

ESP8266-SEED-R2
組み立てマニュアル

マイクロファン

<http://www.microfan.jp/>

<http://store.shopping.yahoo.co.jp/microfan/>

<http://www.microfan.jp/shop/>

2017 年 2 月

Copyright © 2015-2017 MicroFan,
All Rights Reserved.

目次

第 1 章	ESP8266-SEED-R2 の紹介	1
1.1	製品概要	1
1.1.1	IoT ノードの種	1
1.1.2	32 ビットプロセッサ	2
1.1.3	WiFi 無線ネットワーク装置	2
1.1.4	赤外線送受信装置	3
1.1.5	環境センサー	3
1.1.6	回路の試作と機能拡張	4
1.1.7	電源回路	4
1.2	購入・作成上の注意	5
1.2.1	表面実装部品のはんだ付け	5
1.2.2	プログラミング用ソフトウェア	6
1.2.3	PC の動作環境	6
1.3	マニュアルの記載内容に関して	6
第 2 章	部品一覧	8
2.1	部品表	8
2.2	ESP-WROOM-02	8
2.3	赤外線 LED 用電流制限抵抗 R7	9
2.4	赤外線レシーバー	10
2.5	照度センサー	11
第 3 章	作成手順	12
3.1	半田ごての状態の管理	12
3.2	表面実装部品の実装	13
3.2.1	ESP-WROOM-02 の実装	13
3.2.2	電圧レギュレータの実装	15
3.3	スルーホール部品の実装	15
3.3.1	抵抗 (R7 以外)	16
3.3.2	青色 LED、照度センサ	16

3.3.3	セラミックコンデンサ	16
3.3.4	タクトスイッチ	16
3.3.5	タンタルコンデンサ	17
3.3.6	トランジスタ	17
3.3.7	電解コンデンサ	17
3.3.8	抵抗 R7	17
3.3.9	赤外線 LED	17
3.3.10	赤外線レシーバー	18
3.3.11	DC コネクタ	18
3.3.12	ピンヘッダーのはんだ付け	18
3.4	組み立ての確認	18
第 4 章	Arduino スケッチ環境の整備	19
4.1	Arduino Core	19
4.2	ESP8266 用 Arduino 開発環境のインストール	19
4.2.1	Arduino IDE のバージョン確認	20
4.2.2	Boards Manager を使用する準備	20
4.2.3	Boards Manager を使用した ESP8266 用 Core のインストール	20
4.2.4	ボードとパラメーターの選択	21
4.3	USB インターフェースの接続	21
4.4	サンプルスケッチの実行	23
4.4.1	BLINK:LED の単純な点滅	24
4.4.2	FADE:PWM による LED の点滅	25
4.4.3	照度センサーの利用	26
4.4.4	赤外線 LED と赤外線レシーバーの利用	26
4.4.5	WiFi WebServer	27
4.4.6	BME280 の利用	28
第 5 章	資料	32
5.1	ESP8266-SEED-R2 の回路図	32
5.2	BOOT モードの選択	32
5.3	ハードウェア拡張用コネクタ	33
5.4	電源レギュレータの電流容量と電圧	34
第 6 章	購入および問い合わせ先	36
6.1	ご協力をお願い	36
6.2	販売：ネットショップ	36
6.3	製品情報	36
6.4	問い合わせ先	36

6.5	所在地	37
-----	---------------	----

表目次

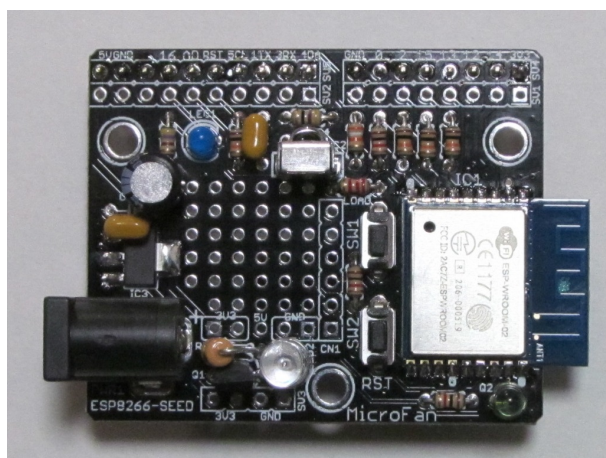
2.1	部品表	9
2.2	R7 の抵抗値と電流	10
4.1	USB インターフェースモジュールの接続	23
5.1	BOOT モード	33
5.2	SV1, SV4 ピン配置	34
5.3	SV2, SV5 ピン配置	34
5.4	CN1(BME280 モジュール) ピン配置	35

目次

1.1	赤外線レシーバと赤外線 LED	3
1.2	BME280 モジュール	3
1.3	BME280 モジュールの搭載例	4
1.4	ブレッドボードへの接続	5
2.1	部品：写真	8
2.2	ESP-WROOM-02	10
2.3	赤外線レシーバー	11
2.4	照度センサー	11
3.1	ESP-WROOM-02 の GND 端子	13
3.2	基板の GND 端子マーク	14
3.3	ESP-WROOM-02 の仮止めと仮はんだ付	14
3.4	ESP-WROOM-02 のはんだ付	15
4.1	Board Manager を利用するための設定	21
4.2	Boards Manager	22
4.3	ESP8266 の構成	22
4.4	USB インターフェースモジュールの例	23
4.5	BLINK:LED の単純な点滅	24
4.6	FADE:PWM による LED の点滅	25
4.7	照度センサー	26
4.8	IRrecvDemo	28
4.9	WiFi WebServer	29
4.10	BME280 モジュールの搭載例	30
4.11	ライブラリマネージャを利用した BME280 ライブラリの導入	30
5.1	ESP8266-SEED-R2 の回路図	32

第 1 章

ESP8266-SEED-R2 の紹介



1.1 製品概要

1.1.1 IoT ノードの種

近年様々なものをインターネットに接続してサービスの高度化を図るモノのインターネット「IoT^{*1}」が注目されており、IoT サービスを実現するための様々な開発や実験が、企業はもちろん個人でも行われています。

その IoT 装置を実現する中核部品として、ESP-WROOM-02 が注目されています。

ESP8266-SEED-R2 は 家庭用の IoT 機器の開発や実験の中核となる機能を ESP-WROOM-02 とともに^{*2}コンパクトな基板にまとめ、IoT 機器の試作や実験を行う人が、高機能な IoT 装置を手軽に作成するための中核（種）として利用できるよう設計されています。

ESP8266-SEED-R2 は以下のような特徴を持っています。

^{*1} Internet of Things

^{*2} 本キットでは ESP-WROOM-02 は別売りです。

- 高性能の 32 ビットマイクロプロセッサを搭載しており、Arduino UNO R3 などと比較して高速な処理が行えるとともに、大容量の FLASH と RAM を利用可能
- ネットワークと接続するための WiFi ネットワーク機能を搭載
- 周囲の家電機器を制御するための赤外線 LED を搭載
- 周囲の家電機器の操作状況をモニタできる赤外線レシーバーを搭載
- 装置の周辺の温度、湿度、明るさなどを取得できるセンサー（BME280:別売り）を搭載可能
- 基板の中央部に独自回路を搭載できるユニバーサルエリアを装備
- 使用者の自由な発想でハードウェア的に機能拡張できるように、ESP-WROOM-02 の信号線が基板上部の一列のピンヘッダーに引き出されており、ブレッドボードに挿して利用できます。
- 電子工作で広く利用されている Arduino などの無償、便利、高機能な開発環境を利用してソフトウェアを開発可能
- 800mA のレギュレータを搭載し、ESP-WROOM-02 に安定した電源を供給可能

1.1.2 32 ビットプロセッサ

ESP8266-SEED-R2 の中核となる ESP-WROOM-02 には、高機能な 32 ビットプロセッサと Arduino UNO R3 などと比較して大容量の Flash(プログラムメモリ) と RAM (データメモリ) が搭載されており、性能が要求されたり、ある程度のプログラムやデータ規模が要求される様々なアプリケーションに余裕をもって対応することができます。

ESP8266-SEED-R2 で利用できる代表的なプログラミング環境としては、下記の 2 種類があります。

- Arduino IDE
- NodeMCU

本文書の 4 章では、Arduino プログラミング環境のインストール法を説明しています。

1.1.3 WiFi 無線ネットワーク装置

ESP8266-SEED-R2 の中核となる ESP-WROOM-02 には、WiFi の無線ネットワーク機能が標準で組み込まれています。従来の MCU をネットワークに接続しようとすると、ネットワーク接続用の周辺機器を追加する必要がありましたが、ESP-WROOM-02 はそれのみでネットワークに接続することができます。

