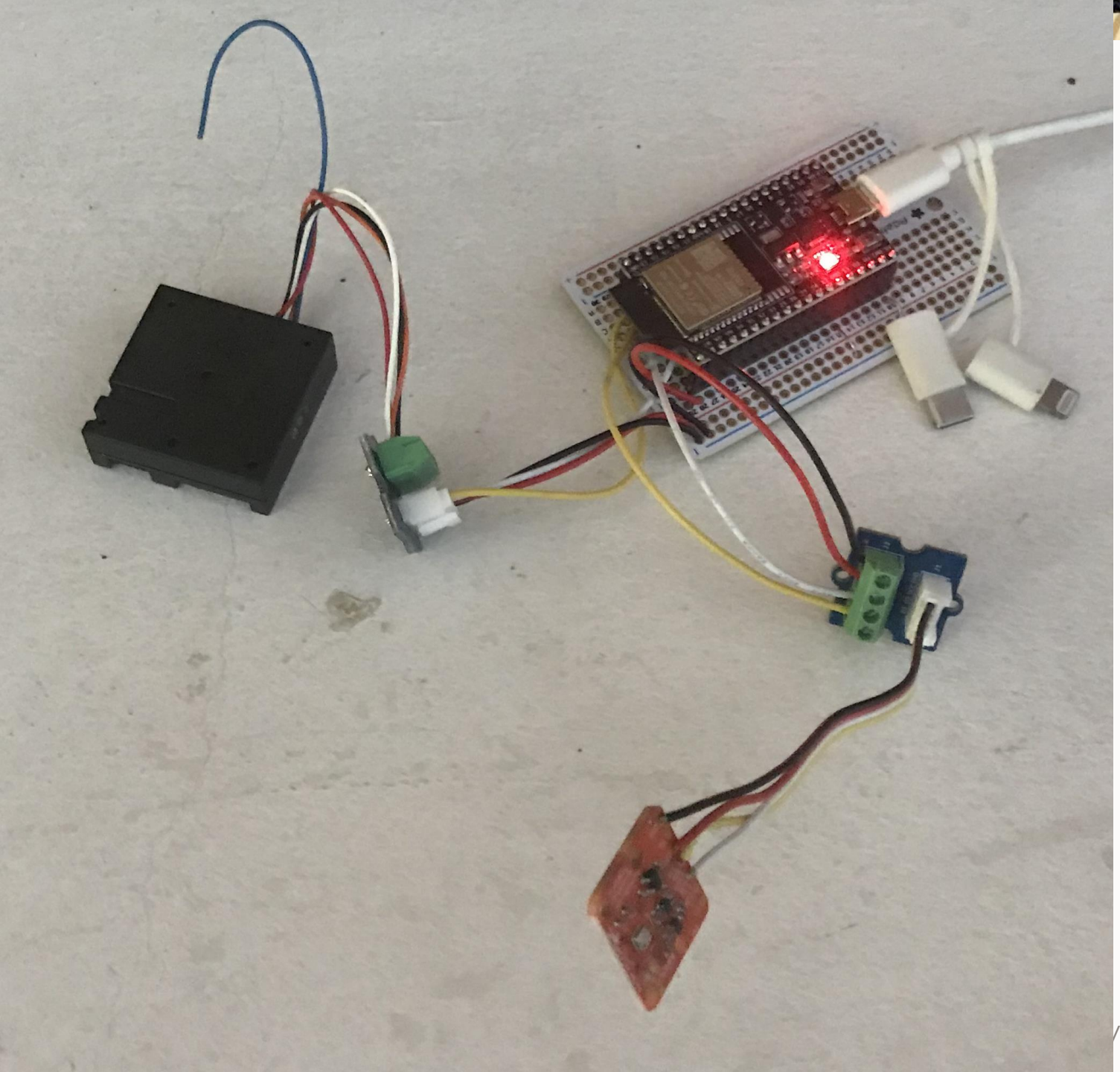
```
Leaving...
Hard resetting via RTS pin...
```

