

ESP-WROOM-02開発キット

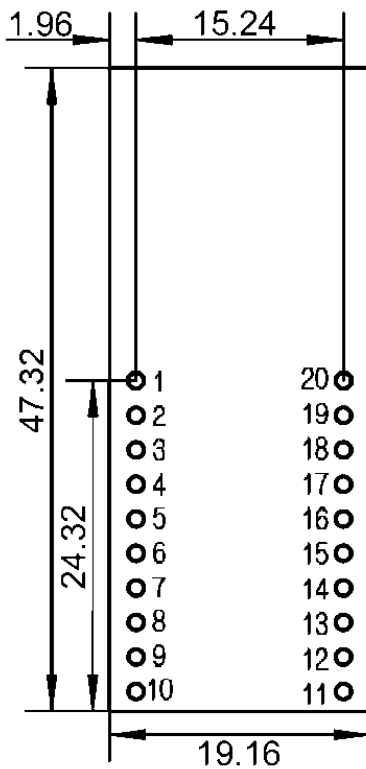
AE-ESP-WROOM-02-DEV

■特徴■

- ESP8266EX (SoC、32bit MCU) を搭載したWi-Fiモジュールを、使いやすい2.54mmピッチ10ピン2列のDIP形状(600mil幅)にしました。細ピンヘッダや丸ピンIC連結ソケット(両端オスピン)が使用できます。
- USB-シリアルインタフェースを搭載していますので、USBケーブルを接続すれば開発を始める事ができます。
- USB接続でArduinoスケッチを使用しての開発が可能です。
- 電源は5V、3.3Vが使用可能で、電源ラインに3.3V 800mAレギュレータと100 μ Fの電解コンデンサ搭載し、動作が安定しています。



■外形図■ (ピン穴径0.9mm)

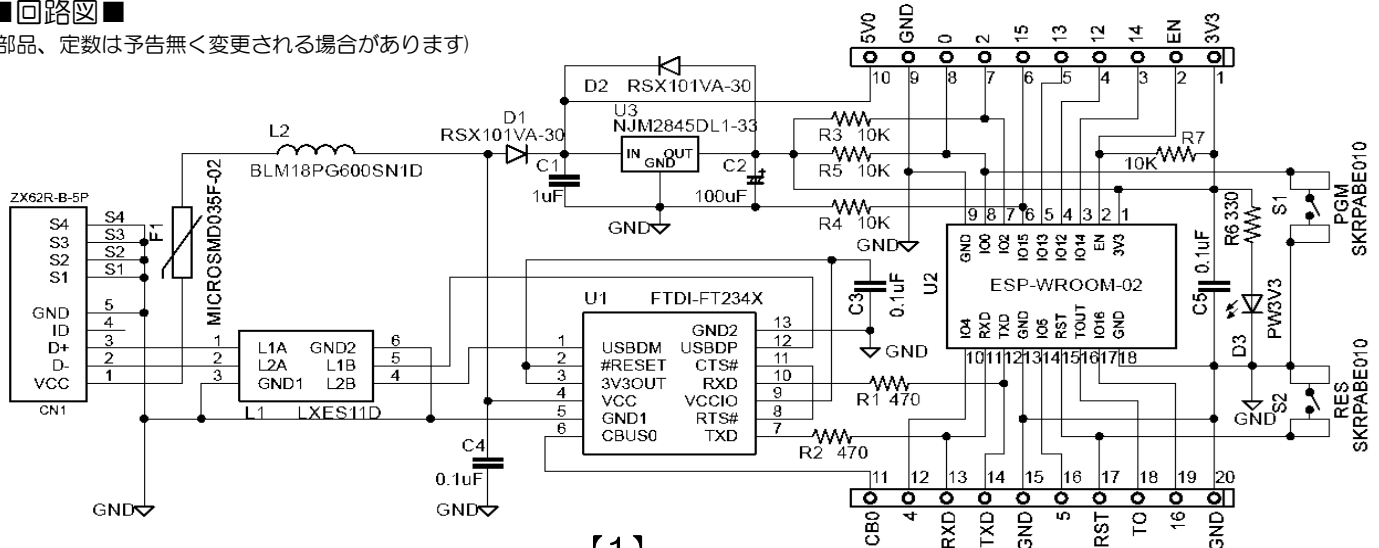


■ピンサイン■

番号	名称	説明
1	3V3	外部への3.3V出力、あるいは3.3V入力 USBや外部から5Vを入力している場合、入力として使用しないでください。
2	EN	イネーブル端子 10K Ω でプルアップされていて通常はイネーブル状態(使用可能状態)です。 GNDに接続する事でアンイネーブル状態(使用不可状態)になります。
3	14	GPIO14端子
4	12	GPIO12端子
5	13	GPIO13端子
6	15	GPIO15端子 10K Ω でプルダウンされています。
7	2	GPIO2端子 10K Ω でプルアップされています。
8	0	GPIO0端子 10K Ω でプルアップされPGMスイッチに接続されています。 PGMスイッチを押すとGNDに接続されます。
9	GND	GND端子
10	5V0	外部からの5V入力 (3V3入力とは併用できません) USBバスパワー5Vの出力としては使用しないでください。
11	CB0	FT234XのCBUS0端子 詳しい使用法はFTDI社FT234Xのマニュアルをご覧ください。
12	4	GPIO4端子
13	RXD	RXD端子 470 Ω を介してFT234XのTXDと接続されています。
14	TXD	TXD端子 470 Ω を介してFT234XのRXDと接続されています。
15	GND	GND端子
16	5	GPIO5端子
17	RST	RST端子 RSTスイッチに接続されています。RSTスイッチを押すと GNDに接続され、ESP-WROOM-02 はリセットされます。
18	TO	TOUT端子
19	16	GPIO16端子
20	GND	GND端子

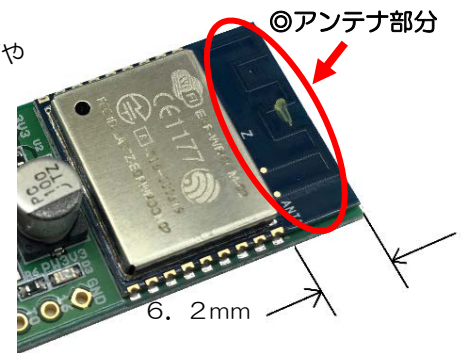
■回路図■

(部品、定数は予告無く変更される場合があります)



■基板などへの取り付け■

- 基板などに取り付ける時は電波の障害にならないように、右図赤丸枠のアンテナ裏面や上面に金属部分が近接しない状態で取り付けてください。(基板端から6.2mm)

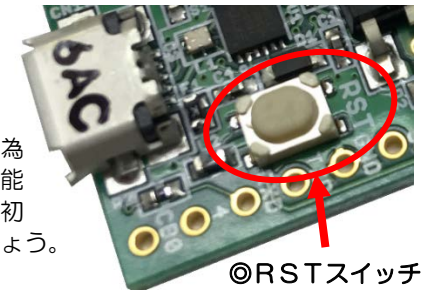


■接続■

- USBから電源を得る場合は、USB 3.0を使用するか、あるいはパワーハブを使用してください。USB 2.0では電流容量が足りない場合があります。

■使い方(ATコマンド)■

- USBコネクタとパソコンをUSB micro Bケーブルで接続します。
- ターミナルソフト(フリーソフトやArduinoスケッチのシリアルモニタ)を立ち上げます。
- デバイスマネージャを確認して、COMポート番号を正しく設定してください。
- ボーレートは115200bpsに設定してください。
- 改行コードの設定を「受信:CR」「送信:CR+LF」にします。
- 開発ボード上の赤LEDが点灯している事を確認して、RST(リセット)スイッチ(右図赤丸枠)をクリックします。
- 文字化けした表示が現れますが、リセット直後76800bpsで通信を行っている為です。ターミナルソフトにボーレート設定の項目があれば、変更する事で読む事が可能ですが、その後すぐに115200bpsで「ready」と送られて来ますので最初から115200bpsにしておいて、その文字が来たら正常と判断すれば良いでしょう。
- 「AT(+改行)」と入力します。「OK」の文字が表示されれば正常動作です。ATコマンドの詳細につきましては、下記メーカー資料をご参照ください。



http://espressif.com/sites/default/files/documentation/4a-esp8266_at_instruction_set_en.pdf

■使い方(Arduinoスケッチを使用)■

スケッチを使用してプログラムを書き込むと、ATコマンドでの制御はできなくなります。ご注意ください。

Arduinoスケッチを使用して、ESP-WROOM-02をArduinoとしてプログラムする方法です。

下記の操作を行い、開発キットにプログラムを書き込める状態にします。

◎PGMスイッチ

★インストール

- 下記ホームページのSOFTWAREからArduinoスケッチをダウンロードしてパソコンにインストールします。

<https://www.arduino.org/>

- 下記URLを参照してESP8266のボード情報を設定します。

<http://www.arduinesp.com/getting-started>

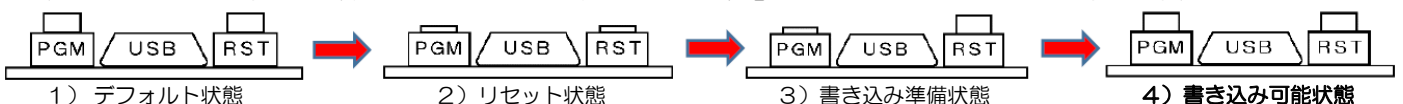
★開発

- パソコンと開発キットをUSB micro Bケーブルで接続します。
- スケッチ上でプログラムを作成します。

★書き込み(下図はUSB micro Bコネクタ側から見た図です)