また、ESP-WROOM-02 は、WiFi のクライアントとしてアクセスポイントに接続したり、自分自身がアクセスポイントになってスマートフォンなどの接続を受け入れることができます。

このため、ESP8266-SEED-R2 は Arduino の入門者にはもちろんですが、特に、Arduino の基本機能を一通り習得し、無線機能を活用した応用に取り組みたい人に最適なボード

です。

1.1.4 赤外線送受信装置

家庭用の IoT 装置やホームオートメーション装置の一般的な機能として、家電機器をネットワークから制御するものがあります。テレビやエアコンに代表される家電機器は、赤外線リモコンで制御できるため、IoT 機器の基本的な機能として、リモコン用の赤外線の送受信機能の装備が強く望まれます。

ESP8266-SEED-R2 は図 1.1 に示すように、赤外線 LED と赤外線レシーバーを標準で装備しており、このような要求に簡単に応えることができます。

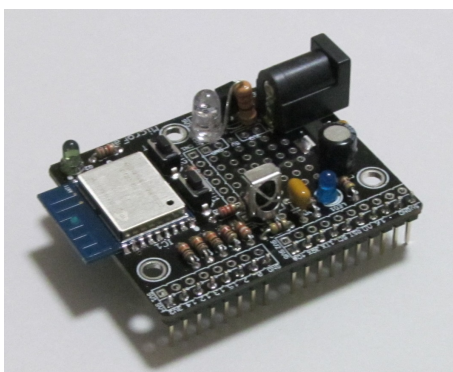


図 1.1 赤外線レシーバと赤外線 LED

1.1.5 環境センサー

IoT 装置で遠隔から家電装置を適切に制御するためには、IoT 装置が気温、湿度、気圧、明るさなどの部屋の環境情報を取得する必要があります。

ESP8266-SEED-R2 は照度センサーを標準で搭載しており、周囲の明るさを取得することができます。

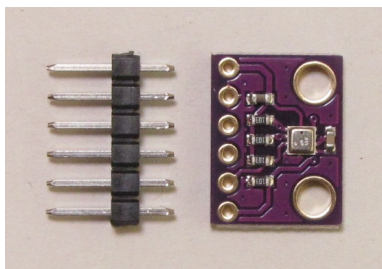


図 1.2 BME280 モジュール

さらに、図 1.2 に示す別売りの BME280 環境センサーモジュール*³を図 1.3 に示すように専用の接続端子に搭載することにより、温度、湿度、気圧を取得し、それを利用して家電機器の操作を行ったり、それらの環境情報をネットワークを通じて遠隔地の利用者に通知することができます。

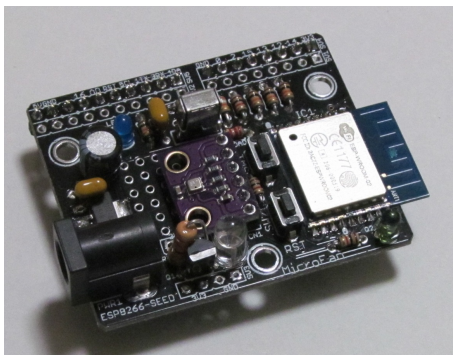


図 1.3 BME280 モジュールの搭載例

1.1.6 回路の試作と機能拡張

開発者ごとに試作や開発したい IoT 装置の機能は異なるため、画一的な一つの装置でそれらのすべてに対応することはできません。

ESP8266-SEED-R2 は、IoT システム開発者の広範囲な要求に応えるために、機能拡張のための 3 種類の方法を用意しています。

- 拡張回路の試作などの際に試行錯誤をする場合には、基板の端に用意されている端子部分にヘッダーピンをはんだ付けし、図 1.4 に示すように ESP8266-SEED-R2 をブレッドボードに接続して試作を行うことができます。
- 確定した小さな回路であれば、基板中央部のユニバーサルエリアに回路を作り込み、独自のコンパクトな IoT 装置を作成することができます。
- 確定した回路がユニバーサルエリアに入らない場合には、基板の端に用意されている端子部分にピンソケットをはんだ付けし、十分なサイズの拡張基板を接続して独自の高性能な IoT 装置を作成することができます。

1.1.7 電源回路

ESP-WROOM-02 は WiFi 機能を稼働させる際に、突入電流として多くの電流を消費することが知られています。このため、電源が貧弱だと、ESP-WROOM-02 の動作が不安定になることがあります。

*³ <http://store.shopping.yahoo.co.jp/microfan/gy-bme280.html>

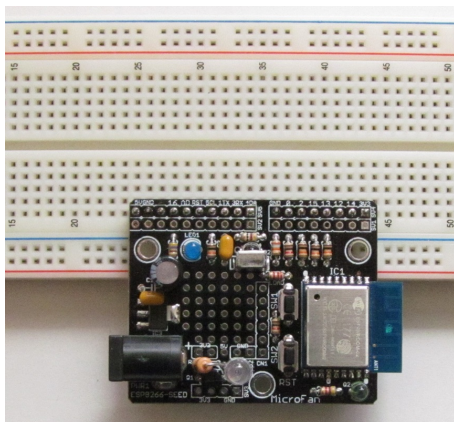


図 1.4 ブレッドボードへの接続

ESP8266-SEED-R2 は、AC アダプタを接続するための DC ジャックと 800mA 供給できる電源レギュレータを搭載し、ESP-WROOM-02 に十分な電力を供給して、安定して動作できるように配慮しています。

1.2 購入・作成上の注意

ESP8266-SEED-R2 をご購入の際には、下記項目をご確認ください。

- ESP-WROOM-02 は別売りです。
<http://store.shopping.yahoo.co.jp/microfan/esp-wroom-02.html>
- BME280 センサーモジュールは別売りです。
<http://store.shopping.yahoo.co.jp/microfan/gy-bme280.html>
- ピンフレームは別売りです。
- USB インターフェースが搭載されていないため、プログラミングなどを行うための PC との接続には別途 USB-シリアルインターフェースモジュールや接続ケーブルが必要になります。
<http://store.shopping.yahoo.co.jp/microfan/usb-uart-converter-cp2102-microusb.html>
<http://store.shopping.yahoo.co.jp/microfan/jumperwire-ff-20cm-10.html>

1.2.1 表面実装部品のはんだ付け

ESP8266-SEED-R2 は、多くの電子工作愛好者が容易に作成できるように、配置やはんだ付けが楽に行えるスルーホール部品を基本的に使っています。しかしながら、2 種類の部品 (ESP-WROOM-02, AMS1117) は、表面実装型の部品を使っています。表面

実装型の部品の配置とはんだ付けは必ずしも難しいものではありませんが、ちょっとしたノウハウと練度が求められます。

ESP8266-SEED-R2 の作成を行う前に、表面実装型の部品の練習を行っておくことをお勧めします。

1.2.2 プログラミング用ソフトウェア

ESP-WROOM-02 は標準的には AT コマンドを使用した無線 LAN 装置として構成されていますが、この文書では、Arduino IDE でプログラミングするための方法を示しています。なお、Arduino IDE を利用して ESP8266-SEED-R2 にスケッチを書き込む際に必要な USB-シリアルインターフェース機能は、本製品に含まれていないので、必要に応じて別途ご用意ください。

Arduino IDE はオープンソースソフトウェアとして開発と提供が行われており、機能の追加や改良が頻繁に行われているため、本文書で提供している情報は、ESP8266-SEED-R2 の購入者の利用時にはすでに古い情報になっている可能性があります。そのため、本文書で示している内容と異なる部分があったり、本文書で示している手順ではうまく動作しないことがあることと、その場合には、各自で対処方法を調査・確認していただく必要があることをご承知おきください。

1.2.3 PC の動作環境

現時点では、ESP8266-SEED-R2 用の Arduino IDE は、Windows 10 上のみで動作確認を行っております。このソフトウェアは Windows の他のバージョンや MacOS や Linux 上でも問題なく稼働する^{*4}と考えられますが、動作確認は行っていません。ESP8266-SEED-R2 を MacOS や Linux 上の Arduino IDE で利用される場合には、各自の責任において動作確認をお願いいたします。

1.3 マニュアルの記載内容に関して

本マニュアルの記載内容と、ご提供するソフトウェア、ハードウェアに差異がある場合には、ご指摘によりマニュアルの迅速な訂正を心がけますが、ご提供するソフトウェア、ハードウェアの現品の仕様が優先されます。

お伝えする内容と本質的な問題がない場合には、本マニュアルには、旧バージョンの製品の写真や他製品の写真などがそのまま使用されている場合がありますのでご承知おきください。