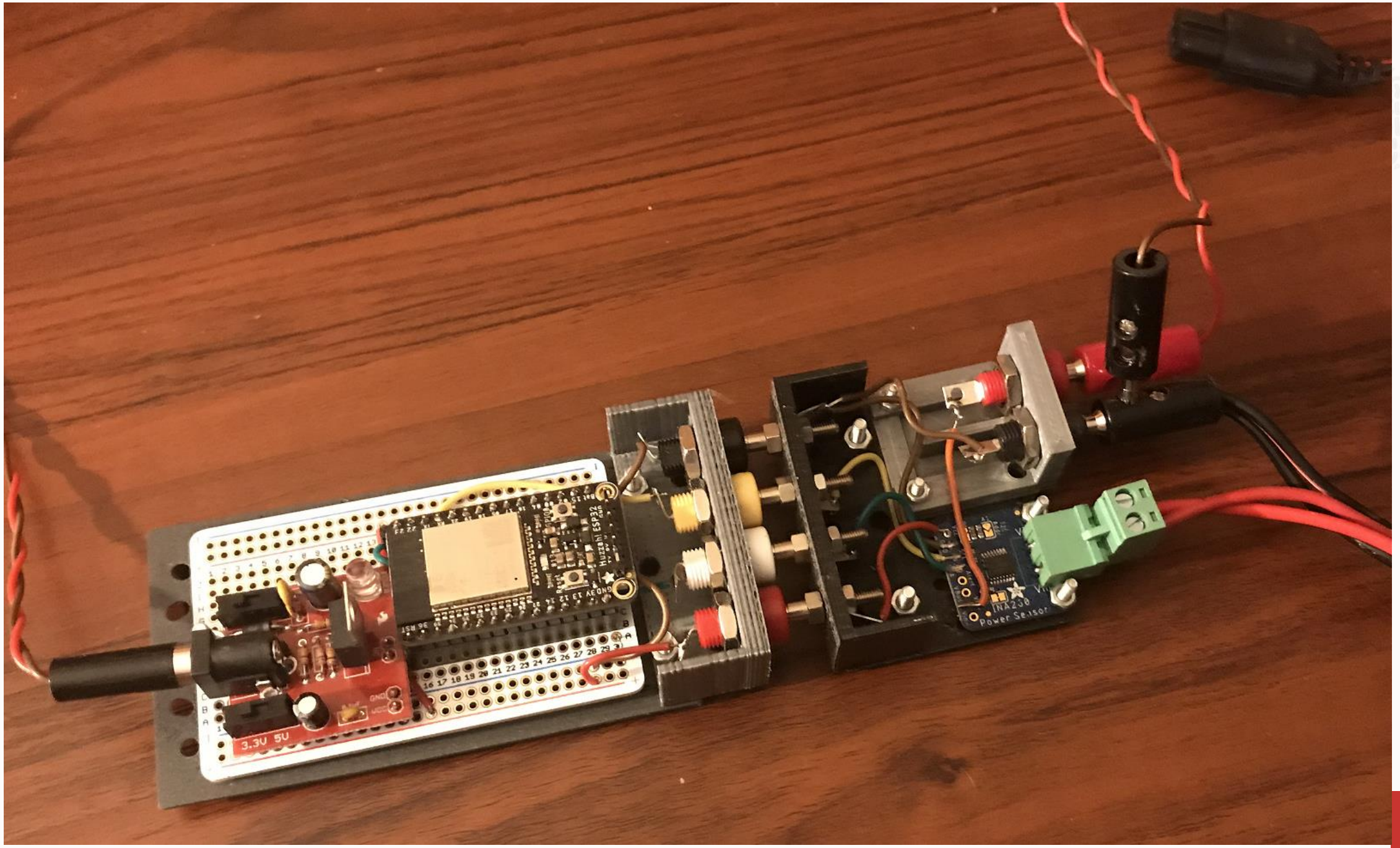
1

Adafruit ESP32 Feather on COM4

10:59
11.05.2021






3.3V 5V
INA230
Power Sensor

Adafruit BME680 - Temperature

https://www.adafruit.com/product/3660?gclid=EAIaIQobChMI78f_x6XB8AIVkNiy(

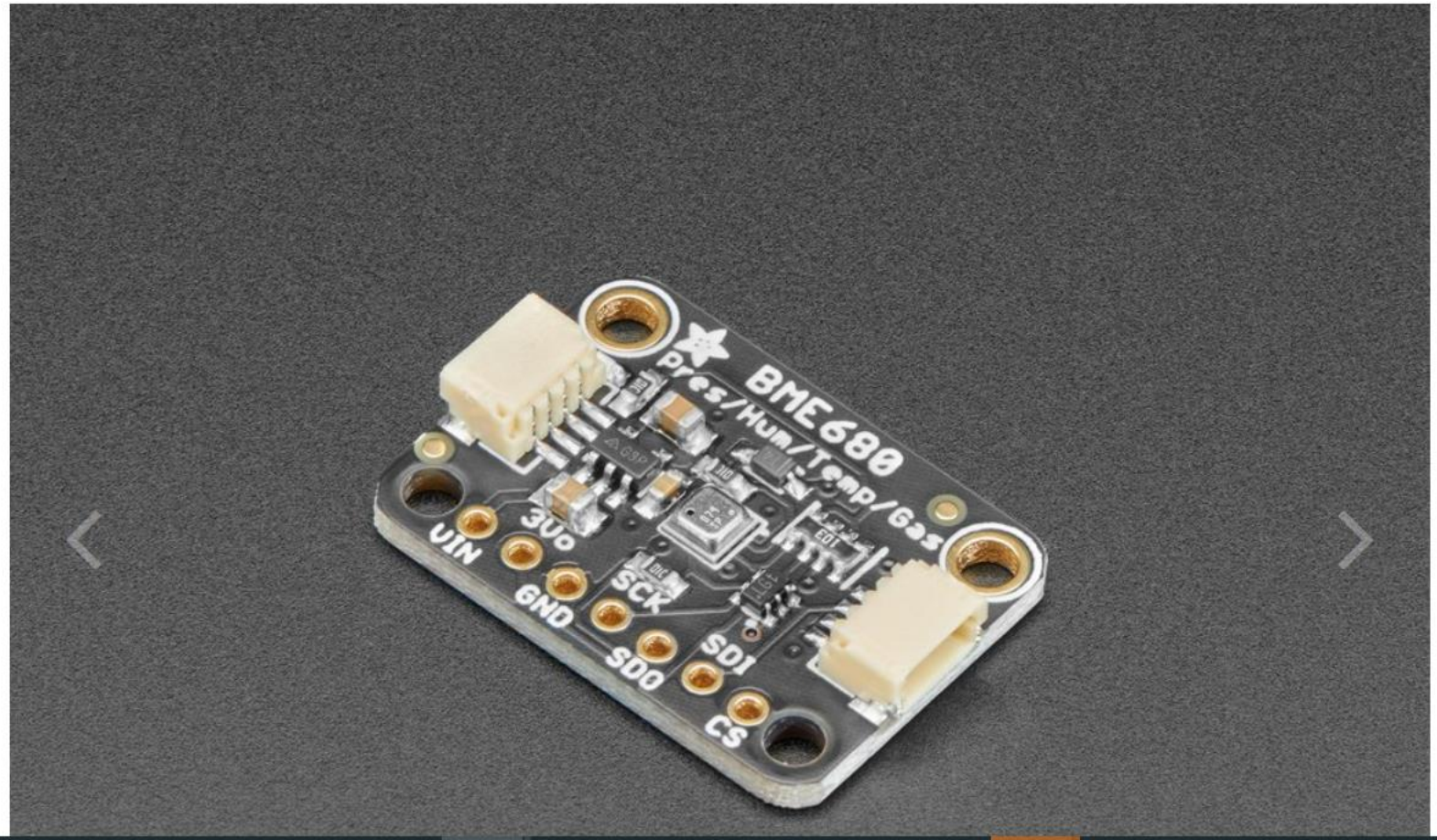
Google

adafruit bme680

Shop Learn Blog Forums LIVE! AdaBox IO

adafruit Products Gift Ideas What's New

SENSORS / TEMPERATURE / I2C / SPI DIGITAL / ADAFRUIT BME680 - TEMPERATURE, HUMIDITY, PRESSURE AND GAS SENSOR



Adafruit BME680 - Temperature, Humidity, Pressure and Gas Sensor - STEMMA QT

PRODUCT ID: 3660

$18.95

OUT OF STOCK

Please enter your details below and we will send you an email when this item is back in stock. You will only be emailed about this product!