- RSTスイッチとPGMスイッチ(右上図赤丸枠)を同時に押し、RST⇒PGMの順で離します。

(RSTスイッチを離れた時点でPGMスイッチが押されている状態にしてからPGMスイッチを離します)

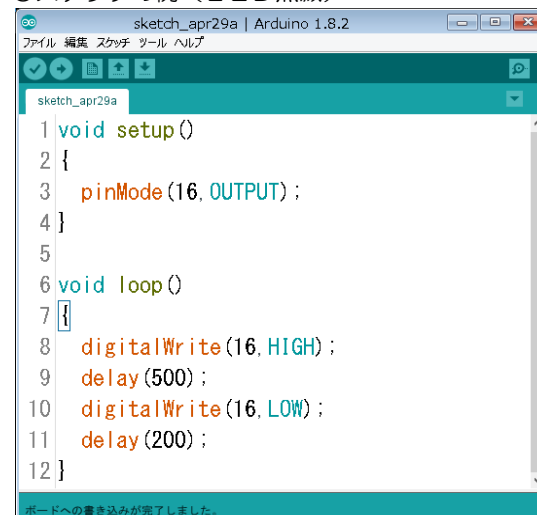


- 書き込み可能状態で、プログラムを転送します。

◎ボード設定例



◎スケッチの例(LED点滅)



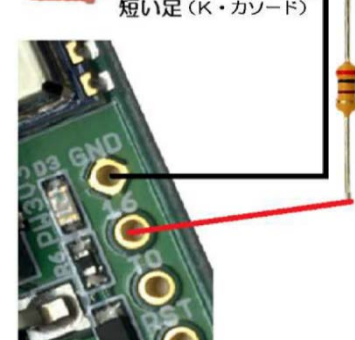
◎上記スケッチの配線例

GPIO16に接続したLEDを点滅させます。

この例ではLEDと開発ボードを200Ωの抵抗を介し接続します。

長い足(A・アノード)

短い足(K・カソード)