本書に記載されている内容に基づく作業、運用などにおいて、いかなる損害が生じても、弊社および著者をはじめとする本文書作成関連者は、一切の責任を負いません。

本文書に記載されている製品名などは、一般的にそれぞれの権利者の登録商標または商

^{*4} WEB 上に MacOS や Linux での ESP-WROOM-02/ESP8266 の利用例が多く報告されています。

標です。

第 2 章

部品一覧

2.1 部品表

ESP8266-SEED-R2 キットの部品表を表 2.1、キットの内容を図 2.1 に示します。部品が不足、破損している場合には、キットを組み立てる前にマイクロファンにお問い合わせください。

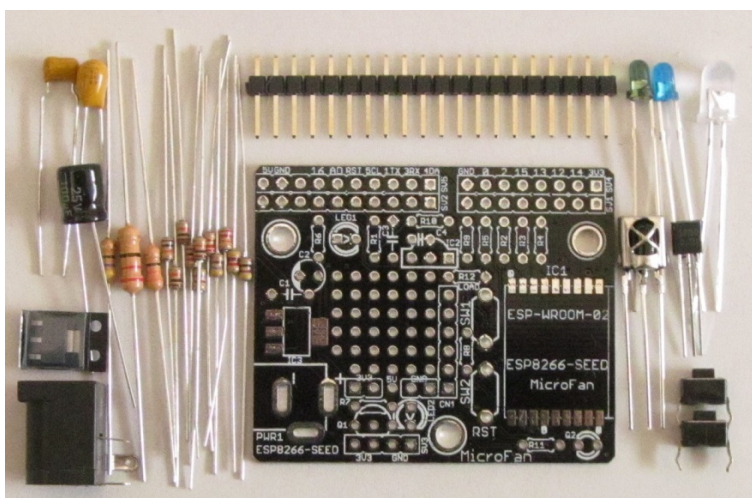


図 2.1 部品：写真

2.2 ESP-WROOM-02

ESP8266-SEED-R2 の中核となる ESP-WROOM-02^{*1}を図 2.2 に示します。

無線装置としての日本国内での利用に必要な技適を取得しており、安心して利用することができます。ESP-WROOM-02 の製造元の WEB ページは以下の通りです。

<http://espressif.com/en/products/hardware/esp-wroom-02/overview>

^{*1} ESP-WROOM-02 はキットには含まれていません。

表 2.1 部品表

部品	シンボル	規格等	個数
プリント基板	ESP8266-SEED	Rev.2	1
IC	IC1	ESP-WROOM-02	別売
	IC2	VS1838B	1
	IC3	AMS1117-3.3	1
トランジスタ	Q1	S8050	1
照度センサー	Q2	NJL7502L (3mm 緑灰色)	1
LED	LED1	3mm 青色	1
赤外線 LED	LED2	5mm 透明	1
抵抗	R1-R4	10K Ω	4
	R5	1K Ω	1
	R6, R8	470 Ω	2
	R7	22 Ω -1W, 33 Ω -1/2W, 47 Ω -1/2W	各 1
	R9	22K Ω	1
	R10	47 Ω	1
	R11	4.7K Ω	1
	R12	2.2K Ω	1
セラミックコンデンサ	C1	10 μ F	1
	C4	-	未実装
電解コンデンサ	C2	100 μ F	1
タンタルコンデンサ	C3	22 μ F	1
タクトスイッチ	SW1, SW2	2 端子	2
環境センサー	CN1	GY-BME280	別売
DC ジャック	PWR1		1
ピンソケット	SV1	1x8	別売
ピンソケット	SV2	1x10	別売
ピンソケット	SV3	1x4	別売
ピンヘッダ	SV4, SV5	1 列 1x20PIN	1

2.3 赤外線 LED 用電流制限抵抗 R7

赤外線 LED の発光強度を制御する R7 は、状況に応じて選択できるよう 3 種類用意しています。

制御する家電機器と赤外線 LED との距離や向きの制約を少なくしようとすると、赤外線 LED の発光強度を上げることが求められます。赤外線 LED の発光強度は、赤外



図 2.2 ESP-WROOM-02

線 LED に流す電流を増加させることによりあげることができます。一方、ESP8266-SEED-R2 で使用している赤外線 LED は、継続的に電流を流す場合には 100mA 以下、ピークではデューティー比が 1/100 の場合には 1000mA 以下で使うことが求められています。

赤外線 LED は 38KHz で明滅（電流の ON, OFF）させてリモコン信号を発信します。平均電流の上限が 100mA が上限だとすると、デューティー比が 1/2 と仮定した場合、赤外線 LED に流す電流値の最大値を 200mA 以下に抑えればよいことになります。

赤外線 LED の V_f とその駆動用のトランジスタの V_{ce} の和が 1.5V 程度とすると、R7 には 3.5V 程度の電圧がかかることになります。その時の R7 の抵抗値と、R7、したがって赤外線 LED に流れる電流の大きさを表 2.2 に示します。

表 2.2 R7 の抵抗値と電流

抵抗値	電流	電力
22 Ω	160mA	0.84W
33 Ω	106mA	0.37W
47 Ω	75mA	0.26W

赤外線 LED の利用やプログラミングに慣れていない場合には、プログラミングミスなどで 100mA 以上の電流が赤外線 LED に継続的に流れて壊れないように、R7 に 47 Ω を使用してください。

赤外線 LED の利用やプログラミングに慣れている場合には、赤外線 LED の発光強度を上げたい場合には、R7 に 22 Ω を使用してください。

2.4 赤外線レシーバー

ESP8266-SEED-R2 では、リモコン用赤外線信号を受信する専用のレシーバーを搭載しています。赤外線レシーバーを図 2.3 に示します。

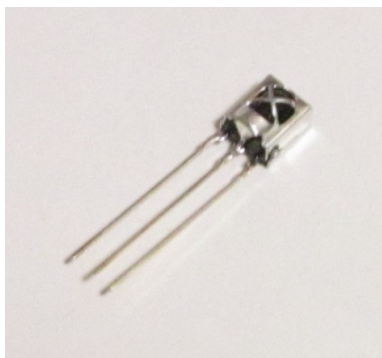


図 2.3 赤外線レシーバー

2.5 照度センサー

ESP8266-SEED-R2 では、周囲の明るさを取得するために、照度センサー（フォトリンジスタ）を搭載しています。照度センサーを図 2.4 に示します。

LED のように見えますが、入力が電流ではなく光になっている一種のトランジスタ（フォトリンジスタ）です。

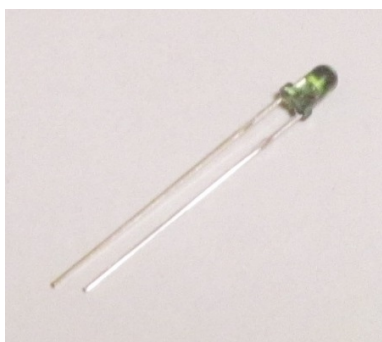


図 2.4 照度センサー

第3章

作成手順

ESP8266-SEED-R2 キットの標準的な作成手順の概要を以下に示します。基本的には、一般的な電子工作の手順と同様で、背の低い部品から実装していきます。

1. 表面実装部品の実装
ESP-WROOM-02, AMS1117
2. 抵抗の実装（R7 以外）
3. 青色 LED、照度センサの実装
4. セラミックコンデンサの実装
5. タクトスイッチの実装
6. タンタルコンデンサの実装
7. トランジスタの実装
8. 電解コンデンサの実装
9. 抵抗 R7 の実装
10. 赤外線 LED の実装
11. 赤外線レシーバーの実装
12. DC ジャックの実装
13. ピンヘッダーの実装
14. 全体の動作確認

実際の作成作業に入る前に、一度作成手順全体に目を通しておくことをお勧めします。

3.1 半田ごての状態の管理

まず最初に、はんだ付けを行う際の、一般的な半田ごての状態の管理に関して示します。

はんだ付けを行う直前に、スポンジなどのこて先クリーナーで半田ごてをクリーニングしてフラックスや酸化膜などの汚れを取り除き、こて先が銀色に輝く状態ではんだ付けを行います。また、こて先にほとんどはんだが乗っておらず乾いていると、こて先から IC のピンや基板のパターンなどに熱が伝わりにくいので、こて先に少し（薄く）はんだを付けてこて先がはんだで濡れた状態にしてはんだ付けを行います。

3.2 表面実装部品の実装

最初に、プリント基板に表面実装部品 (ESP-WROOM-02, CH340G) を実装します。表面実装部品は、不慣れだと実装に苦労しますが、ちょっとしたコツを知っておくとそれほど苦労せずにきれいに実装することができます。

