

sakura.io + Grove センサーセット [Ver. Arduino]

説明書 第1版 180522

型番:KP-SAKSET01

当製品は「さくらのIoT Platform 共立電子ラビッドバック / 専用SIM付属:KEI-SAKURAI0」用の拡張セットです。実験を行うには必要になりますので別途お買い求めいただきますようお願いいたします。

概要

Grove(seeed studio製)のベースシールドを使用して、Grove システムのセンサーやパーツを取り付け、より実用的な実験を行います。当セットでは、

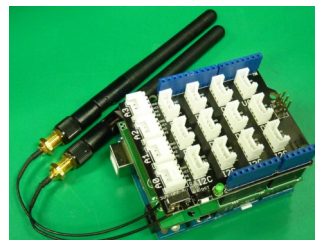
基板 → sakura.io 方向への通信だけでなく、
sakura.io → 基板 への通信実験も行えます。

実験内容

- ・ Grove-Temp & Humi Sensor 温湿度センサー
※WebSocketサービスを使用して温度、湿度の状態を観察します。
- ・ Grove-PIR Motion Sensor モーションセンサー
※WebSocketサービスを使用してセンサー出力の状態変化を観察します。
- ・ Servo Motor サーボモーター(MG-90S)
※Incoming Webhookサービスを使用してネットワークで
接続された遠離地からサーボモーターを操作します。

目次

1. セット内容／部品表	1
2. 通信モジュールの登録	1
3. ハードウェアの準備	1
4. ソフトウェアの準備	
(1) ライブラリの追加	2
(2) スケッチの作成	2
(3) スケッチの書き込み	5
5. ソフトウェアの動作確認 1 (WebSocketサービス)	6
6. ソフトウェアの動作確認 2 (Incoming Webhook サービス)	7



※sakura.io さくらの通信モジュール、専用 Arduinoシールド、Arduino UNOは別売りです。

★サポートに関しまして★

当製品のサポートは、各ハードウェアが正常動作することのみが対象です。
他の製品との組み合わせや「ソフトウェア(開発環境の構築も含む)」「sakura.ioの各種設定・登録」に関してはサポート対象外です。

当説明書にはソフトウェアのコード、各種設定方法を記載していますが、将来バージョンの変更などにより記載方法が変わることがあります。その際はお手数ですが、ご自身でインターネット、書籍などでお調べ頂きますようお願いいたします。(記載情報は、2018年 5月現在のものです)

開発環境の構築、ソフトウェアのコード記載、各種設定方法などは、お電話・メールなどでのサポートは一切行っておりません。

「sakura.io」に関しては下記のお問い合わせ窓口にお問い合わせください。

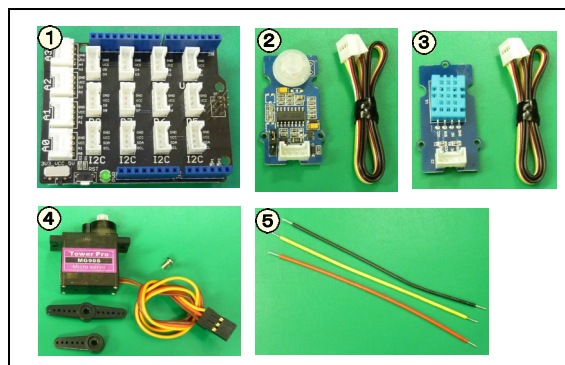
<https://sakura.io/product/>

1. セット内容／部品表

ご使用前にセット内容をご確認ください。
万が一不足、破損がある場合は弊社までご連絡ください。

部品表		
	品名	数量
1	Grove BASE SHIELD V2 ベースシールド V2	1
2	Grove-PIR Motion Sensor モーションセンサー ※専用ケーブル付き	1
3	Grove-Temp & Humi Sensor 温湿度センサー ※専用ケーブル付き	1
4	サーボモーター WR-MG90S ※ねじ、片羽サーボホーン、両羽サーボホーン付き	1
5	配線材 (AWG24) 10cm ※赤、黒、黄、両端ハンダあげ	3

※構成は予告なく変更することがあります。



2. 通信モジュールの登録

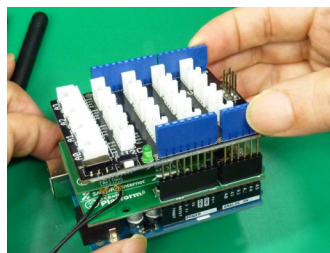
使用するには、さくらインターネットへの会員登録、モジュール登録、プロジェクト作成、基板の設定が必要です。
手順は「共立電子ラビットパック」のマニュアル(P2～P4「通信モジュール登録」)(P5「ハードウェアの準備」(1)の項)を参照ください。