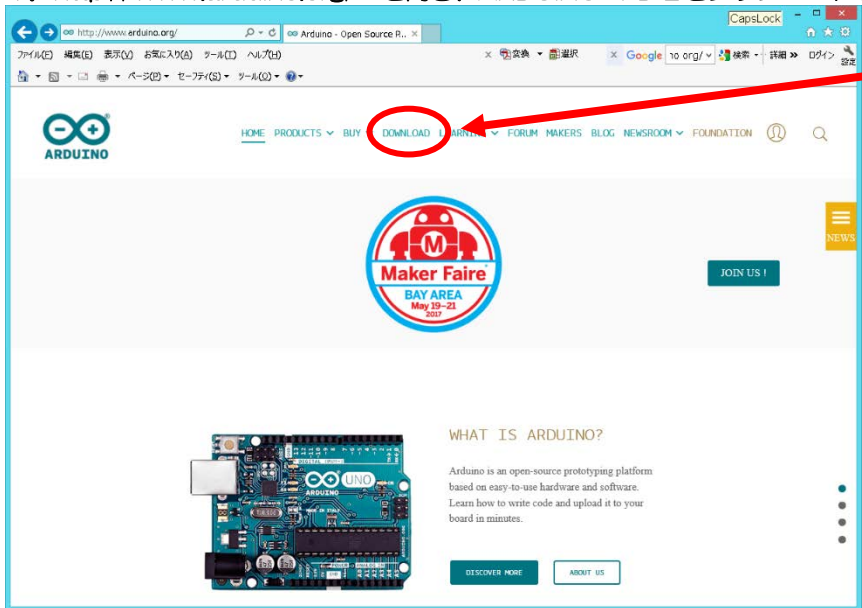


■詳しくはESP-WROOM-02、ESP8266メーカーマニュアルをご覧ください■

■インストールの手順詳細■（インターネットに接続されている必要があります）

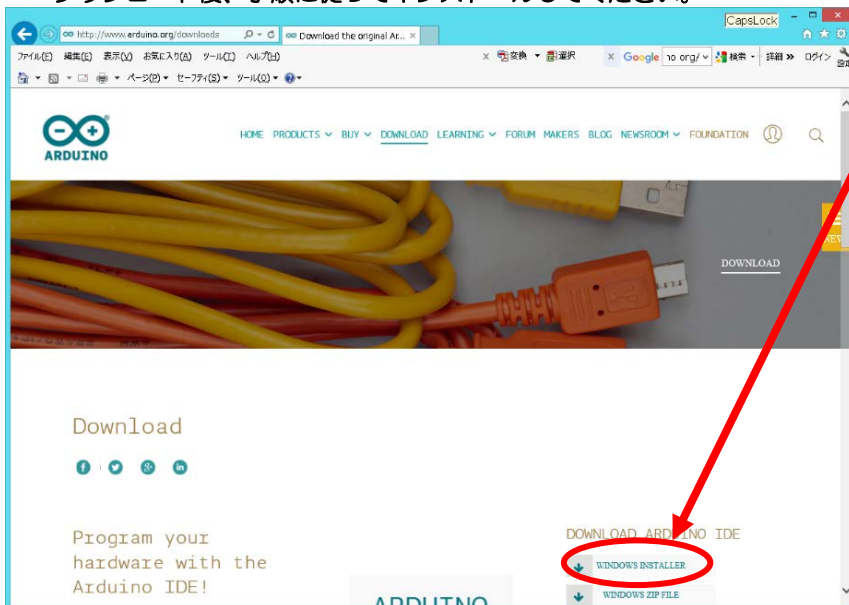
以下Windowsマシンでの説明です。

1. <http://www.arduino.org/> を開き、ARDUINO IDEをダウンロードします。



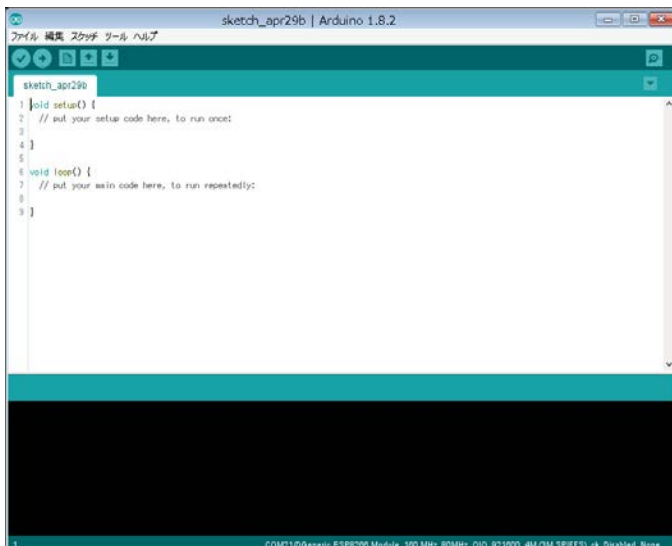
DOWNLOAD をクリック

2. インストーラーをダウンロードします。
ダウンロード後、手順に従ってインストールしてください。



WINDOWS INSTALLER をクリック。
実行または保存して実行してください。

3. インストールが終了したら、ソフトを立ち上げてください。



【3】

4. 環境設定をおこないます。

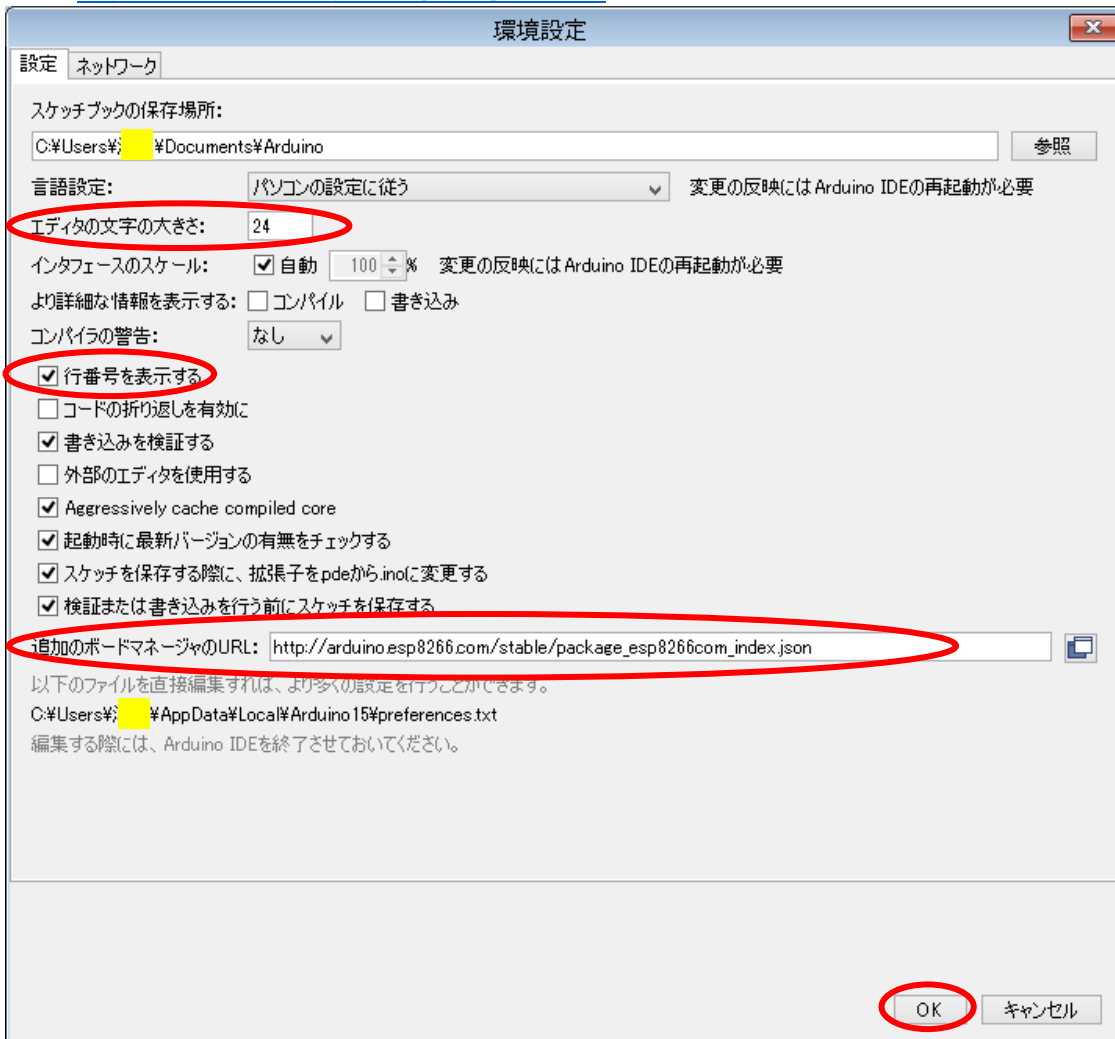
メニューバーの「ファイル」⇒「環境設定」を開きます。

- ・「エディタの文字の大きさ：」を適宜変更し、「行番号を表示する」にチェックを入れます。
- ・「追加のボードマネージャのURL：」に下記のように入力します。

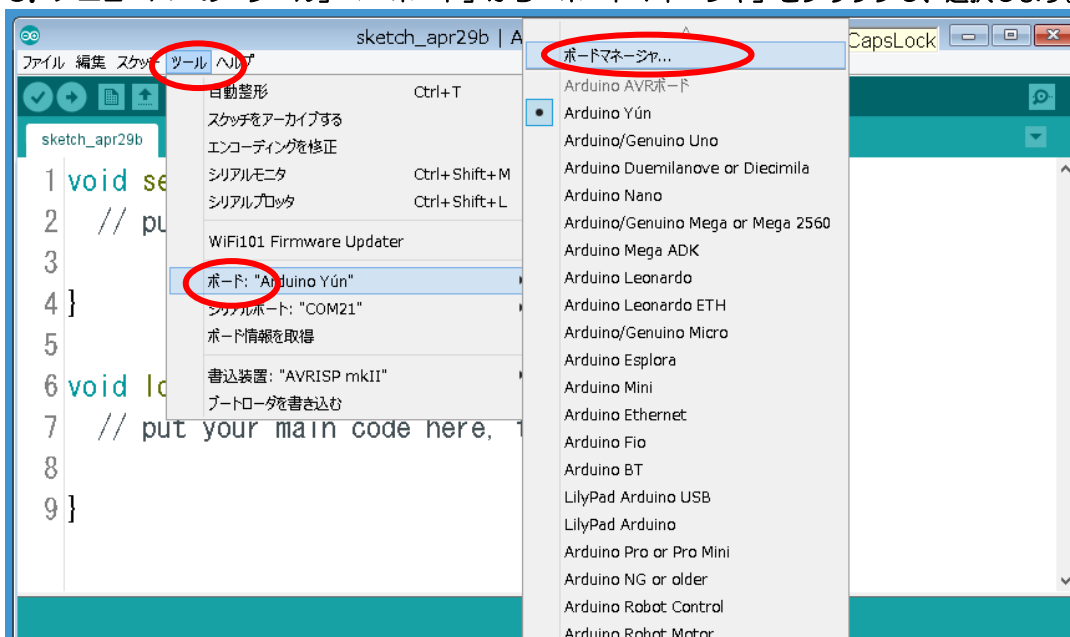
http://arduino.esp8266.com/stable/package_esp8266com_index.json

最下部の「OK」をクリックして「環境設定」を終了してください。

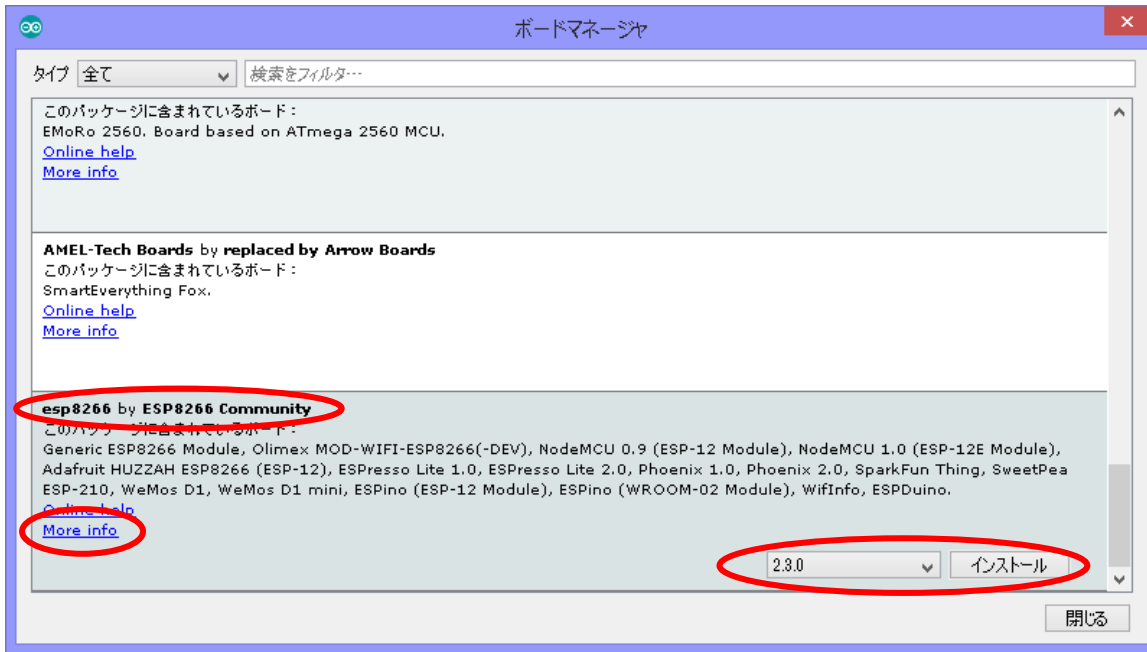
(<http://www.arduinesp.com/getting-started> も併せてご参照ください)



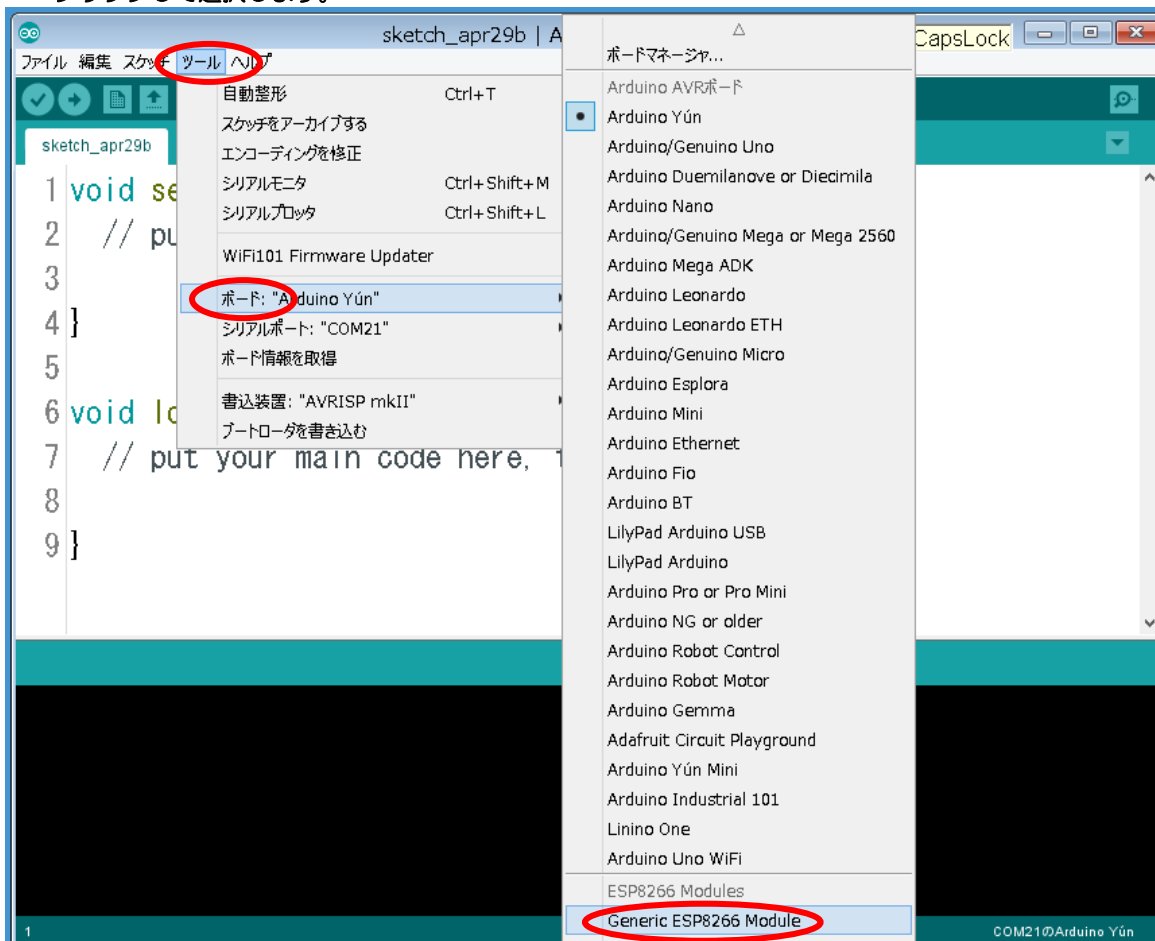
5. メニューバーの「ツール」⇒「ボード」から「ボードマネージャ」をクリックし、選択します。