3.2.1 ESP-WROOM-02 の実装

ESP8266-SEED-R2 キットの基本構成には ESP-WROOM-02 は含まれていないので別途ご購入ください。

グランド端子への配慮

ESP-WROOM-02 はグランド端子^{*1}の熱容量が大きい^{*2}ようで、はんだ付けの際に、はんだが ESP-WROOM-02 の端子になじまず付きにくい場合がありますので注意が必要です。そのような場合には、端子に少し長めに半田ごてを当てて、はんだがしっかりと端子に乗るように配慮してください。また、グランド端子をはんだ付けすると ESP-WROOM-02 がかなり加熱されるため、過熱による損傷を防止するために複数の端子を連続的にはんだ付けせず熱を冷ますなどの配慮を行ってください。

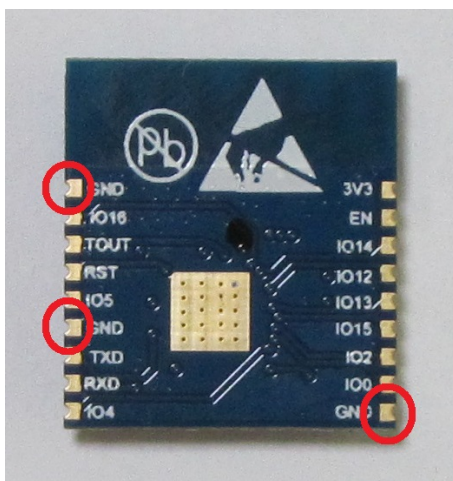


図 3.1 ESP-WROOM-02 の GND 端子

はんだ付けの際に、GND 端子に注意を払いやすいように、ESP8266-SEED-R2 の基板の ESP-WROOM-02 の GND 端子部分には、図 3.2 の赤丸に示すように、白いマーキングが施されています。

^{*1} ESP-WROOM-02 チップの 9, 13, 18 ピン

^{*2} 端子に半田ごてを当てても端子の温度が上がるのに少し時間がかかります。

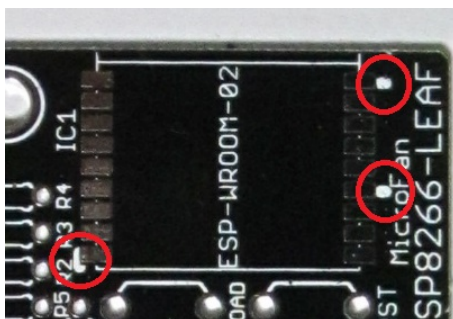


図 3.2 基板の GND 端子マーク

フラックス入りのはんだを利用する場合には、端子等に別途フラックスを塗布する必要はありませんが、端子（特にグランド）にはんだが付きにくい場合には、適宜フラックスを端子に塗布してからはんだ付けを行うとよいでしょう。

ESP-WROOM-02 の仮止めと位置調整

まず、ESP-WROOM-02 の端子とプリント基板の端子パターンの位置がずれないように注意して ESP-WROOM-02 を基板上に配置し、テープなどを貼って仮止めします。



図 3.3 ESP-WROOM-02 の仮止めと仮はんだ付

次に ESP-WROOM-02 の端の 1 つの端子をはんだ付けして IC を基板に仮止めします。この際、GND 端子をはんだ付しようとする、はんだ付けに手間取る可能性がある、GND 端子以外を選んでのはんだ付けするとよいでしょう。

ESP-WROOM-02 の取り付け位置がずれていないか確認し、ずれがあるようであれば、仮止めした部分を半田ごてで再加熱して溶かして取り付け位置を修正します。

さらに、すでに仮止めしている端子と異なる列の ESP-WROOM-02 の 1 つの端子をはんだ付けします。ここでも、GND 端子以外を選んでのはんだ付けをするとよいでしょう。

再度 ESP-WROOM-02 の取り付け位置がずれていないか確認し、ずれがあるようであれば、仮止めした部分を半田ごてで再加熱して溶かして取り付け位置を修正します。

ESP-WROOM-02 の取り付け位置が問題ない様であれば、図 3.4 に示すようにすべて

の端子のはんだ付けを行います。

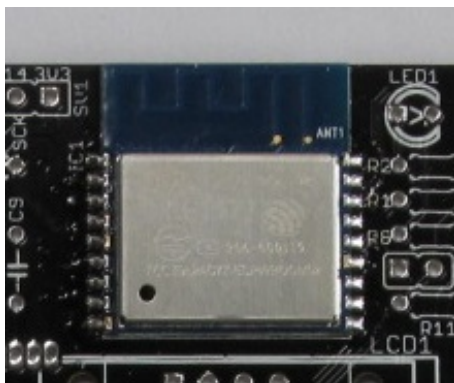


図 3.4 ESP-WROOM-02 のはんだ付

3.2.2 電圧レギュレータの実装

ASM1117 をプリント基板の取り付け位置に仮置きし、ASM1117 の向きに注意して位置決めをします。ASM1117 の端子とプリント基板端子のパターンの位置がずれないように注意して、ASM1117 をテープなどで基板に仮止めします。

ASM1117 の 1 つの端子をはんだ付けして IC を基板に仮止めします。

この状態で、ASM1117 の位置がずれていないか確認し、ずれているようであれば半田ごてを端子に当ててはんだを融かし、ASM1117 の位置調整を行います。

位置調整ができれば仮止めのテープを外し、ASM1117 の他の端子をはんだ付けします。

3.3 スルーホール部品の実装

以下の手順で残りのスルーホール部品を実装します。

1. 抵抗 (R7 以外)
2. 青色 LED、照度センサ
3. セラミックコンデンサ
4. タクトスイッチ
5. タンタルコンデンサ
6. トランジスタ
7. 電解コンデンサ
8. 抵抗 R7
9. 赤外線 LED
10. 赤外線レシーバー
11. DC ジャック
12. ピンヘッダー

13. 全体の動作確認

抵抗、ダイオード、コンデンサ、発光ダイオードなどの2本足の部品のはんだ付けは、まず片方の足をはんだ付けし、部品の取り付け姿勢などを必要に応じて修正してからもう片方の足をはんだ付けすると、部品の配置をきれいに整えて取り付けることができます。はんだ付け後に、基板裏面から伸びて余っている足をニッパーなどで切り取ります。

ピンソケット、ピンヘッダー、ICなどの複数の端子を持つ部品のはんだ付けは、端子の端の1ピン、もしくは両端が対角上の2ピンをはんだ付けし、部品の取り付け姿勢などを必要に応じて修正してから残りの端子をはんだ付けすると、部品の姿勢をきれいに整えて取り付けることができます。

ESP8266-SEED-R2のプリント基板はベタアースになっており、熱容量が大きくなっております。このため、各部品のGND端子をはんだ付けする際には基板の端子部分（ランド）の温度が上がりはんだが融けるまで時間がかかるため、他の端子と比較して少し長めにはんだごてを当てておく必要がありますのでご注意ください。

3.3.1 抵抗（R7 以外）

抵抗をプリント基板上の所定の位置に取り付けはんだ付けします。抵抗は数種類の抵抗値があるため、部品表 2.1 で確認して取り付けてください。

3.3.2 青色 LED、照度センサ

青色 LED, LED1 と照度センサ Q2 を所定の位置に取り付けはんだ付けします。

青色 LED の端子には極性があり、長い脚のほうがアノード (+)、短い脚のほうのカソード (-) で、カソード側のプラスチックモールドに切り欠きがあります。照度センサの端子にも極性があり、長い脚のほうがコレクタ (+)、短い脚のほうのエミッタ (-) で、カソード側のプラスチックモールドに切り欠きがあります。その切り欠きを基板のシルク印刷の切り欠きに合わせて取り付けはんだづけします。

3.3.3 セラミックコンデンサ

セラミックコンデンサ C1 を所定の位置に取り付けはんだ付けします。

3.3.4 タクトスイッチ

初めてタクトスイッチの取り付けを行う際には、タクトスイッチの足がそのままではプリント基板に刺さりにくいように思えますが、タクトスイッチの2本の足を基板の穴にあてがった状態で、タクトスイッチを基板に少し強く押し込むと、タクトスイッチの足が基板の穴に入って固定されます。その後、それぞれの足をはんだ付けします。

3.3.5 タンタルコンデンサ

タンタルコンデンサ C3 をプリント基板に取り付けはんだ付けします。

タンタルコンデンサの端子には極性があり、足の長いほうが+です。タンタルコンデンサは、C3 の取り付け穴の間隔に合うように、本体に大きな力が加わらないように注意してラジオペンチなどで事前に足の幅を広げて基板に取り付けてください。足の短い側を SV3 コネクタ側、足の長い側をユニバーサルエリア側の取り付け穴に取り付けるようにしてください。