YOUR EMAIL

NOTIFY ME

BUY ON DIGI-KEY

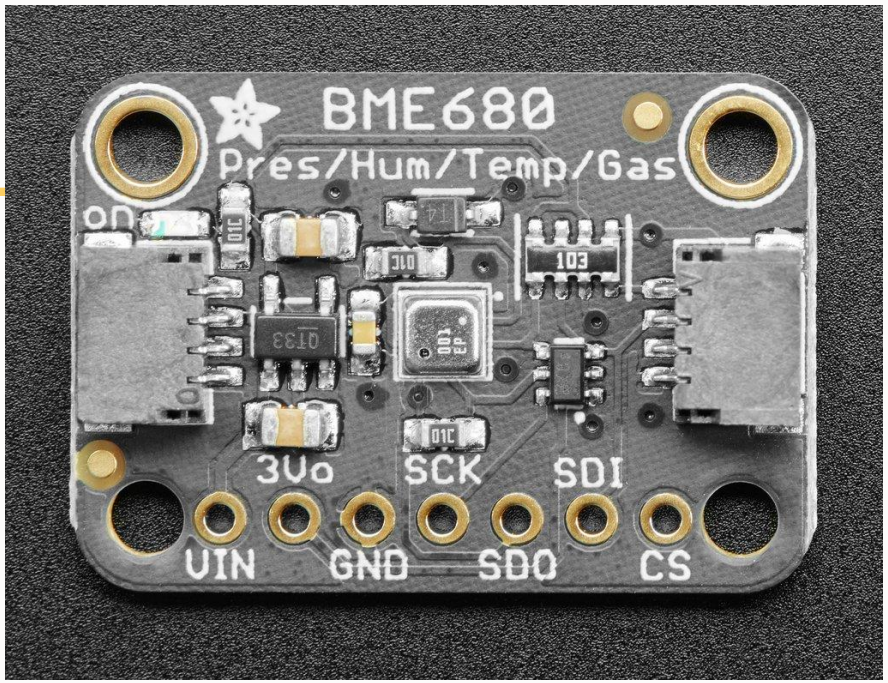
Why buy on Digi-Key?