3. ハードウェアの準備

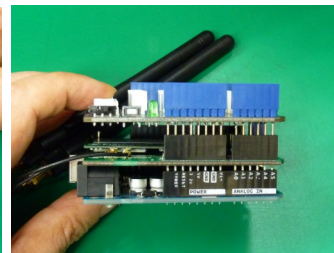
ベースシールド基板を取り付けます。
作業中は絶対に電源を接続しないでください。
電源(ACアダプター、USBケーブル)が入っていないことを確認してください。

「さくらの通信モジュールのついた専用Arduinoシールド」の上に重ねて取り付けます。

差込位置がズレないようによく確認してまっすぐに入れてください。



※ベースシールド基板を一番上にします



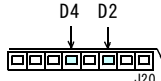
※取り付け側面写真、ピンがズレていないかよく確認してください

●温湿度センサー を「A0」に接続
(専用ケーブルで取り付けます)

●モーションセンサー を「D3」に接続
(専用ケーブルで取り付けます)

●サーボモーターを青色コネクタ「D4」(Arduino)に接続
(付属の24AWG配線材で取り付けます)

D4(黄)-サーボモーター配線(橙)
5V(赤)-サーボモーター配線(赤)
GND(黒)-サーボモーター配線(茶)

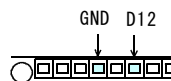


●LEDは青色コネクタ「D2」(Arduino)に接続
(D2をLED(A:アノード)側、GNDに抵抗側を接続)

※サーボモーターに「5V」端子を使用するので「共立電子ラビットパック」のマニュアルと接続のナーの向きが逆になります。

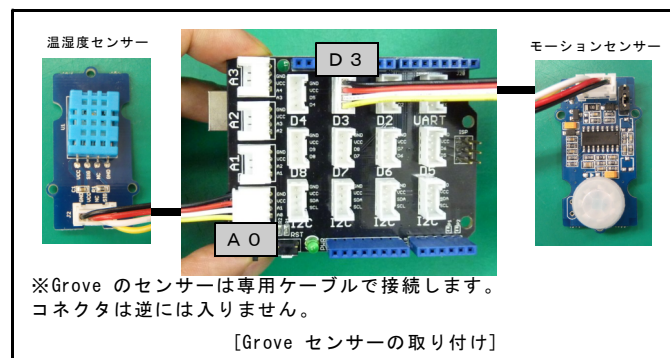
●ブッシュスイッチは青色コネクタ「D12」(Arduino)に接続

片側は「GND」へ接続。極性はありません。
配線のどちらをさしこんでも問題ありません。



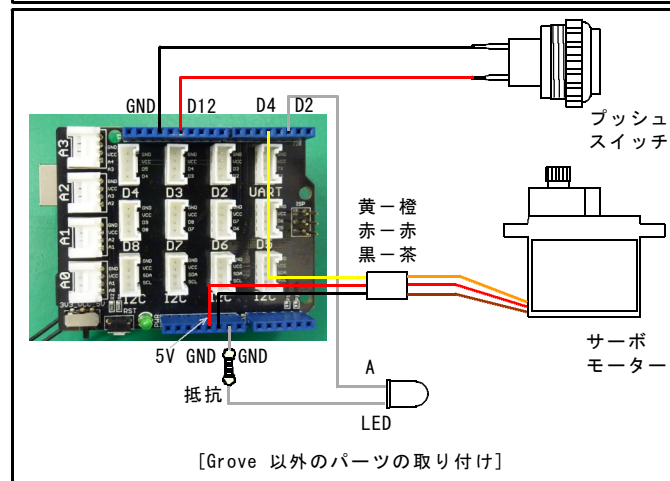
※動作実験用のサンプルスケッチは上記のポート設定で動作するように記載されています。
上記以外の位置に差し込むと正常に動作しません。

最後に接続に間違いがないか確認してから電源を接続してください。電源は付属のACアダプタをご使用ください。
ArduinoのUSB端子を接続しているとArduino基板は動作しますが、通信モジュールは電流不足で正常に動作しませんのでご注意ください。



※Grove のセンサーは専用ケーブルで接続します。
コネクタは逆には入りません。

[Grove センサーの取り付け]



[Grove 以外のパーツの取り付け]

4. ソフトウェアの準備

(1) ライブラリの追加

実験用スケッチを動作させるには下記の2種のライブラリを登録する必要があります。

1. 「SakuraIO」・・・さくらの通信モジュール用
2. 「Grove_Temperature_And_Humidity_Sensor-master」・・・Grove Temperature_And_Humidity_Sensor（温湿度センサー）用

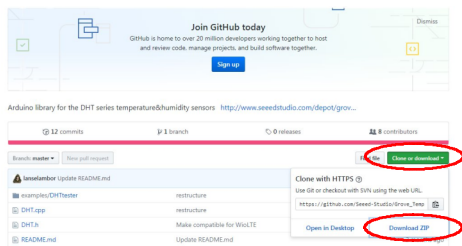
「SakuraIO」ライブラリは「共立電子ラビットパック」マニュアルの(P6)を参照してインストールしてください