3.3.6 トランジスタ

トランジスタ Q1 をプリント基板に取り付けはんだ付けします。

トランジスタ Q1 は、取り付け穴の間隔に合うように、本体に大きな力が加わらないように注意してラジオペンチなどで事前に足の幅を広げて基板に取り付けてください。トランジスタ本体の切りかけ部分とシルクの切りかけ部分の方向を合わせて取り付けてください。

3.3.7 電解コンデンサ

電解コンデンサをプリント基板に取り付けはんだ付けします。

電解コンデンサの端子には極性があり、本体の白い帯と基板のシルクの白塗り部分が合致するように取り付けてください。

電解コンデンサと周囲の部品の配置が密で取り付けにくい場合、セラミックコンデンサ C1 を電源レギュレータ側に傾けるなど、電解コンデンサが所定の位置に収まるよう調整をして取り付けてください。

3.3.8 抵抗 R7

抵抗をプリント基板上の所定の位置に取り付けはんだ付けします。

R7 に取り付ける抵抗の選択は、節 2.3 を参照してください。

抵抗 R7 は他の抵抗に比べてワット数が大きく基板に水平に取り付けられないので、基板に立てて取り付けます。取り付けの際には、抵抗の本体が DC ジャック側、抵抗のワイヤーが赤外線 LED 側になるように取り付けてください。

はんだ付けや後処理は、他の抵抗のはんだ付けに準じます。

3.3.9 赤外線 LED

赤外線 LED, LED2 を所定の位置に取り付けはんだ付けします。

赤外線 LED の本体のモールド部分は、必要性に応じて向きを調整して取り付けてくだ

さい。

赤外線 LED の端子には極性があり、長い脚のほうがアノード (+)、短い脚のほうのカソード (-) で、カソード側のプラスチックモールドに切り欠きがあります。その切り欠きを基板のシルク印刷の切り欠きに合わせて基板に取り付けはんだ付けします。

はんだ付けや後処理は、抵抗のはんだ付けに準じます。

3.3.10 赤外線レシーバー

赤外線レシーバー IC2 を、シルクの図に合わせて取り付けはんだ付けします。

赤外線レシーバー IC2 を基板に標準的な姿勢で取り付けると、SV2 にピンソケットを取り付けた場合に、ピンソケットが赤外線レシーバー IC2 の受光窓をふさいでしまいます。したがって、SV2 にピンソケットを取り付ける場合には、赤外線レシーバーの足を長めにして基板からある程度浮かせて取り付けるか、赤外線レシーバーの足を受光窓と反対側に L 型に曲げたうえで基板に取り付け^{*3}、受光窓を基板の上側に向けて基板に取り付けるなどの対応をとってください。

はんだ付けや後処理は、抵抗のはんだ付けに準じます。

3.3.11 DC コネクタ

DC コネクタをプリント基板に取り付けはんだ付けします。

3.3.12 ピンヘッダーのはんだ付け

ESP8266-SEED-R2 をブレッドボードで利用する場合には、必要に応じて、SV4, SV5 (あるいは SV1, SV2) に 10, ピンと 8 ピンのピンヘッダーを取りつけます。

ピンヘッダーはブレッドボードに挿せるように、SV4, SV5 (あるいは SV1, SV2) の基板の裏側に取り付けます。

ピンヘッダーは基板裏面からピンの短いほうをプリント基板に差し込み、プリント基板の表面ではんだ付けします。ピンヘッダーは、まず両端の一つのピンずつをはんだ付けし、ピンヘッダーの傾きを確認、修正した後に、残りのピン全部をはんだ付けします。

3.4 組み立ての確認

基板の電源ラインがショートしていないか確認します。

目視で確認を行うだけでなく、テスターでの確認を推奨します。テスターでは、GND と 5 V, GND と 3.3V の電源ライン間の抵抗を確認します。抵抗がほぼ 0 の場合には、電源ラインがショートしていることになるので、ショート場所の確認と修正が必要になります。

^{*3} 赤外線レシーバーの本体がユニバーサルエリアに乗ることになります。

第 4 章

Arduino スケッチ環境の整備

4.1 Arduino Core

ESP8266-SEED-R2 のプログラム開発環境として、広く利用されている Arduino IDE を利用することができます。

標準的な Arduino ボード Arduino UNO R3 は MCU として ATmega328P を搭載しており、スケッチの作成環境である Arduino IDE には、ATmega328P 用のコンパイラ等が組み込まれています。Arduino には UNO 以外にも複数のボードがあり、それぞれに異なる MCU が搭載されています。Arduino IDE では、このような個別の MCU 用のコンパイラ等を [Core] と呼んでおり、ATmega328P 以外の MCU 用の Core をインストールできるようになっています。

Arduino IDE を利用して ESP-WROOM-02 用のスケッチを開発するためには、Arduino IDE に ESP8266 用の Core をインストールする必要があります。Core のインストール法は、下記の WEB ページに示されています。

<https://www.arduino.cc/en/Guide/Cores>

Installing additional Arduino Cores

この手順に基づいた ESP8266 用 Core のインストール法を以下に示します。

4.2 ESP8266 用 Arduino 開発環境のインストール

ESP8266 用の Core は ESP8266 community によって開発され、以下の WEB ページで LGPL ライセンスで公開^{*1}されています。

<https://github.com/esp8266/Arduino>

^{*1} 高機能で利便性の高い Core を LGPL ライセンスで公開されている ESP8266 community に感謝いたします。

4.2.1 Arduino IDE のバージョン確認

なお、このドキュメントの作成時点 (2017/2) では、すでに Arduino IDE の最新版は 1.8.1 となっていますが、ESP8266 用の Core をインストールする Arduino IDE のバージョンは 1.6.8 を選択してください。

Arduino IDE 1.6.8 はこちらからダウンロードできます。

<https://www.arduino.cc/en/Main/OldSoftwareReleases>

なお、ESP8266 用 Core は改良が続けられており、対象となる Arduino IDE のバージョンも今後変わっていくものと考えられます。したがって、Arduino IDE の対象バージョンは ESP8266 用コアのページ [Arduino core for ESP8266 WiFi chip] を参照して確認してください。

<https://github.com/esp8266/Arduino>

4.2.2 Boards Manager を使用する準備

Arduino IDE に新しい Core をインストールするには、Board Manager を利用します。

Arduino IDE のメニューバーから、[ファイル] ⇒ [環境設定] を選択します。すると、図 4.1 に示すようなダイアログが表示されます。

ダイアログの下方の [Additional Boards Manager URLs] の入力枠に下記の URL を入力し、ダイアログの右下の [OK] ボタンを押してダイアログを閉じます。

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

これで、Boards Manager を利用して ESP8266 用 Core をインストールする準備ができました。

4.2.3 Boards Manager を使用した ESP8266 用 Core のインストール

Arduino IDE のメニューバーから、[ツール] ⇒ [マイコンボード : ...] ⇒ [ボードマネージャ...] を選択します。すると、図 4.2 に示すようなダイアログが表示されます。

このダイアログのリストの中から、左上に [esp8266 by ESP8266 Community] と表示されている項目をクリックして選択します。項目を選択するとその項目の右下に Install ボタンが表示されるので、それを押します。

これにより ESP8266 用コアのインストールが開始され、インストールの進行状況を示すプログレスバーが表示されます。

インストールが完了したら、ダイアログ右下の [閉じる] ボタンを押してダイアログを閉じます。

ESP8266 用のコアがインストールされると、Arduino IDE のメニューバーから、[ツール] ⇒ [マイコンボード : ...] を選択します。すると、選択肢として開発対象のボードのリ