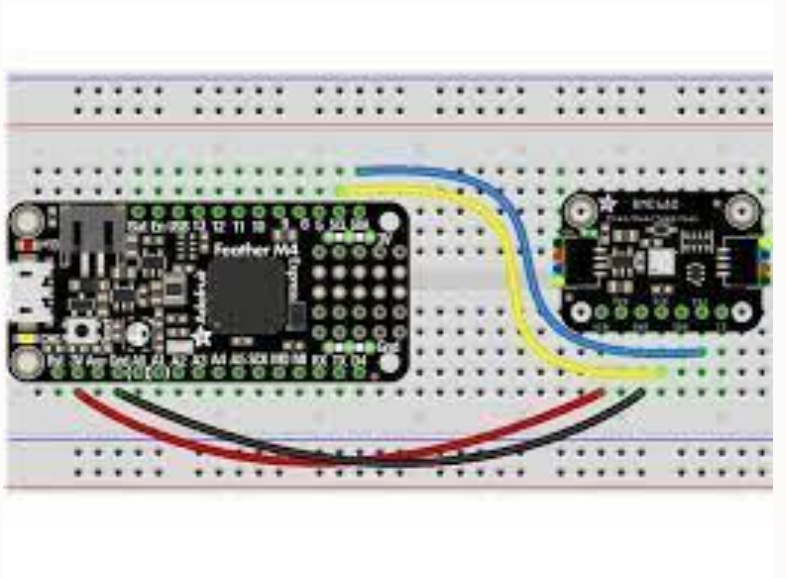
ADD TO WISHLIST

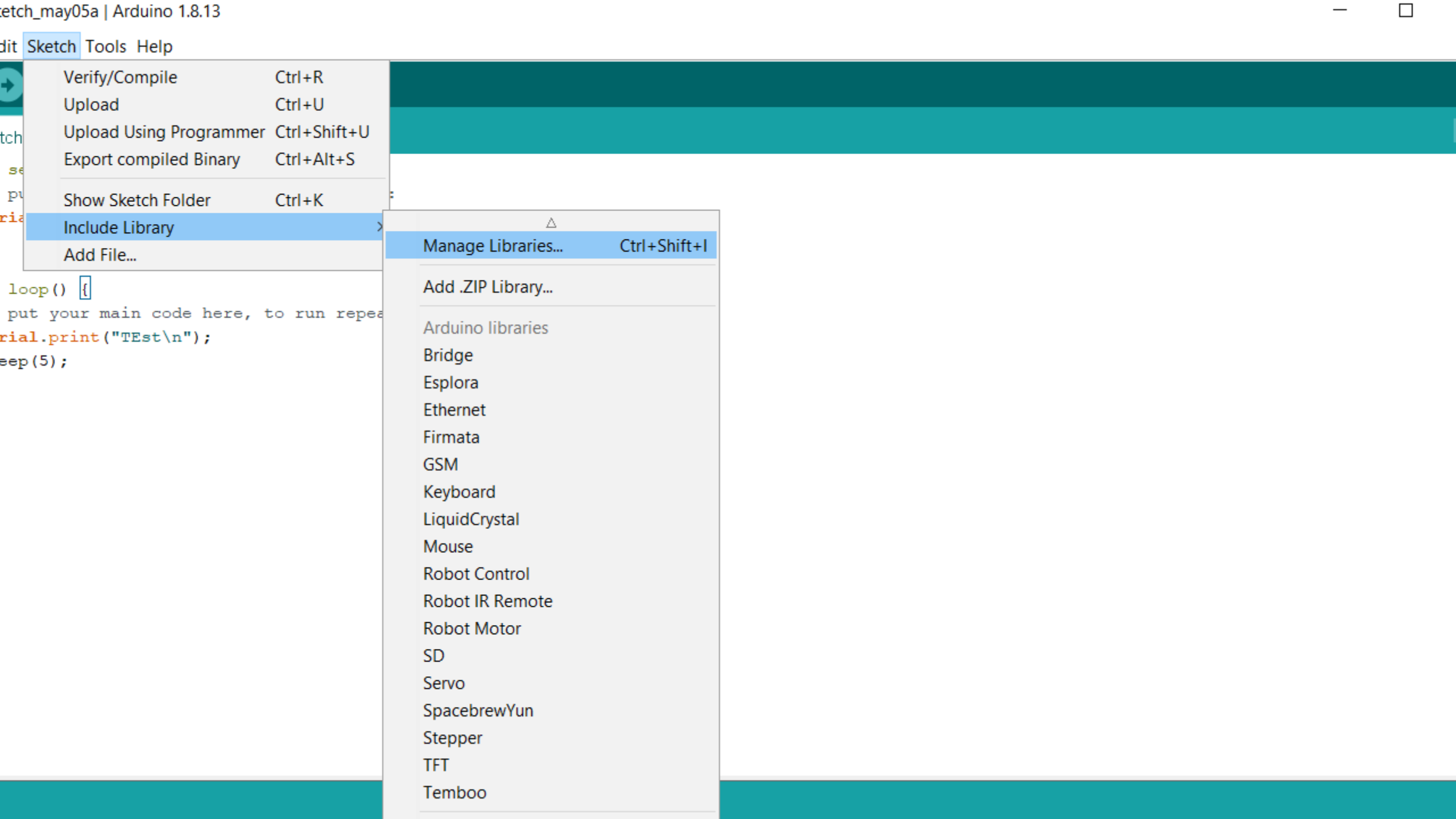
0.0 ★★★★★
No rating available

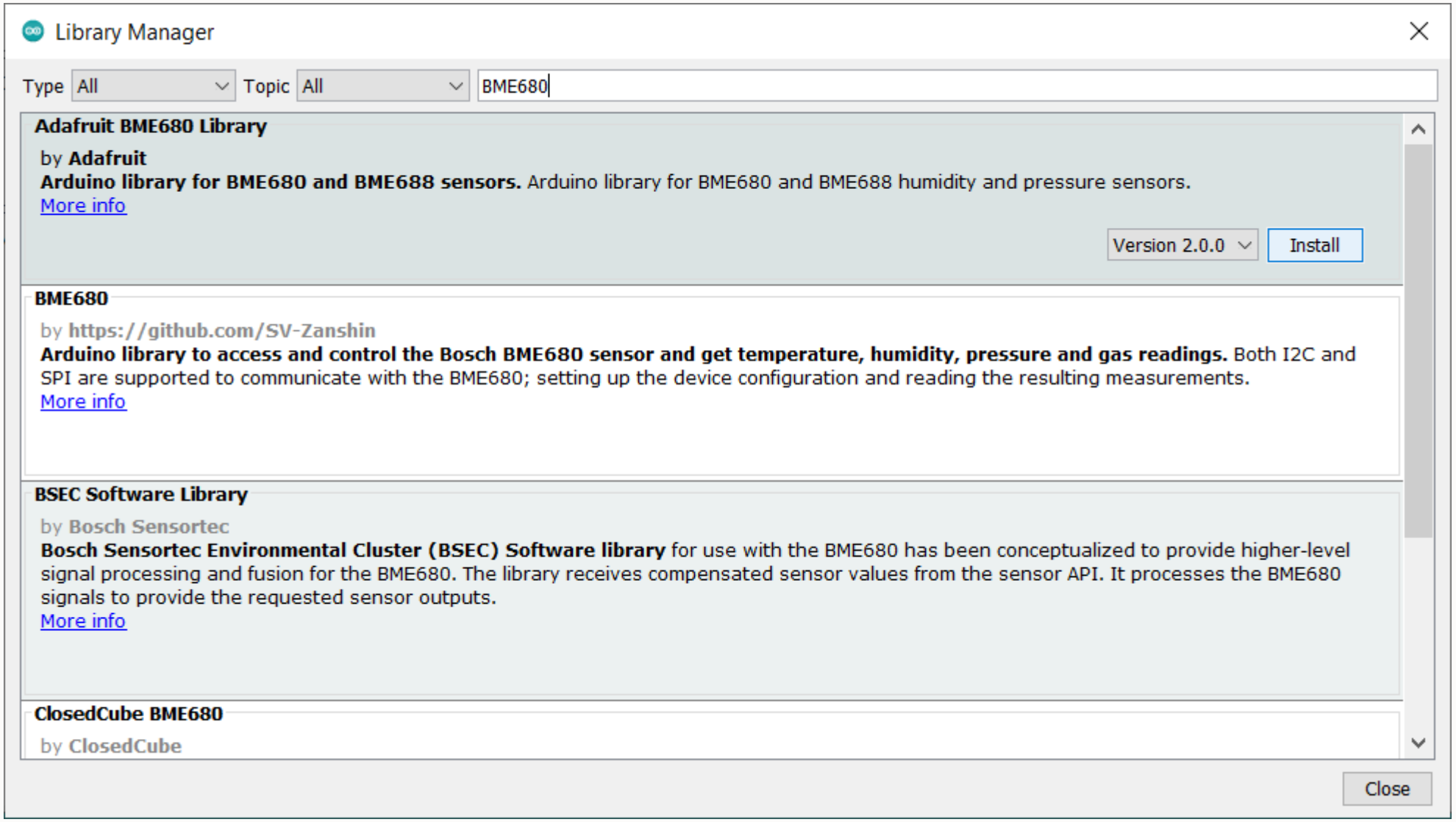
11:19
11.05.2021

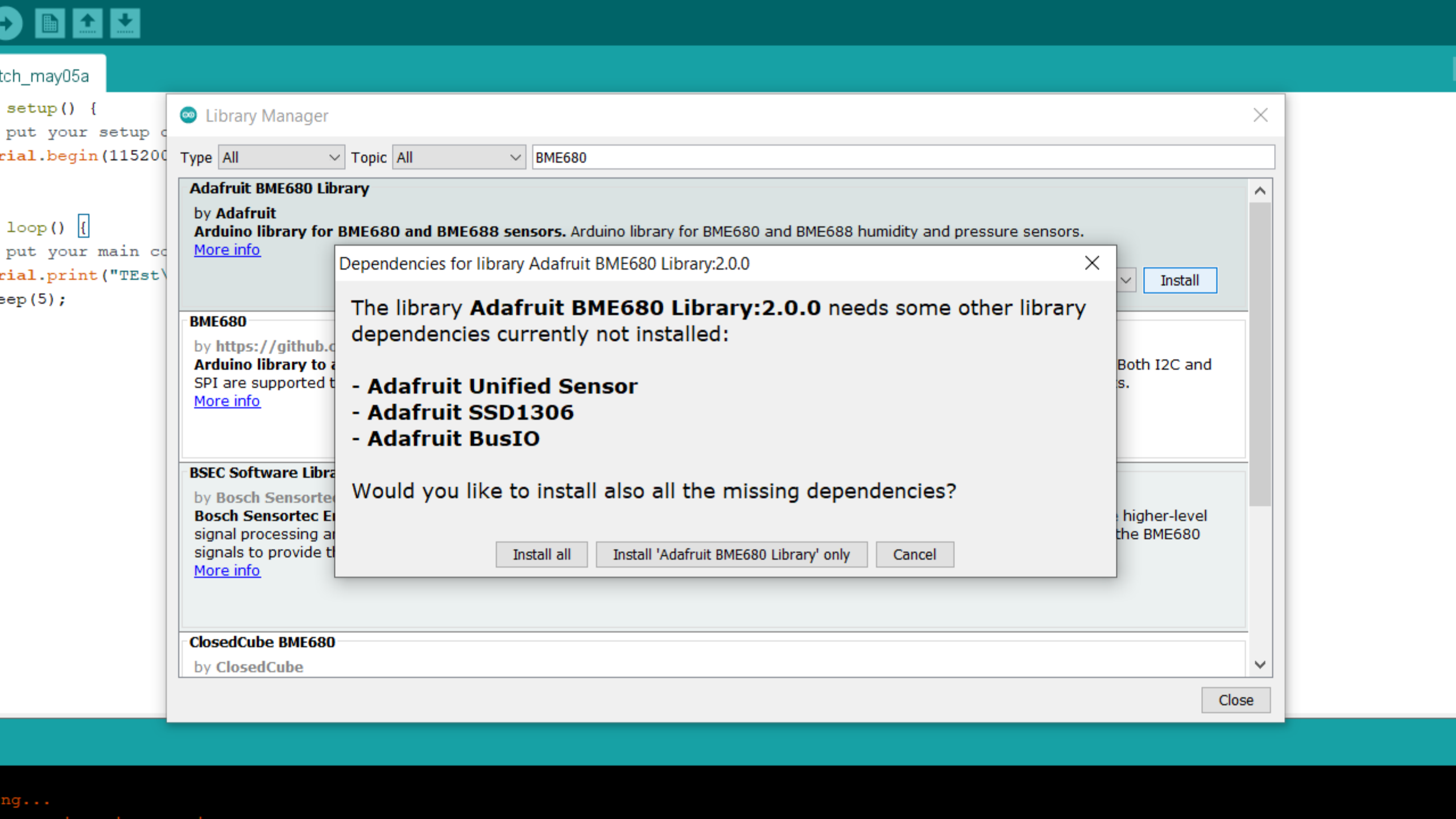