6. ボードマネージャが開いたら、右の縦スクロールバーで一番下まで持っていき、esp2866を確認し、「More info」をクリックし、バージョンを選択して最新バージョンをインストールします。
 (インストールには時間がかかります。ダウンロードバイト数がアップカウントしているのを確認してください)

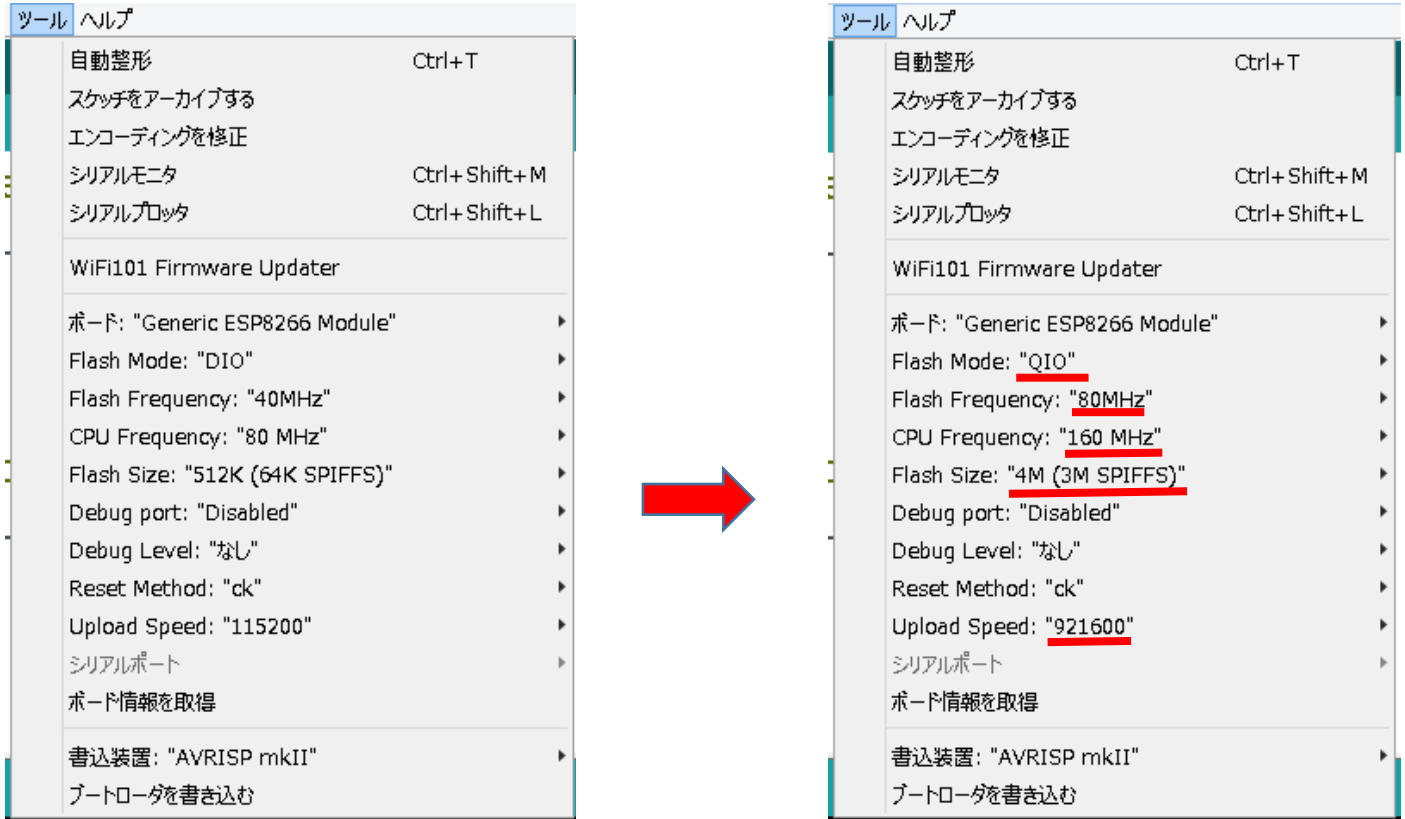


7. メニューバーの「ツール」⇒「ボード」から「Generic ESP8266 Module」が選択可能になっているので、クリックして選択します。



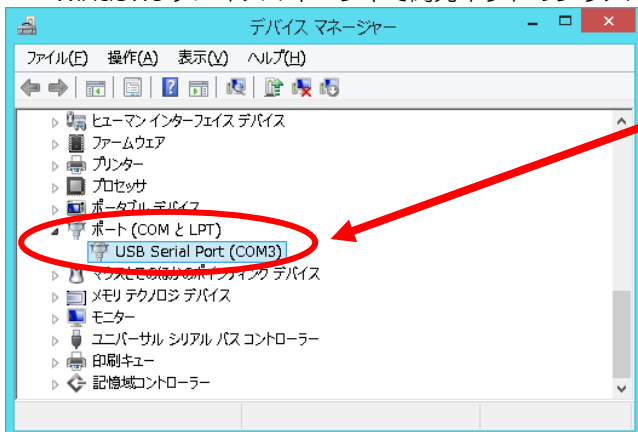
8. ESP8266 (ESP-WROOM-02) を設定します。

最初は下図のような設定になっていますので、各項目を修正します。(書き込み時に不具合が出た場合は戻してください)



9. シリアルポートを設定します。開発ボードとパソコンをUSBmicro-Bケーブルで接続してください。

Windows デバイスマネージャで開発キットのシリアルポート番号を確認し、メニューバーの「ツール」から設定します。

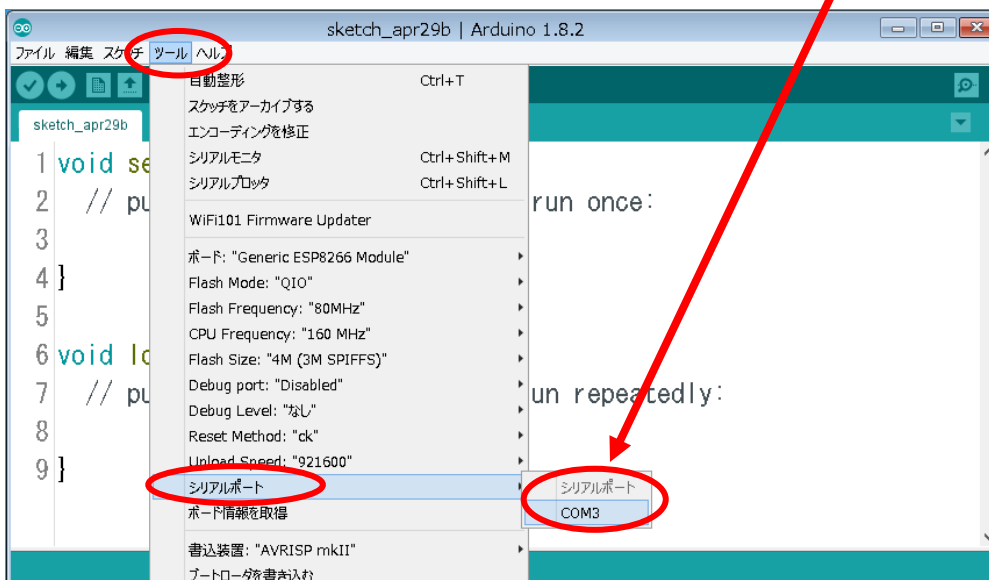


開発ボードがCOM*に設定されている事を確認
(この場合はCOM3)