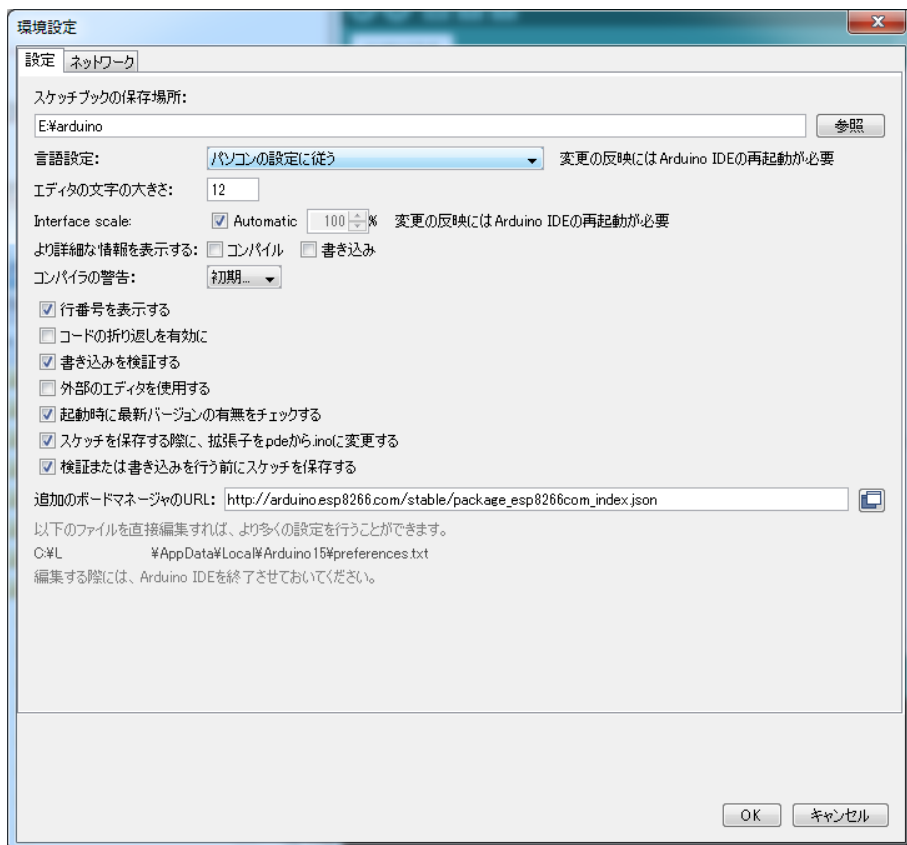


図 4.1 Board Manager を利用するための設定

ストが表示されます。そのリストの下の方に、複数の ESP8266 ボードの選択肢が表示され、ESP8266 用コアがインストールされていることが確認できます。

4.2.4 ボードとパラメーターの選択

メニューから [ツール] ⇒ [マイコンボード:...] を選択します。すると、選択肢として開発対象のボードのリストが表示されます。そのリストの下の方から、[Generic ESP8266 Module] を選択します。

選択後、再度 [ツール] を選択すると、ESP8266 の構成を選択するための項目が追加されているのが確認できます。これらの項目の値を、図 4.3 に設定します。

4.3 USB インターフェースの接続

ESP8266-SEED-R2 にスケッチを書き込むためには、PC と ESP8266-SEED-R2 を接続するために USB インターフェースモジュールを接続する必要があります。

USB インターフェースモジュールとしては様々なものを入手することができますが、



図 4.2 Boards Manager

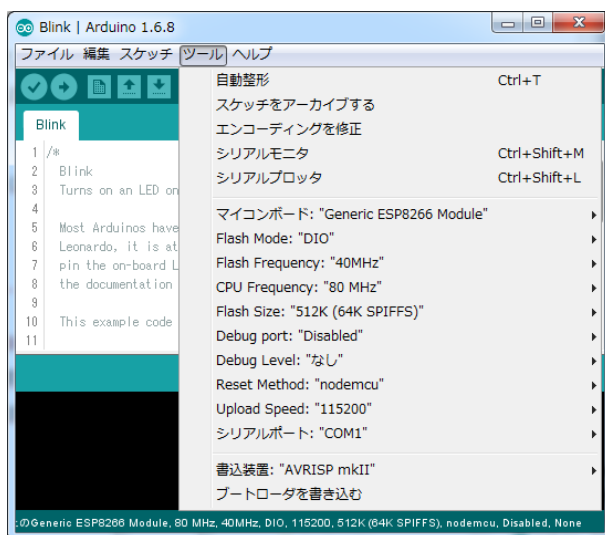


図 4.3 ESP8266 の構成

ESP8266-SEED-R2 で使用するためには、シリアルを送受信信号の電圧が 3.3V である必要があります。

図 4.4 に USB インターフェースモジュールの例を示します。このモジュールは CP2102 を使用しており、シリアルを送受信信号の電圧が 3.3V となっているので、ESP8266-SEED-R2 に接続して問題なく使用することができます。

この USB インターフェースモジュールは、以下の URL で購入することができます。

<http://store.shopping.yahoo.co.jp/microfan/usb-uart-converter-cp2102-microusb.html>

表 4.1 に ESP8266-SEED-R2 と USB インターフェースモジュールの接続法を示しま

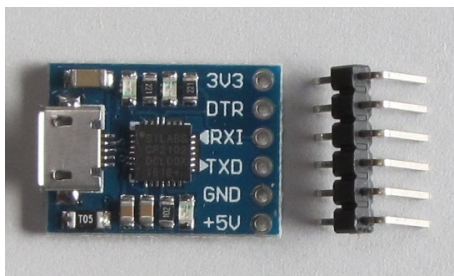


図 4.4 USB インターフェースモジュールの例

す。ESP8266-SEED-R2 の接続端子に関しては、表 5.3 をご参照ください。

送受信の信号線は、双方の送受信の信号線をクロス（相互の TX を RX に）で接続します。

USB インターフェースモジュールから ESP8266-SEED-R2 への電源の供給は、3.3V ではなく 5V で行ってください。USB インターフェースモジュールの 3.3V の電源容量は非常に小さいので、ESP-WROOM-02 の要求量を賄うことができず動作が不安定になります。USB インターフェースモジュールの 5V を ESP8266-SEED-R2 に接続し、ESP8266-SEED-R2 の電源レギュレータで必要な 3.3V の電圧を生成して利用するようにします。

表 4.1 USB インターフェースモジュールの接続

ESP8266-SEED-R2	USB インターフェース
GND	GND
5V	+5V
TX	RXI
RX	TXO

USB インターフェースモジュールを接続した状態では、ESP8266-SEED-R2 には USB から 5V が給電されますので、DC ジャックには AC アダプタを接続しないでください。

4.4 サンプルスケッチの実行

ESP8266 用の Core をインストールすると、Arduino IDE の [ファイル] ⇒ [スケッチの例] に、ESP8266 用の多くのサンプルスケッチが追加されます。これらのサンプルスケッチを試すことで、ESP8266 のプログラミングを学ぶことができます。

ここでは、それらのスケッチとは別に、ESP8266-SEED-R2 の動作確認のために、いくつかのスケッチの実行を試してみましょう。

- 発光ダイオードの点滅
- 照度センサーの利用

- 赤外線 LED と赤外線レシーバーの利用
- WebServer
- BME280 の利用

4.4.1 BLINK:LED の単純な点滅

電子工作界の hello world、LED の点滅スケッチを実行しましょう。

Arduino IDE の [ファイル] ⇒ [スケッチの例] ⇒ [01.Basics] から Blink を選択してください。ESP8266-BLADE/LEAF の LED は 16 番ピンに接続されているので、スケッチの pinMode(),digitalWrite() の第 1 引数の 13 を 16 に変更します。

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(16, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(16, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);            // wait for a second
  digitalWrite(16, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);            // wait for a second
}
```

図 4.5 BLINK:LED の単純な点滅

スケッチのコンパイルとマイコンボードへの書き込み

スケッチの入力・修正が終わったら、まず問題なくコンパイルを行えるかどうか、Arduino IDE の左上部のチェックマーク [検証] のアイコンをクリックして、スケッチをコンパイルします。

問題なくコンパイルできたならば、ESP8266-SEED-R2 の LOAD(SW1) スイッチを押したままの状態、RST(SW2) を押して離します。この後、LOAD(SW1) を離します。この操作で、ESP-WROOM-02 はスケッチを書き込める状態で待機します。

この状態で、Arduino IDE の先ほどのチェックマークの右隣の矢印マーク [マイコンボードに書き込む] のアイコンをクリックします。スケッチの再コンパイルの後に、Arduino IDE の下部のメッセージエリアに橙色の文字でメッセージが出て、ピリオド (点) が連続して表示され、スケッチの書き込みが開始されます。

スケッチが ESP8266-SEED-R2 に正しく書き込まれたら、ボード上の LED が点滅します。

4.4.2 FADE:PWM による LED の点滅

Arduino IDE の [ファイル] ⇒ [スケッチの例] ⇒ [01.Basics] から Fade を選択してください。ESP8266-BLADE/LEAF の LED は 16 番ピンに接続されているので、スケッチの変数 led を 9 から 16 に変更します。

通常の Arduino の analogWrite() の第 2 引数で指定する PWM の強度は 0-255 ですが、ESP8266 の Arduino の実装の初期値では 0-1023 になっています。この上限の値 1023 は、analogWriteRange() を使用して変更することができるので、setup() の中で analogWriteRange(255) を実行して、PWM の強度の範囲を通常の Arduino に合わせます。

```
int led = 16;           // the pin that the LED is attached to
int brightness = 0;     // how bright the LED is
int fadeAmount = 5;     // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 16 to be an output:
  pinMode(led, OUTPUT);

  analogWriteRange(255) ; // for ESP8266
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 16:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