Google adafruit bme680

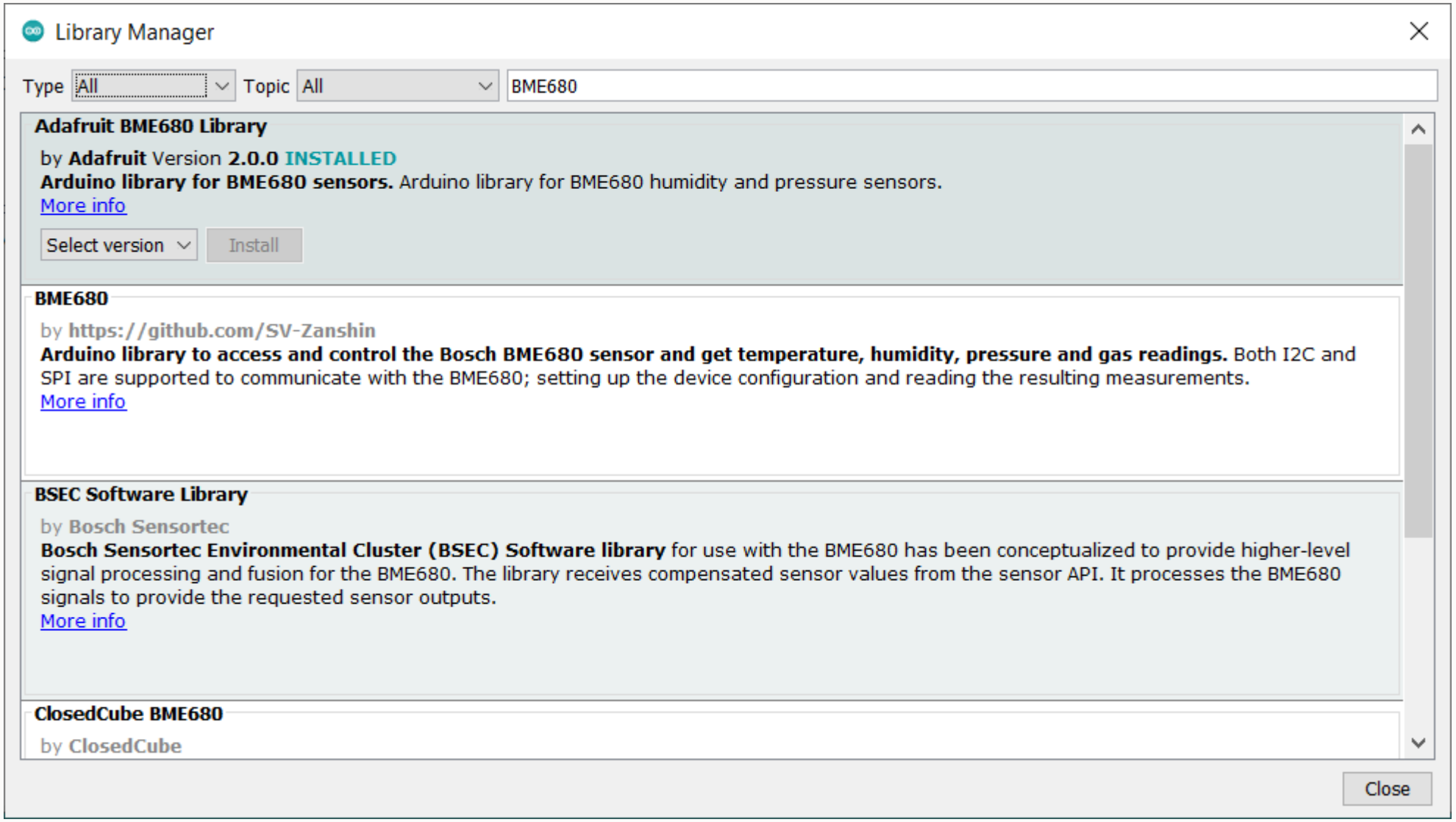


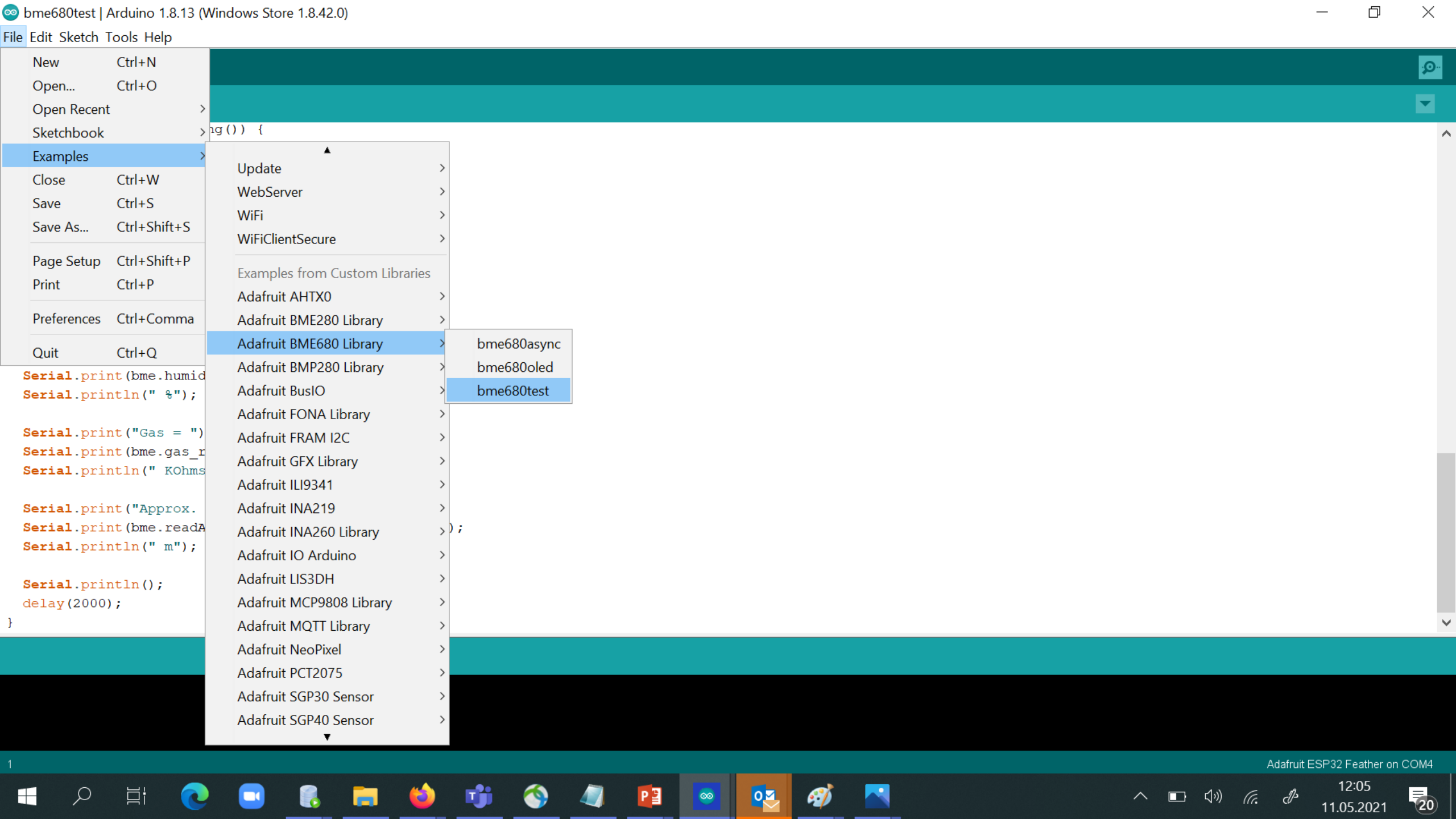



etch_may05a | Arduino 1.8.13
dit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
Manage Libraries... Ctrl+Shift+I
Add .ZIP Library...
Arduino libraries
Bridge
Esplora
Ethernet
Firmata
GSM
Keyboard
LiquidCrystal
Mouse
Robot Control
Robot IR Remote
Robot Motor
SD
Servo
SpacebrewYun
Stepper
TFT
Temboo
loop() {
put your main code here, to run repea
rial.print("TEst\n");
eep(5);