ドライバがインストールされていない場合は、FTDI社のホームページからVCPドライバをインストールしてください。

<http://www.ftdichip.com/Drivers/VCP.htm>

同じCOM*に設定 (この場合はCOM3)
ご使用の環境により異なる場合があります。

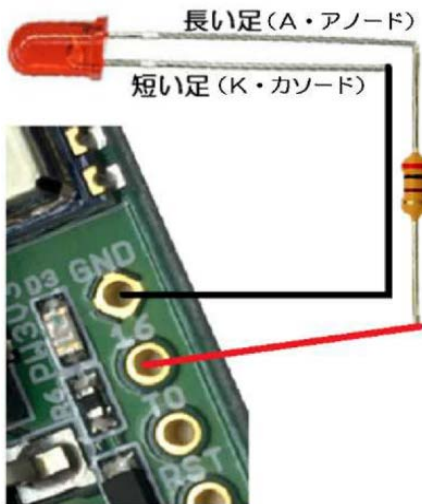


10. 実際に動作確認をしてみます。最初に動作確認用のLEDを接続します。

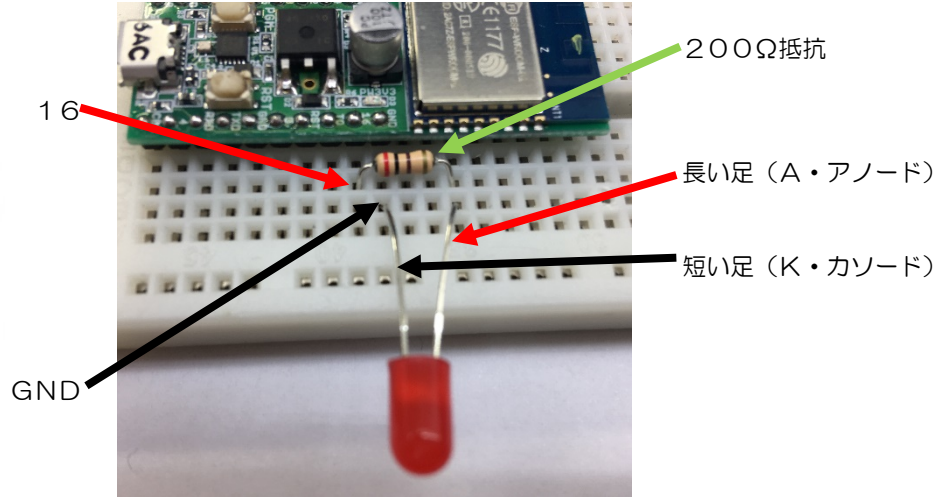
一旦開発キットからUSBケーブルを抜き、ブレッドボードなどを使用して下記のようにLEDを接続してください。適当なLED（この場合、赤色5mm）と200Ωの抵抗（1/4W、1/6Wどちらでも結構です）を使用します。16番ピン～200Ω抵抗～LED長い足、LED短い足～GNDと配線します。配線が終わったら、間違いが無いか確認してUSBケーブルでパソコンと接続します。

- ・赤色LED : OSDR5113A（弊社通販コード：I-11655）など
- ・200Ω抵抗 : CF25J200RB（弊社通販コード：R-25201）など
- ・ブレッドボード : EIC-301（弊社通販コード：P-00313）など

◎直接配線する例



◎ブレッドボードを使用した例



11. プログラムを書いてみる

LEDを点滅させるプログラムです。一旦全ての文字を削除して、下記の通り入力してください。

```
sketch_apr29a | Arduino 1.8.2
ファイル 編集 スケッチ ツール ヘルプ
sketch_apr29a
1 void setup()
2 {
3   pinMode(16, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(16, HIGH);
9   delay(500);
10  digitalWrite(16, LOW);
11  delay(200);
12 }
ボードへの書き込みが完了しました。
```

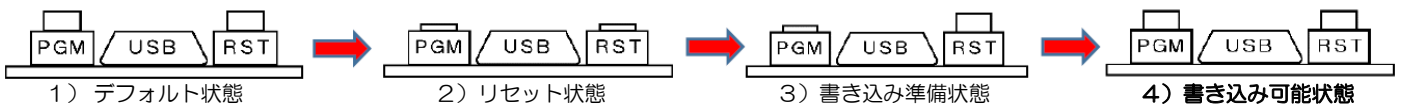
1 2. プログラム書き込みを可能な状態にする

開発キットとパソコンをUSBケーブルで接続し、シリアルポート番号を設定してください。

開発キットのスイッチを下図のように操作してください。

RSTスイッチとPGMスイッチ(右上図赤丸枠)を同時に押し、RST⇒PGMの順で離します。

(RSTスイッチを離れた時点でPGMスイッチが押されている状態にしてからPGMスイッチを離します)

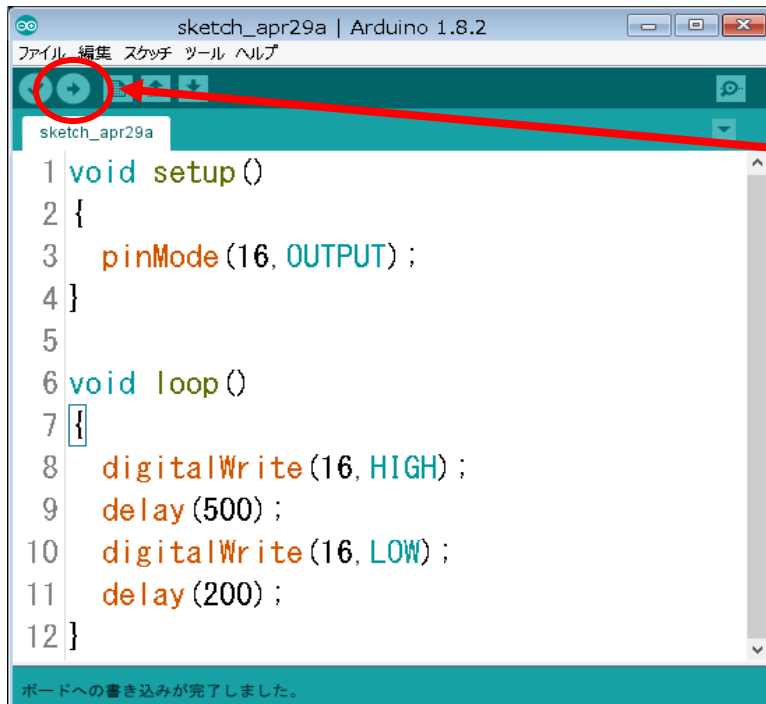


1 3. プログラムを転送・書き込みします。

下図赤丸印の⇒をクリックしてプログラムを書き込みます。

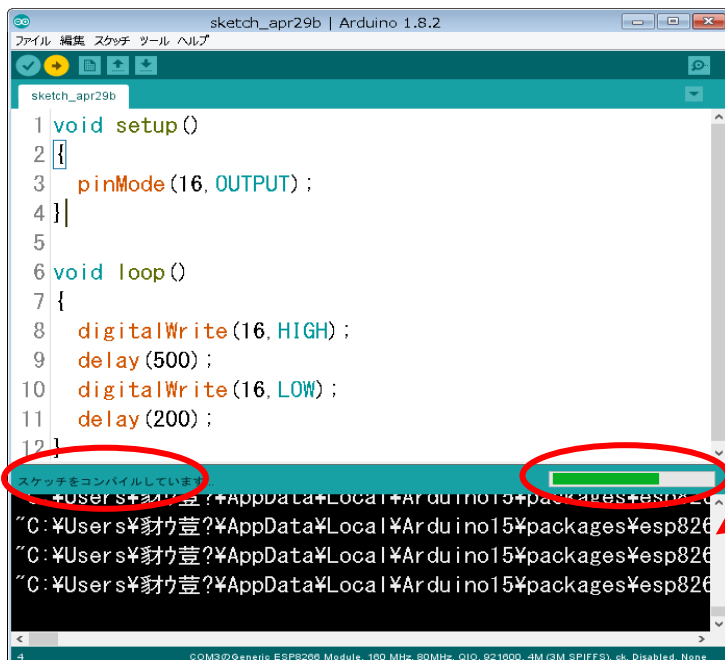
最初は「スケッチのフォルダの保存先」を聞いてきますので、適当な場所を指定します。