図 4.6 FADE:PWM による LED の点滅

スケッチのコンパイルとマイコンボードへの書き込みは、4.4.1 節を参照してください。
スケッチが ESP8266-SEED-R2 に正しく書き込まれたら、ボード上の LED が徐々に明るくなったり暗くなったりします。

4.4.3 照度センサーの利用

照度センサーの出力値はアナログ値となっており、ESP-WROOM-02 の AD コンバーターに接続されています。AD コンバーターの値は `analogRead(0)` で読み出すことができ、0-1023 の値を取ります。照度センサーの周囲が暗いときには小さな値が、明るいときには大きな値が出力されます。

照度センサーの値を読み出し、Arduino IDE のシリアルモニタに表示するサンプルスケッチを図 4.7 に示します。

ESP8266-SEED-R2 にスケッチを書き込んだら、Arduino IDE の右上部の虫眼鏡アイコンを押してシリアルモニタを開いてください。照度センサの読み取り値が 1 秒ごとに表示され、照度センサを明るい方に向けたり指でセンサを覆って暗くしたりすると、照度センサの出力値が変化するのが確認できます。

シリアルモニタの出力が文字化けする場合には、`setup()` 内で指定したボーレートと、シリアルモニタの右下で選択できるボーレートが合致していることを確認してください。

```
void setup()
{
    Serial.begin(115200);
}

void loop()
{
    Serial.println("Intensity: " + String(analogRead(0))) ;
    delay(1000) ;
}
```

図 4.7 照度センサー

4.4.4 赤外線 LED と赤外線レシーバーの利用

赤外線 LED と赤外線レシーバは、IRremoteESP8266 ライブラリを使用すると手軽に利用できます。IRremoteESP8266 は、Arduino で広く利用されている IRremote を ESP8266 に移植したもので、以下の URL からダウンロードできます。

<https://github.com/markszabo/IRremoteESP8266>

このページの右上の [Clone or download] をクリックし、[Download ZIP] を選択す

ると、ライブラリのファイルを IRremoteESP8266-master.zip としてダウンロードできます。

Arduino IDE のメニューから、[スケッチ] => [ライブラリをインクルード] => [ZIP 形式のライブラリをインストール...] を選択すると、ファイルのダイアログが表示されるので、ダウンロードした IRremoteESP8266-master.zip を指定して [開く] を押します。これで、ライブラリがインストールされ、スケッチで利用できるようになります。

Arduino IDE メニューから、[ファイル] => [スケッチの例] を選択すると、リストに IRremoteESP8266 フォルダが追加されているのが確認できます。IRremoteESP8266 フォルダの中を確認すると、いくつかのサンプルスケッチがあり、選択して実行することができます。

例として、IRremoteESP8266 を利用する際には、赤外線を送信を行う `irsend` コンストラクタでは赤外線 LED の IO ピン 15 を、赤外線の受信を行う `irrecv` コンストラクタでは赤外線レシーバーの IO ピン 12 を指定します。

- `irsend`:赤外線 LED: 15
- `irrecv`:赤外線レシーバー: 12

IRremoteESP8266 のサンプルスケッチ `IRrecvDemo` を ESP8266-SEED-R2 用に修正したスケッチを図 4.8 に示します。

ESP8266-SEED-R2 にスケッチを書き込んだら、Arduino IDE の右上部の虫眼鏡アイコンを押してシリアルモニタを開いてください。赤外線リモコンを赤外線レシーバーに向けてボタンを押すと、そのボタンを押すと発信されている赤外線コードを解析してシリアルモニタに表示します。

シリアルモニタの出力が文字化けする場合には、`setup()` 内で指定しているボーレートと、シリアルモニタの右下で選択できるボーレートが合致していることを確認してください。

赤外線 LED でリモコンコードを送信する場合には、赤外線 LED が割り当てられている 15 番ピンを指定して `irsend(15)` コンストラクタを使用してスケッチを作成します。

4.4.5 WiFi WebServer

ESP8266 Core のサンプルスケッチの中に、いくつかの Web サーバーの例があります。それらはまず ESP-WROOM-02 を既存の WiFi のアクセスポイントに接続させておいて、クライアントはそのアクセスポイント経由で ESP-WROOM-02 の Web サーバーにアクセスするようになっています。

ここでは、ESP-WROOM-02 をアクセスポイントに設定して、スマートフォンなどで直接 ESP-WROOM-02 に接続して利用する例を示します。

- SSID ESP8266
- パスワードなし

```
#include <IRremoteESP8266.h>

int RECV_PIN = 12; // 赤外線レシーバのピン番号を 12 に変更する

IRrecv irrecv(RECV_PIN);

decode_results results;

void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn(); // Start the receiver
}

void loop() {
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume(); // Receive the next value
  }
  delay(100);
}
```

図 4.8 IRrecvDemo

スケッチのコンパイルとマイコンボードへの書き込みは、4.4.1 節を参照してください。

ESP8266-SEED-R2 にスケッチを書き込み、Arduino IDE のシリアルモニタを開くと ESP-WROOM-02 で動作している WebServer のアドレスが表示されます。

スマートフォンなどで、まず ESP-WROOM-02 のアクセスポイント（この場合は SSID:ESP8266）に WiFi 接続します。この次に、接続したスマートフォンなどの WEB ブラウザで、シリアルモニタに表示された WebServer のアドレスに接続すると、[Hello MicroFan!] と書かれた WEB ページが表示されます。

4.4.6 BME280 の利用

ESP8266-SEED-R2 は、温度、湿度、気圧を測る環境センサーとして、BME280 モジュールを搭載することができます。基板上に BME280 モジュールを搭載した例を図 4.10 に示します。

BME280 用のライブラリは、Arduino IDE のライブラリマネージャを利用してインストールすることができます。ライブラリマネージャの検索フィルタに [BME280] を入力して絞り込むと、BME280 用のライブラリは、この文書の作成時点では 3 個ほど列挙さ

```
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>

const char ssid[] = "ESP8266";

ESP8266WebServer server(80);

void handleRoot()
{
  String msg = "<html><head><title>ESP8266 WebSerber</title>"
              "</head><body><h1>Hello MicroFan!</h1>" ;

  server.send(200, "text/html", msg + "<hr>"
              "<h1>" + millis() + "</h1></body></html>");
}

void setup()
{
  WiFi.softAP(ssid); // パスワードなし
  server.on("/", handleRoot);
  server.begin();

  Serial.begin(115200);
  Serial.println();
  Serial.print("WebServer IP address: ");
  Serial.println(WiFi.softAPIP());
}

void loop()
{
  server.handleClient();
}
```

図 4.9 WiFi WebServer

れます。

今回は、[BME280 by Tyler Glenn] とタイトルが付けられているライブラリを選択してインストールします。

ライブラリマネージャのダイアログ上でインストールするライブラリの欄をクリックすると、インストールボタンが表示されるので、最新バージョンを選択して、ライブラリを

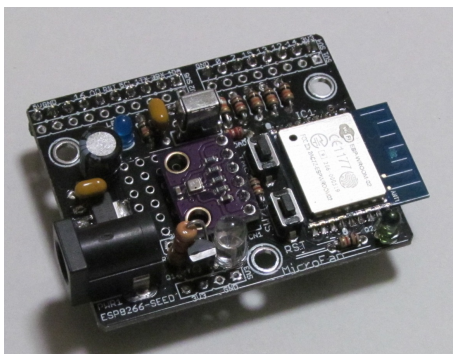


図 4.10 BME280 モジュールの搭載例

インストールします。

このライブラリは、以下の URL で取得することもできます。

<https://github.com/finitespace/BME280>

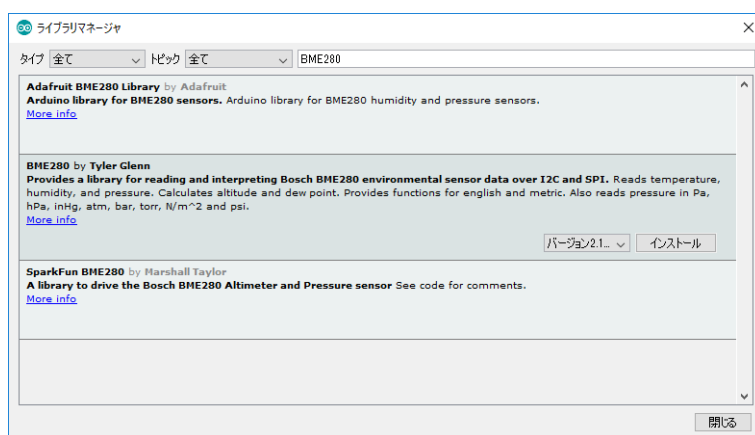


図 4.11 ライブラリマネージャを利用した BME280 ライブラリの導入

ライブラリのインストール後、Arduino IDE メニューから [ファイル] ⇒ [スケッチの例] を選択すると、リストに BME280 フォルダが追加されているのが確認できます。BME280 フォルダの中を確認するといくつかのサンプルスケッチがあり、選択して実行することができます。