Library Manager
Type All Topic All BME680
Adafruit BME680 Library
by Adafruit
Arduino library for BME680 and BME688 sensors. Arduino library for BME680 and BME688 humidity and pressure sensors.
More info
Version 2.0.0 Install
BME680
by https://github.com/SV-Zanshin
Arduino library to access and control the Bosch BME680 sensor and get temperature, humidity, pressure and gas readings. Both I2C and SPI are supported to communicate with the BME680; setting up the device configuration and reading the resulting measurements.
More info
BSEC Software Library
by Bosch Sensortec
Bosch Sensortec Environmental Cluster (BSEC) Software library for use with the BME680 has been conceptualized to provide higher-level signal processing and fusion for the BME680. The library receives compensated sensor values from the sensor API. It processes the BME680 signals to provide the requested sensor outputs.
More info
ClosedCube BME680
by ClosedCube
Close


tch_may05a
setup() {
put your setup c
rial.begin(11520
loop() {
put your main co
rial.print("TEst
eep(5);
Library Manager
Type All
Topic All
BME680
Adafruit BME680 Library
by Adafruit
Arduino library for BME680 and BME688 sensors. Arduino library for BME680 and BME688 humidity and pressure sensors.
More info
Install
BME680
by https://github.c
Arduino library to a
SPI are supported t
More info
Both I2C and
s.
BSEC Software Libra
by Bosch Sensortec
Bosch Sensortec Et
signal processing ar
signals to provide th
More info
higher-level
the BME680
ClosedCube BME680
by ClosedCube
Close
Dependencies for library Adafruit BME680 Library:2.0.0
The library Adafruit BME680 Library:2.0.0 needs some other library dependencies currently not installed:
- Adafruit Unified Sensor
- Adafruit SSD1306
- Adafruit BusIO
Would you like to install also all the missing dependencies?
Install all
Install 'Adafruit BME680 Library' only
Cancel
ng...


Library Manager
Type All Topic All BME680
Adafruit BME680 Library
by Adafruit Version 2.0.0 INSTALLED
Arduino library for BME680 sensors. Arduino library for BME680 humidity and pressure sensors.
More info
Select version Install
BME680
by https://github.com/SV-Zanshin
Arduino library to access and control the Bosch BME680 sensor and get temperature, humidity, pressure and gas readings. Both I2C and SPI are supported to communicate with the BME680; setting up the device configuration and reading the resulting measurements.
More info
BSEC Software Library
by Bosch Sensortec
Bosch Sensortec Environmental Cluster (BSEC) Software library for use with the BME680 has been conceptualized to provide higher-level signal processing and fusion for the BME680. The library receives compensated sensor values from the sensor API. It processes the BME680 signals to provide the requested sensor outputs.
More info
ClosedCube BME680
by ClosedCube
Close


bme680test | Arduino 1.8.13 (Windows Store 1.8.42.0)
File Edit Sketch Tools Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q
Update
WebServer
WiFi
WiFiClientSecure
Examples from Custom Libraries
Adafruit AHTX0
Adafruit BME280 Library
Adafruit BME680 Library
Adafruit BMP280 Library
Adafruit BusIO
Adafruit FONA Library
Adafruit FRAM I2C
Adafruit GFX Library
Adafruit ILI9341
Adafruit INA219
Adafruit INA260 Library
Adafruit IO Arduino
Adafruit LIS3DH
Adafruit MCP9808 Library
Adafruit MQTT Library
Adafruit NeoPixel
Adafruit PCT2075
Adafruit SGP30 Sensor
Adafruit SGP40 Sensor
bme680async
bme680oled
bme680test
Serial.print(bme.humid
Serial.println(" %");
Serial.print("Gas = ")
Serial.print(bme.gas_r
Serial.println(" KOhms
Serial.print("Approx.
Serial.print(bme.readA
Serial.println(" m");
Serial.println();
delay(2000);
}
Adafruit ESP32 Feather on COM4
12:05
11.05.2021

meetup20210511demo | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

meetup20210511demo §

```
  Serial.print("Temperature = ");
  Serial.print(bme.temperature);
  Serial.println(" *C");
  Serial.print("Pressure = ");
  Serial.print(bme.pressure / 100.0);
  Serial.println(" hPa");
  Serial.print("Humidity = ");
  Serial.print(bme.humidity);
  Serial.println(" %");
  Serial.print("Gas = ");
  Serial.print(bme.gas_resistance / 1000.0);
  Serial.println(" KOhms");
  Serial.print("Approx. Altitude = ");
  Serial.print(bme.readAltitude(SEALEVELPRESSURE_HPA));
  Serial.println(" m");
  Serial.print("Gas = ");
  Serial.print(bme.gas_resistance / 1000.0);
  Serial.println(" KOhms");
  Serial.println();
  if (updateCloud){
    float temp=bme.temperature;
    temp=temp*1000;
    int tempi=(int) temp;
    postCloud(sensorId,"TempMC",tempi);
    postCloud(sensorId,"Pres",bme.pressure/100);
    postCloud(sensorId,"Hum",bme.humidity);
```

Done compiling.

```
Sketch uses 883454 bytes (67%) of program storage space. Maximum is 1310720 bytes.
Global variables use 40672 bytes (12%) of dynamic memory, leaving 287008 bytes for local variables. Maximum is 327680 bytes.
```