エラーがでて書き込めなかった場合は、再度前項「1 2. プログラムを書き込み可能な状態にする」の操作をおこない赤丸印の「⇒」をクリックしてください。



この「⇒」マークをクリックします。

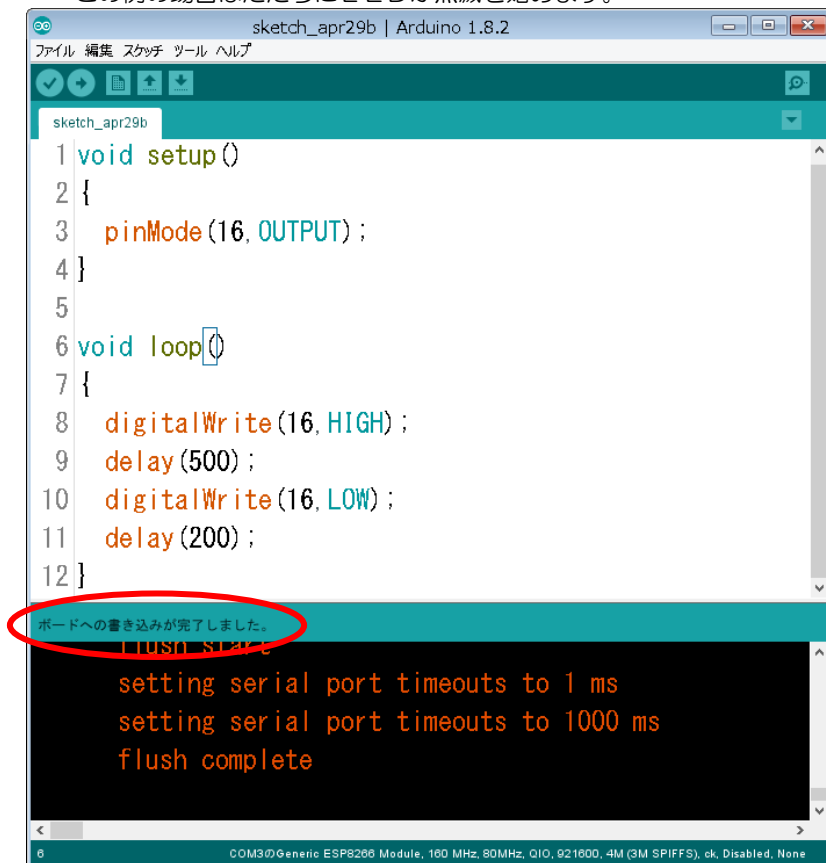
1 4. コンパイル中は下図赤丸印の「スケッチをコンパイルしています」のメッセージが表示され、緑色のプログレスバーが進行状況を表示します。



4. 「環境設定」で「より詳細な情報を表示する」にチェックを入れていた場合、このウィンドウに詳細な情報が表示されます。

17. 書き込み終了

書き込みが終わると「ボードへの書き込みが完了しました」のメッセージが表示されます。
この例の場合はただちにLEDが点滅を始めます。



The screenshot shows the Arduino IDE interface. The top window displays a sketch named 'sketch_apr29b' with the following code:

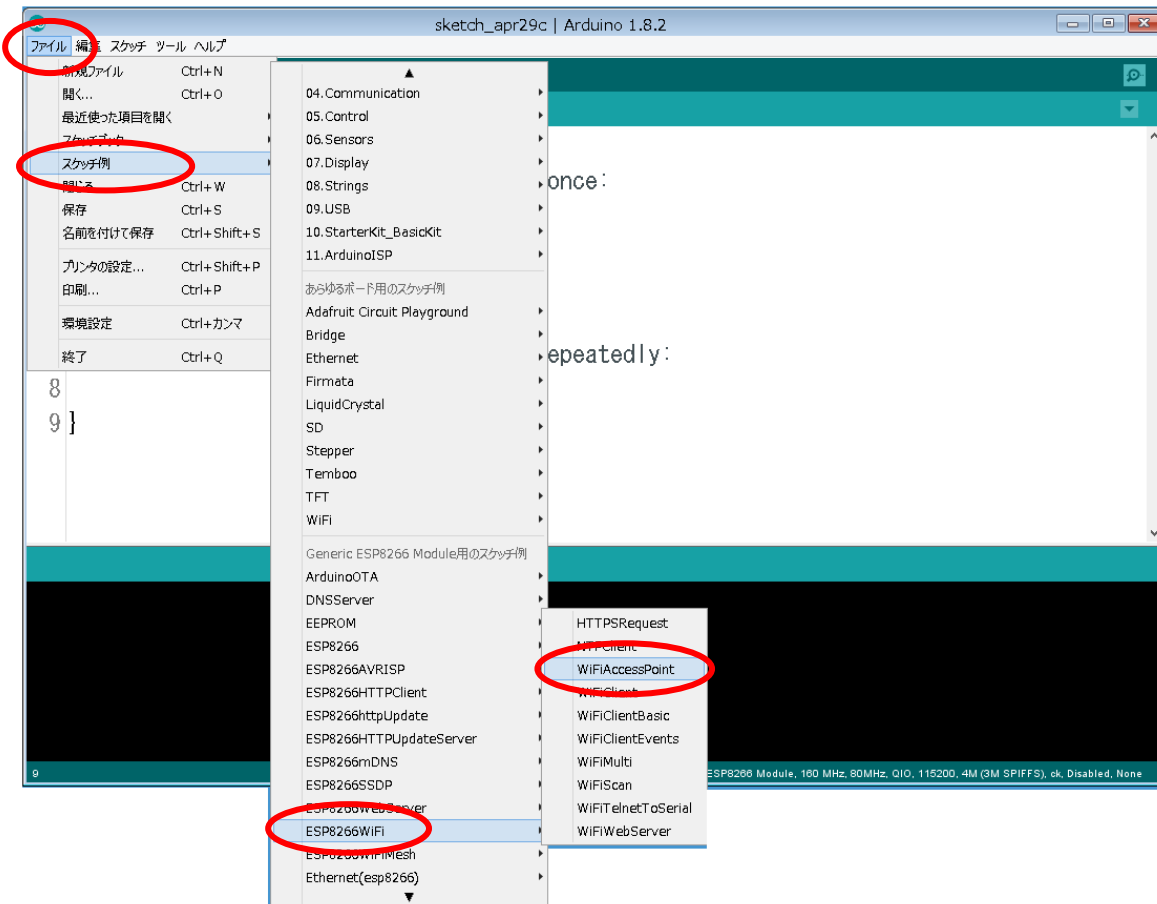
```
1 void setup()
2 {
3   pinMode(16, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(16, HIGH);
9   delay(500);
10  digitalWrite(16, LOW);
11  delay(200);
12 }
```

The bottom window shows the serial monitor output, which includes the message 'ボードへの書き込みが完了しました。' (Board upload complete) circled in red, followed by 'FLUSH SERIAL' and 'setting serial port timeouts to 1 ms', 'setting serial port timeouts to 1000 ms', and 'flush complete'. The status bar at the bottom indicates the board is a 'COM3のGeneric ESP8266 Module, 160 MHz, 80MHz, QIO, 921600, 4M (3M SPIFFS), ck, Disabled, None'.

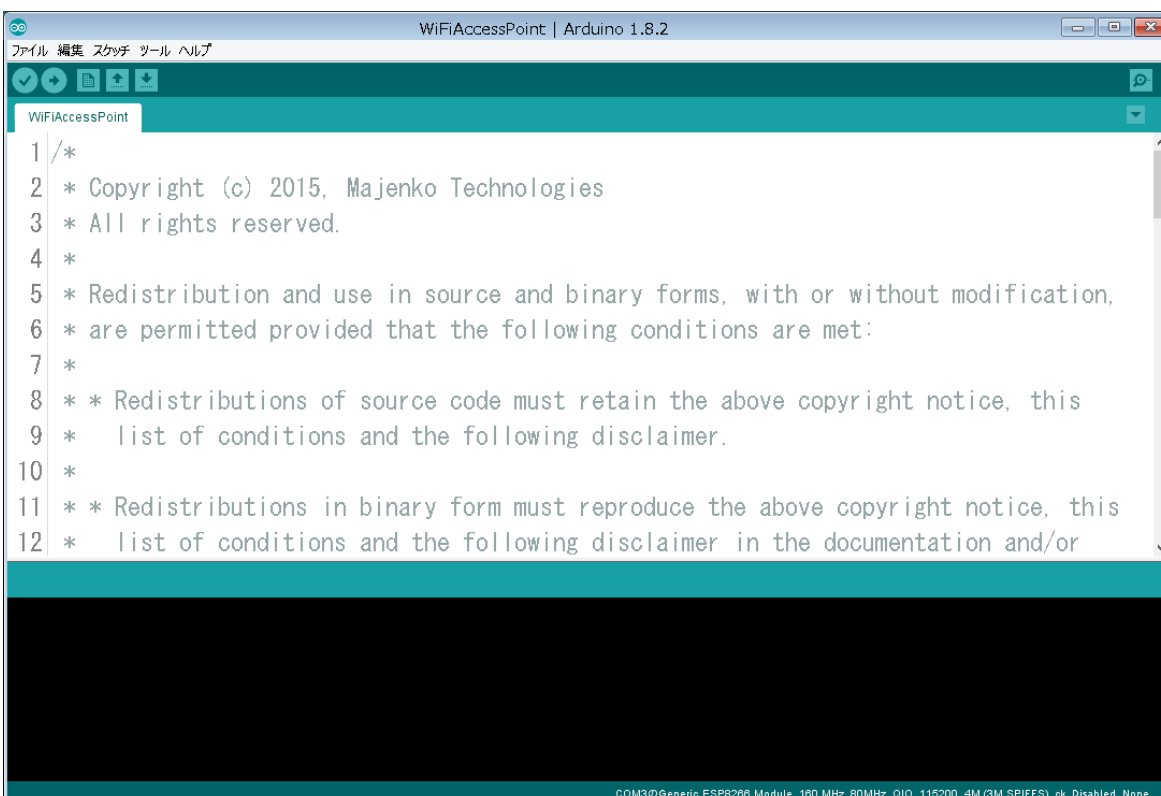
次ページからはW i - F i を使用した接続例です。

18. Wi-Fiでダイレクトに接続する

Wi-Fiルータを介さず、開発キットをアクセスポイントに設定し、スマートフォンなどから直接開発キットに接続します。メニューバーの「ファイル」⇒「スケッチの例」にマウスポインタを合わせるとスケッチの例が開きます。そのまま最下部の「▼」にマウスポインタを合わせると「ESP8266WiFi」が表示されますので、マウスポインタを合わせます。横に表示されたウィンドウの「WiFiAccessPoint」をクリックします。