ESP8266-SEED-R2 では、BME280 モジュールは I2C で接続されているので、サンプルスケッチのうち [BME_280_I2C_Test] を選択します。このサンプルスケッチは、温度、湿度、気圧およびそれらから算出できる値を 0.5 秒ごとに取得してシリアルモニタに出力します。

サンプルスケッチはこのままコンパイル、書き込んで実行させることができますが、温度が華氏であったり標高がフィートだったりするので、46 行の `bool metric = false;` の `false` を `true` に変更します。この変更により、温度は摂氏で、標高がメートルで表示され

るようになります。

シリアルモニタの出力が文字化けする場合には、`setup()` 内で指定したボーレートと、シリアルモニタの右下で選択できるボーレートが合致していることを確認してください。

第 5 章

資料

5.1 ESP8266-SEED-R2 の回路図

ESP8266-SEED-R2 の回路図を図 5.1 に示します。

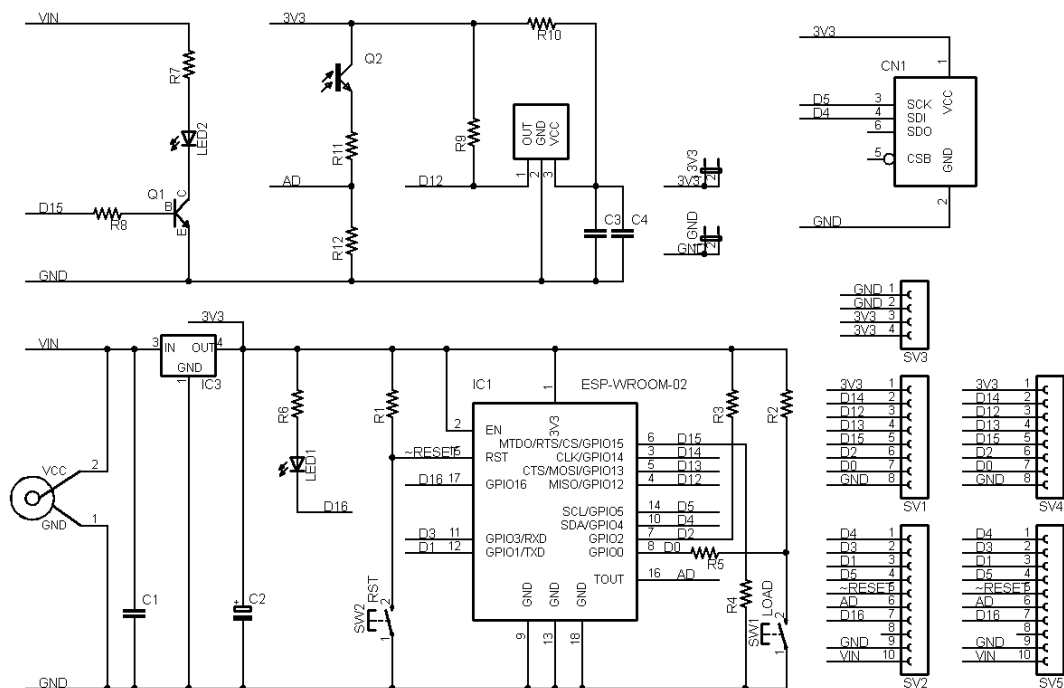


図 5.1 ESP8266-SEED-R2 の回路図

5.2 BOOT モードの選択

ESP-WROOM-02(ESP8266) には、下記の 2 種類のブート（起動）モードがあります。

- Flash Boot Mode

- UART Download Mode

Flash Boot Mode が一般的なブートモードで、ESP-WROOM-02 のフラッシュメモリに書き込まれたプログラムに基づいて処理が行われます。

UART Download Mode は、ESP-WROOM-02 のフラッシュメモリの内容を書き換えるためのモードで、Arduino IDE を利用したスケッチの書き込みや、NodeMCU などの firmware の書き込みに利用されます。

Flash Boot Mode と UART Download Mode の切り替えは、ESP-WROOM-02 の起動時（リセット後）に、表 5.1 に示す 3 端子の状態を設定することによって行います。両者の違いは GPIO(D0) だけで、GPIO0(D0) の状態は ESP8266-SEED-R2 の SW1(Load) スイッチで切り替えることができます。SW1(Load) スイッチが押されていない場合には GPIO0(D) は H、押されている場合には L となります。

表 5.1 BOOT モード

ピン番号	Flash Boot Mode	UART Download Mode
GPIO15(D15)	L	L
GPIO2(D2)	H	H
GPIO0(D0)	H	L

Arduino IDE を利用したスケッチの書き込みや、NodeMCU などの firmware の書き込み時には、SW1 を押したままにしてリセットボタン SW2 を押して離します。これで ESP-WROOM-02 は UART Download Mode に入りますので、SW1 を離し、Arduino IDE でスケッチの書き込みを行います。

5.3 ハードウェア拡張用コネクタ

ESP8266-SEED-R2 には、ハードウェアの拡張を行えるように、ESP-WROOM-02 の信号線の引き出しコネクタが用意されています。信号線は、SV1, SV2, SV4, SV5 としてまとめられ、ヘッダーピンやピンフレームで引き出せるようになっています。SV1,4 のピン配置を表 5.2 に、SV2,5 のピン配置を表 5.3 に示します。

拡張用の信号線は様々な応用に対応できるように、デジタル入出力線、SPI の信号線、アナログ入力線などから構成されています。

また、基板上に BME280 モジュールを搭載するための CN1 端子を備えています。CN1 のピン配置を表 5.4 に示します。

表 5.2 SV1, SV4 ピン配置

ピン番号	信号線	備考	周辺回路
1	3.3V		
2	D14	SCK	
3	D12	MI	赤外線レシーバー
4	D13	MO	
5	D15	SS, プルダウン	赤外線 LED
6	D2	プルアップ	
7	D0	プルアップ	SW1
8	GND		

表 5.3 SV2, SV5 ピン配置

ピン番号	信号線	備考	周辺回路
1	D4	SDA	
2	D3	RX	
3	D1	TX	
4	D5	SCL	
5	RESET	プルアップ	SW2
6	AD	アナログ入力	照度センサー
7	D16		LED1
8	-	未接続	
9	GND		
10	VIN	5V	

5.4 電源レギュレータの電流容量と電圧

ESP-WROOM-02 は 200mA 程度^{*1}の電流を消費します。ESP8266-SEED-R2 で利用している電源レギュレータ AMS1117-3.3 は、最大で 800mA 供給できますので ESP-WROOM-02 を余裕をもって稼働させることができます。

AMS1117-3.3 の入力電圧から出力電圧のドロップダウンは 1V 程度ですので、余裕を見て 4.5V 以上の電源を DC ジャックに接続してください。電源電圧が高くなると、電源レギュレータが発熱することになるため、DC ジャックには、5V 程度の AC アダプタの接続をお勧めします。

^{*1} 瞬間的にはさらに大きな電流を消費するようです。

表 5.4 CN1(BME280 モジュール) ピン配置

ピン番号	信号線	備考
1	3V3	
2	GND	
3	D5	SCL
4	D4	SDA
5	-	CSB
6	-	SDO

第 6 章

購入および問い合わせ先

6.1 ご協力をお願い

製品をより良くし、多くの方々にお楽しみいただけるよう、製品の向上に努めて参ります。問題点やお気づきの点、あるいは製品の企画に対するご希望などございましたら、microfan_shop@yahoo.co.jp までご連絡いただけますようよろしくお願いいたします。末永くご愛顧いただけますよう、お願いいたします。

6.2 販売：ネットショップ

製品の販売は下記のネットショップで行っています。対面販売は行っておりません。

マイクロファン Yahoo!ショップ

WEB アドレス：<http://store.shopping.yahoo.co.jp/microfan/>

6.3 製品情報

マイクロファンの製品情報や活用情報を下記のサイトで紹介しています。ぜひご利用ください。

マイクロファン ラボ

WEB アドレス：<http://www.microfan.jp/>

6.4 問い合わせ先

お問い合わせは基本的にメールでお願いいたします。

株式会社ピープルメディア マイクロファン事業部

E-Mail: microfan_shop@yahoo.co.jp

TEL: 092-938-0450

6.5 所在地

株式会社ピープルメディア マイクロファン事業部
〒811-2316 福岡県糟屋郡粕屋町長者原西 2-2-22-503