「Grove_Temperature_And_Humidity_Sensor-master」ライブラリは下記にてダウンロードしてください。

温度湿度センサー ライブラリダウンロード(英文) Grove_Temperature_And_Humidity_Sensor-master.zip

https://github.com/Seeed-Studio/Grove_Temperature_And_Humidity_Sensor

※URLは2018年5月現在のものです。



下記の手順で

「Grove_Temperature_And_Humidity_Sensor-master」ライブラリをインストールしてください。

1. 上記のURLにアクセスしてください。表示された画面の右端の中央部「Clone or Download」をクリックします

2. 小さいウィンドが出るので、その中の「Download ZIP」をクリックしてダウンロードを開始してください。

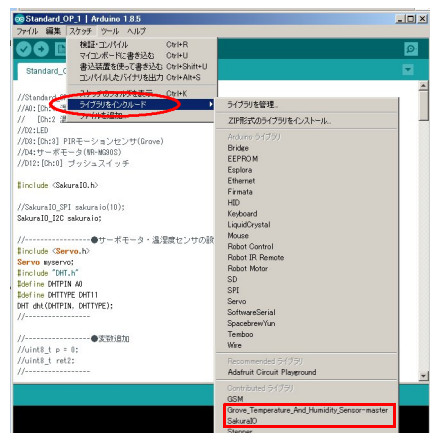
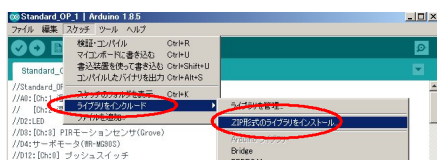
3. ダウンロードしたファイルは ZIPファイルのまま(解凍しない)にしておいてください。

4. Arduino IDE を起動して、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...」をクリックします。

5. ウィンドが開いたら先ほどダウンロードしたZIPファイル「Grove_Temperature_And_Humidity_Sensor-master.zip」を選択して「開く」をクリックするとライブラリがインストールされます。

6. 再度、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」を開き、「Grove_Temperature_And_Humidity_Sensor-master」が見つければインストール成功です。

同じ所に「SakuraIO」もあるか確認してください。ない場合は「共立電子ラビットパック」マニュアルの(P6)を参照してインストールしてください。



(2) スケッチの作成

「共立電子ラビットパック」マニュアルの(P6-13)で基本的な使用方法、ライブラリの追加などの方法をご確認ください。ここでは「共立電子ラビットパック」マニュアルの(P6-13)の内容を実行して正常に動作した環境を想定しています。

スケッチは、sakura.io 提供の「Standard」スケッチをベースにして、Grove基板を使えるように追記します。ベーススケッチは、Arduino IDE メニュー「ファイル」→「スケッチ例」→「SakuraIO」→「Standard」からロードできます。「Standard」スケッチをロードして次ページ以降記載のスケッチの「灰色箇所」を追記、変更してください。

※左端が「//」で始まる所はコメント(解説)行でプログラム上は記載する必要はありません。

※大文字、小文字も識別されますので正確に入力してください。

※コメント行以外に全角の空白、文字が入るとエラーが出ます。

```
//Standard_OP_1 オプションプログラム
//A0:[Ch:1 湿度] 温湿度センサ(Grove)
//  [Ch:2 温度]
//D2:LED
//D3:[Ch:3] PIRモーションセンサ(Grove)
//D4:サーボモータ(WR-MG90S)
//D12:[Ch:0] プッシュスイッチ
```

```
#include <SakuraIO.h>
```

```
//SakuraIO_SPI sakuraio(10);
SakuraIO_I2C sakuraio;
```

```
//-----●サーボモータ・温湿度センサの設定
#include <Servo.h>
Servo myservo;
#include "DHT.h"
#define DHTPIN A0
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
//-----
```

```
void setup() {
//-----●各ポートの設定
pinMode(12, INPUT_PULLUP);
pinMode(2, OUTPUT);
digitalWrite(2, HIGH);
pinMode(3, INPUT);
myservo.attach(4, 544, 2090);
//-----
```

```
Serial.begin(9600);
Serial.print("Waiting to come online");
for(;;){
    if( (sakuraio.getConnectionStatus() & 0x80) == 0x80 )
break;
    Serial.print(".");
    delay(1000);
}
Serial.println("");
```

```
//-----●送信バッファの初期化 サーボの初期化
uint8_t ret;
ret = sakuraio.clearTx();
uint8_t svset = 90;
myservo.write(svset);
ret = sakuraio.enqueueTx((uint8_t)4, (int32_t)svset);
delay(5);
ret = sakuraio.send();
//-----
}
```

```
uint8_t counter = 0;
```

```
//-----●変数を追加
uint8_t v0 = 0; //PIRセンサの1つ前の値を格納
//-----
```