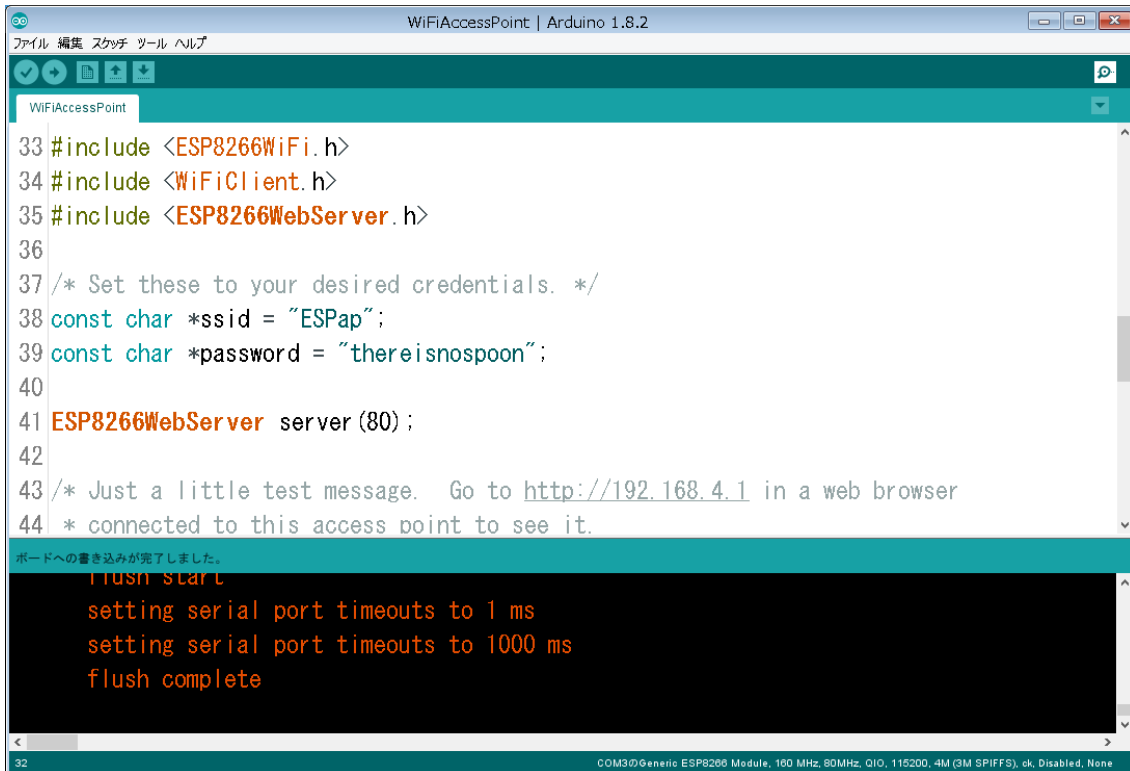


19. 新たにウィンドウが開きますのでそちらで作業します。



20、下にスクロールして、プログラム本体を確認してください。

下記画面以下がプログラム本体です。行番号表示の設定になっている場合は左端の33（33行目）からです。



```
WiFiAccessPoint | Arduino 1.8.2
ファイル 編集 スケッチ ツール ヘルプ
WiFiAccessPoint
33 #include <ESP8266WiFi.h>
34 #include <WiFiClient.h>
35 #include <ESP8266WebServer.h>
36
37 /* Set these to your desired credentials. */
38 const char *ssid = "ESPap";
39 const char *password = "thereisnospoon";
40
41 ESP8266WebServer server(80);
42
43 /* Just a little test message.  Go to http://192.168.4.1 in a web browser
44 * connected to this access point to see it.
ボードへの書き込みが完了しました。
Upload started
setting serial port timeouts to 1 ms
setting serial port timeouts to 1000 ms
flush complete
COM3のGeneric ESP8266 Module, 160 MHz, 80MHz, QIO, 115200, 4M (3M SPIFFS), ck, Disabled, None
```

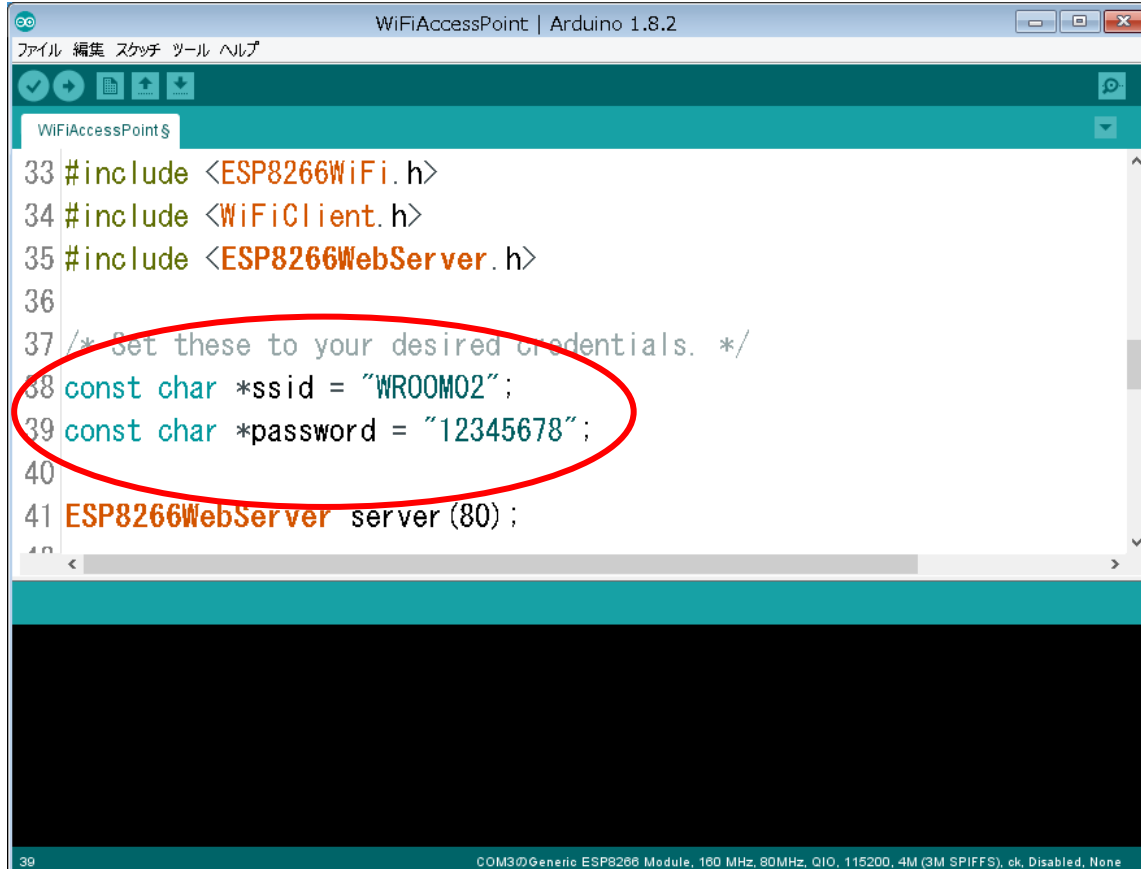
21、上図38行目、39行目を書き換えます。任意に設定可能ですが、例えば下記のように設定します。

ここでは ssid は判りやすいように、また password は後で入力しやすいように数字にしておきます。

```
const char *ssid = "WROOM02";
```

```
const char *password = "12345678";
```

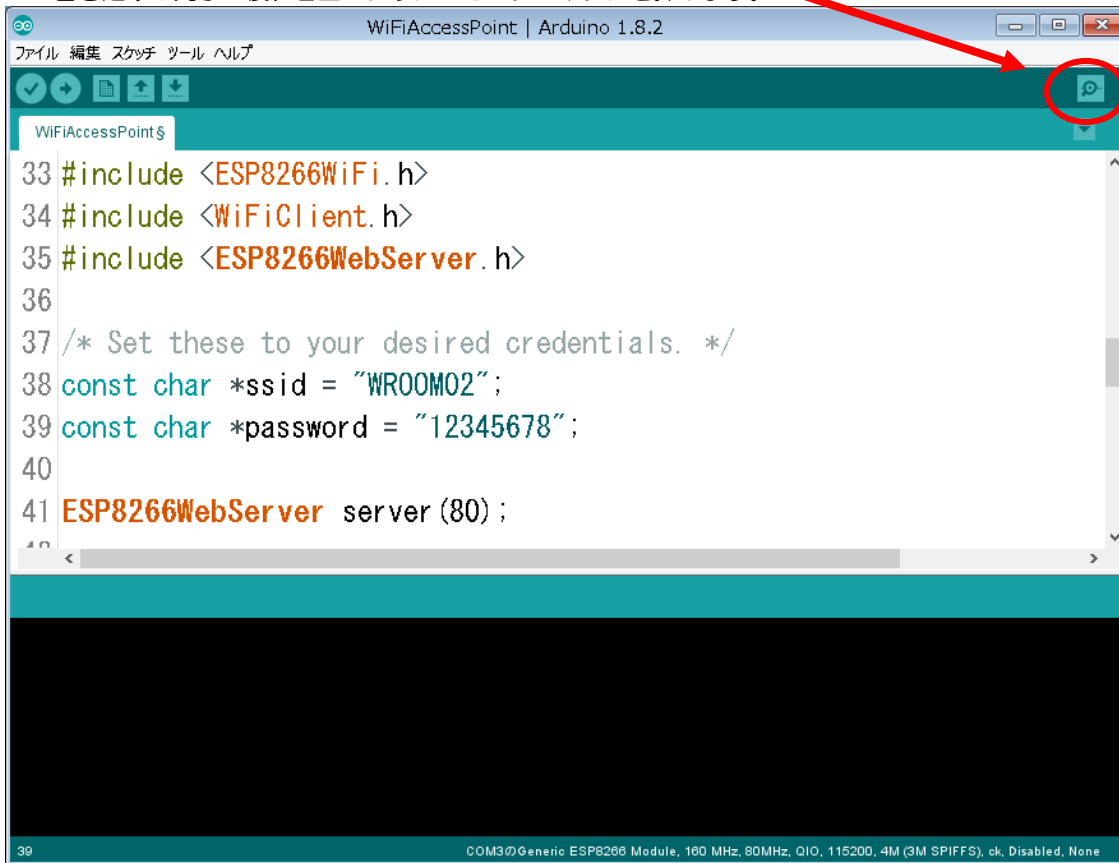
【8】 ページ「12。」から【11】 ページ「17。」までの手順でプログラムを書き込みます。