262

Adafruit ESP32 Feather on COM4

12:23
11.05.2021

COM4

Send

```
12:32:18.407 -> load:0x40078000,len:8896
12:32:18.407 -> load:0x40080400,len:5816
12:32:18.407 -> entry 0x400806ac
12:32:18.733 -> Setup...1
12:32:18.733 -> BME680 init
12:32:18.733 -> BME680 ready
12:32:18.733 -> Setup...2
12:32:18.733 -> Setup...3
12:32:20.743 -> Setup...4
12:32:20.743 -> ----------------------------
12:32:20.743 -> setupESP:
12:32:20.743 -> SSID:8016D8BD9E7C
12:32:20.743 -> Id:ESP32:8016D8BD9E7C
12:32:20.743 -> ----------------------------
12:32:20.743 -> Setup...5
12:32:20.743 -> Setup...6
12:32:20.743 -> ESP32 Chip ID = 8016D8BD9E7C
12:32:20.743 -> Id:ESP32:8016D8BD9E7C
12:32:20.883 -> Connect WIFI .
12:32:21.865 -> Connected, IP address: 192.168.0.161
12:32:32.269 -> Temperature = 22.50 *C
12:32:32.269 -> Pressure = 1012.19 hPa
12:32:32.269 -> Humidity = 42.51 %
12:32:32.269 -> Gas = 0.00 KOhms
12:32:32.269 -> Approx. Altitude = 9.00 m
12:32:32.644 -> Gas = 29.66 KOhms
12:32:32.644 ->
12:32:32.644 -> Post:{"objectname":"ESP32:8016D8BD9E7C:0:119:BME680","sensorname":"TempMC","sensorvalue":"22620"}
12:32:33.761 -> 200
12:32:33.761 -> Post:{"objectname":"ESP32:8016D8BD9E7C:0:119:BME680","sensorname":"Pres","sensorvalue":"1012"}
12:32:34.693 -> 200
12:32:34.693 -> Post:{"objectname":"ESP32:8016D8BD9E7C:0:119:BME680","sensorname":"Hum","sensorvalue":"42"}
12:32:35.671 -> 200
12:32:35.671 -> Post:{"objectname":"ESP32:8016D8BD9E7C:0:119:BME680","sensorname":"AirQ","sensorvalue":"29665"}
12:32:36.696 -> 200
```

Autoscroll Show timestamp

Newline 115200 baud Clear output

12:32
11.05.2021

meetup20210511demo | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

meetup20210511demo §

```
  Adafruit invests time and resources providing this open source code,
  please support Adafruit and open-source hardware by purchasing products
  from Adafruit!

  Written by Limor Fried & Kevin Townsend for Adafruit Industries.
  BSD license, all text above must be included in any redistribution
 ***************************************************************************/
 // Config :
int serialSpeed=115200;
//   wifi:
const char* ssid = "***SSID***";
const char* password = "***KEY***";
// Autonomous Database
String url = "https://tn1tv18ynzxubz5-iosp.adb.eu-frankfurt-1.oraclecloudapps.com/ords/sensordata/sensors/iotapi/";
String BME680sensorId=":0:119:BME680";
/***************************************************************************/
// config - timing (ms)
int DelayBeforeConnectWiFi=2000;
//int DelayBeforeUpdateCloud=10000;  //prod
int DelayBeforeUpdateCloud=5000; // test
int DelayBefore2ndUpdateCloud=0;
int DelayBeforeDisconnectWifi=1000;
//int DelayBeforeReboot=40000;  // prod
int DelayBeforeReboot=4000;  // test
//int DelayBeforeReboot=0;  // no reboot
boolean ConnectCloud=true;
/***************************************************************************/
```

```
Leaving...
Hard resetting via RTS pin...
```

32

Adafruit ESP32 Feather on COM4

12:37
11.05.2021

Setup() 1/2

```
void setup() {
 Serial.begin(serialSpeed);
 Serial.println ("Setup...1");
 while (!Serial);
 sensorBME680(Id+BME680sensorId,true,false,ConnectCloud);
 Serial.println ("Setup...2");
 if (ConnectCloud){
  Serial.println ("Setup...3");
  delay(DelayBeforeConnectWiFi);
  Serial.println ("Setup...4");
  setupESP();
  Serial.println ("Setup...5");
 }
 Serial.println ("Setup...6");
 uint64_t chipid;
 chipid=ESP.getEfuseMac();//The chip ID is essentially its MAC address(length: 6 bytes).
 Serial.printf("ESP32 Chip ID = %04X",(uint16_t)(chipid>>32));//print High 2 bytes
 Serial.printf("%08X\n",(uint32_t)chipid);//print Low 4bytes.
 Serial.println ("Id:"+Id);
 if (ConnectCloud){
```

Setup() 2/2

```
if (ConnectCloud){
    connectWiFi();
    delay(DelayBeforeUpdateCloud);
  }
  if (DelayBeforeReboot>0){
    sensorBME680(Id+BME680sensorId,false,true,ConnectCloud);
    if (DelayBefore2ndUpdateCloud>0){
      delay(DelayBefore2ndUpdateCloud);
      sensorBME680(Id+BME680sensorId,false,true,ConnectCloud);
    }
    delay(DelayBeforeDisconnectWifi);
    delay(DelayBeforeReboot);
    ESP.restart();
  }
}
```

```
void loop() {
  sensorBME680(Id+BME680sensorId,false,true,ConnectCloud);
  delay(DelayBeforeUpdateCloud);
}

void connectWiFi(){
  WiFi.begin(ssid, password);
  Serial.print("Connect WIFI ");
  int wificonnectcnt=0;
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
    wificonnectcnt++;
    if (wificonnectcnt>10){
      Serial.println ("Connect wifi error cnt:restart after 10 sec");
      delay(10000);
      ESP.restart();
    }
  }
  Serial.println();
  Serial.print("Connected, IP address: ");
  Serial.println(WiFi.localIP());
}
```

```
void postCloud(String sensorId,String sensorName,int sensorValue) {
 if(WiFi.status()== WL_CONNECTED){   //Check WiFi connection status
   HTTPClient http;
   http.begin(url,rootCACertificate);
   http.addHeader("Content-Type", "application/json");
   int httpResponseCode = 0;
   httpResponseCode =http.POST(getMsg(sensorId,sensorName,sensorValue));
   if(httpResponseCode>0){
     String response = http.getString(); //Get the response to the request
     Serial.println(httpResponseCode);   //Print return code
     if (DebugRest) Serial.println(response);          //Print request answer
     http.end();  //Free resources
   }
 }
 else{
    Serial.println("Error in WiFi connection");
 }
}
```

```
String getMsg(String sensorId,String sensorName,int sensorValue){
  String msg="{\"objectname\":\"";
    msg=msg+sensorId;
    msg=msg+"\",\"sensorname\":\"";
    msg=msg+sensorName;
    msg=msg+"\",\"sensorvalue\":\"";
    msg=msg+sensorValue;
    msg=msg+"\"}";
    Serial.print ("Post:");
    Serial.println (msg);
  return msg;
}
```