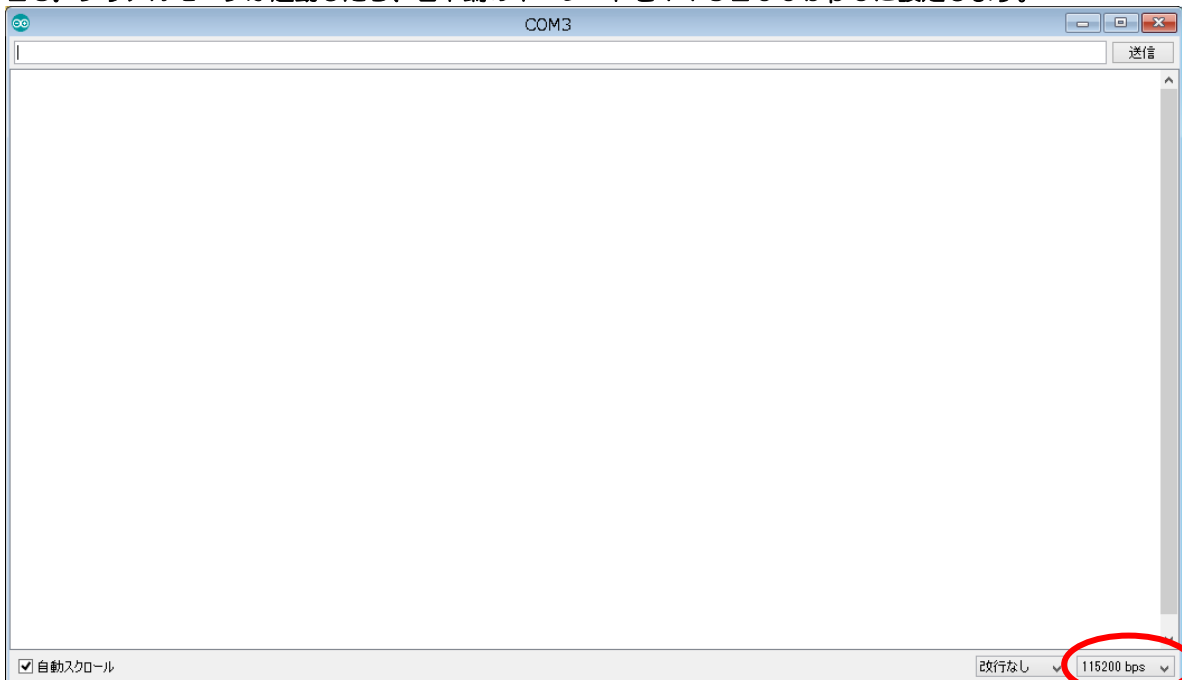
```
WiFiAccessPoint | Arduino 1.8.2
ファイル 編集 スケッチ ツール ヘルプ
WiFiAccessPoint$
33 #include <ESP8266WiFi.h>
34 #include <WiFiClient.h>
35 #include <ESP8266WebServer.h>
36
37 /* Set these to your desired credentials. */
38 const char *ssid = "WROOM02";
39 const char *password = "12345678";
40
41 ESP8266WebServer server(80);
42
COM3のGeneric ESP8266 Module, 160 MHz, 80MHz, QIO, 115200, 4M (3M SPIFFS), ck, Disabled, None
```

22. シリアルモニタを起動します。

書き込みが完了の後、右上のシリアルモニタのボタンを押します。

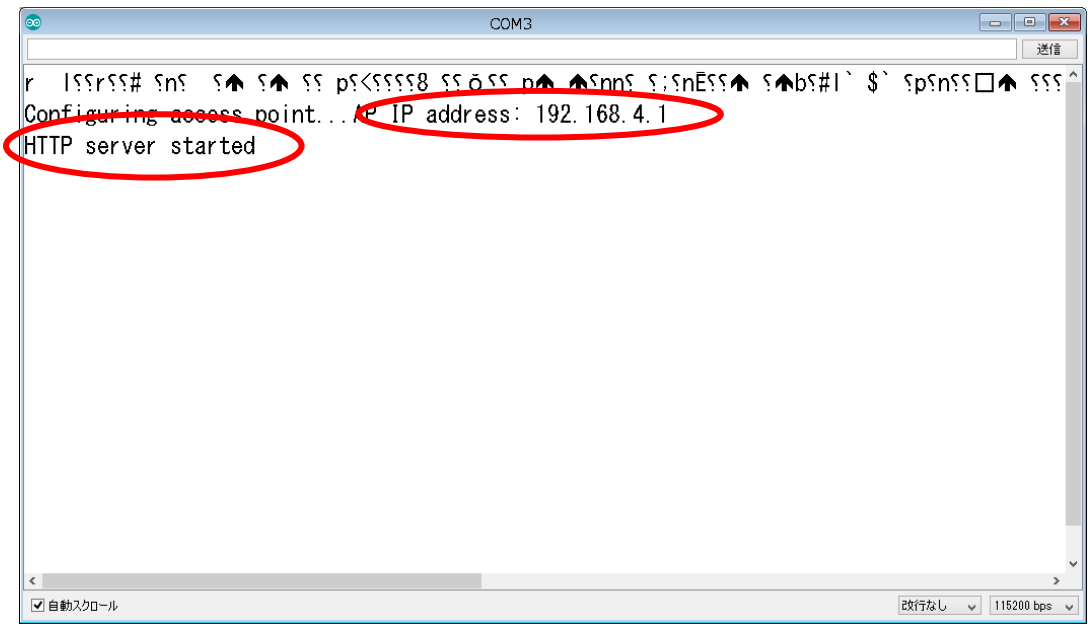


23. シリアルモニタが起動したら、右下端のボーレートを115200bpsに設定します。



115200bps に設定

24. 開発キットのRSTスイッチを押して、下記画面の赤丸内の表示になるのを確認してください。
何回かRSTスイッチを押してリセットする必要がある場合もあります。

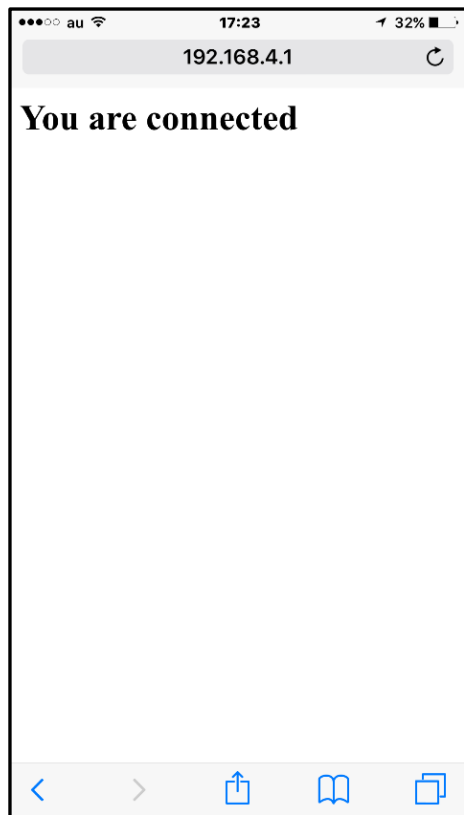


25. 手元のスマートフォンなどでWi-Fiの接続をします。

スマートフォンのWi-Fi設定画面を開きます。(各機種により異なりますので、画像は省略します)
プログラムの「const char *ssid = "WROOM02";」で設定したサーバー「WROOM02」を探し、
「const char *password = "12345678";」で設定したパスワード「12345678」を入力します。

26. スマートフォンなどのブラウザを開き、<http://192.168.4.1>を開きます。

「You are connected」と表示が出れば接続成功です。



詳しくはESP-WROOM-02、ESP8266のマニュアルをご覧ください。

(株)秋月電子通商