

マイコン勉強会

金沢工業大学 工学部 情報工学科
河並研究室
4EPI-60 山崎 晃平

お品書き

1. はじめに
2. マイコンって？
3. 実際にプログラムしてみよう -1 (LEDを点滅)
4. 実際にプログラムしてみよう -2 (環境センサーを使ってみよう)
5. 実際にプログラムしてみよう -1 (サーボモーターを使ってみよう)
6. クラウドサービスにアップデートしてデータを可視化しよう (ambient)
7. 最新の技術をのぞいてみよう
8. 研究について
9. 終わりに

はじめに

- これはマイコン系の講義である、「マイコンって難しそう…」や「IoTってやってみたいけど何をできるのかわからない」という方に「マイコンってこんなことができるんだ！」って思ってもらいたくて機会を作りました。
- できれば興味を持っていろいろなものを作ってみたいと思ってもらえると助かります…
- 結構すっ飛ばしながら手を動かすのをメインで行きたいので頑張っていきましょう！
- 一応メモ書きで細かく詳細を書いておきます。おそらく飛ばしながらやるので見直して意味を理解してもらえると幸いです。

作成に参考にした本

IoT開発スタートブック —— ESP32でクラウドにつなげる電子工作をはじめよう！

<https://gihyo.jp/book/2019/978-4-297-10736-9>

電子工作 パーフェクトガイド

<https://www.seibundo-shinkosha.net/book/kids/20602/>



興味があったら研究費で購入しましょう！
ボロボロでもよければ貸します。

マイコンって？

センサーやモーターなどものを制御するための小さなコンピュータのこと

- Arduino

Arduino言語と呼ばれるC++, Cに近い言語で開発できるマイコン
マイコンの入門でよく使われる。(工大の講義でも使われます)

- ESP32

Arduinoの開発環境と同様の環境で開発できるマイコン
WifiやBluetoothも使用でき、Arduinoより基本的に性能がいい

- M5stack

ESP32を内蔵して、SDカードやLCD、スピーカーなどなどがまとめて入ってるマイコン。様々なモジュールを積み重ねて使うことで簡単に様々なセンサー、モーター、通信を使うことができる。



M5stackを研究費で購入していただきました。感謝！ただ資料はESP32で…(間に合わなかった)

Arduinoのプログラムを見てみる

マイコンのプログラムを打ってみましょう！

マイコンのプログラムは他の言語と比べ癖があり無限ループの中で処理を行っていきます

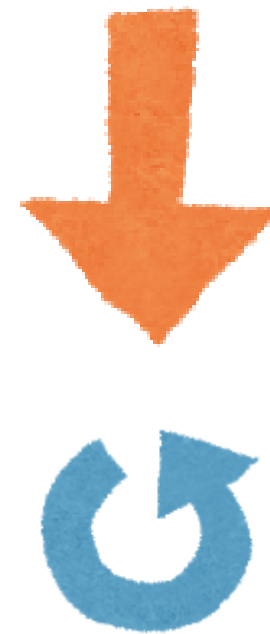
プログラム全体図

Setup()

一度だけ実行される部分。

Loop()

無限ループされる部分
実行が繰り返し行われる。



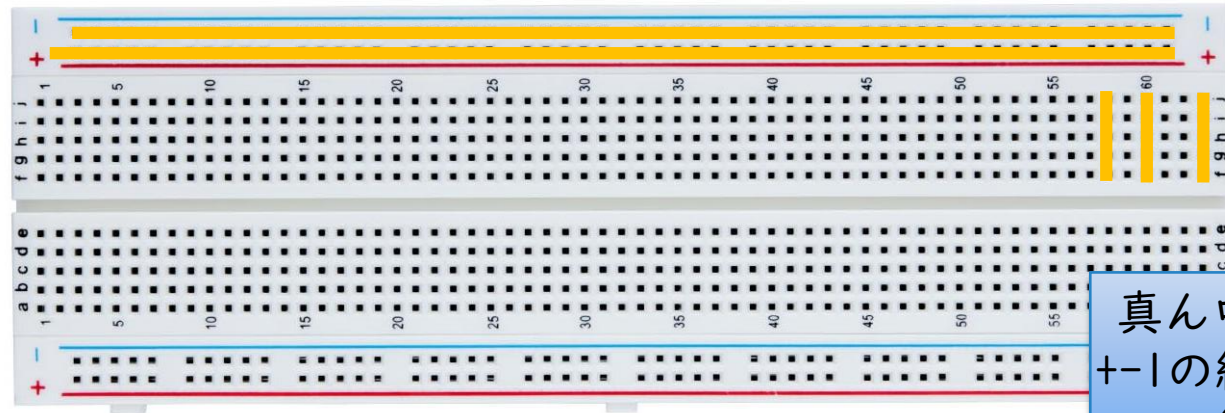
このほかにLoopやsetupの外にマクロ定義(#define)やグローバル変数を宣言したり、関数定義をします。

実際に動かしてみよう！

まずはLチカをしてみます！

LチカはLEDちかちかすることで、プログラムで`print(“Hello,world!”)`のようなお試して動かしてみるのによく使われるものです

まずはブレッドボードとLEDを取り出してみましょう！
ブレッドボードはプロトタイプを使うために使う基板のもの



真ん中の各縦線がつながっていて、
+-の線部分は横につながっています。
(オレンジ線のような感じ)

試しにブレッドボードに配線してみよう

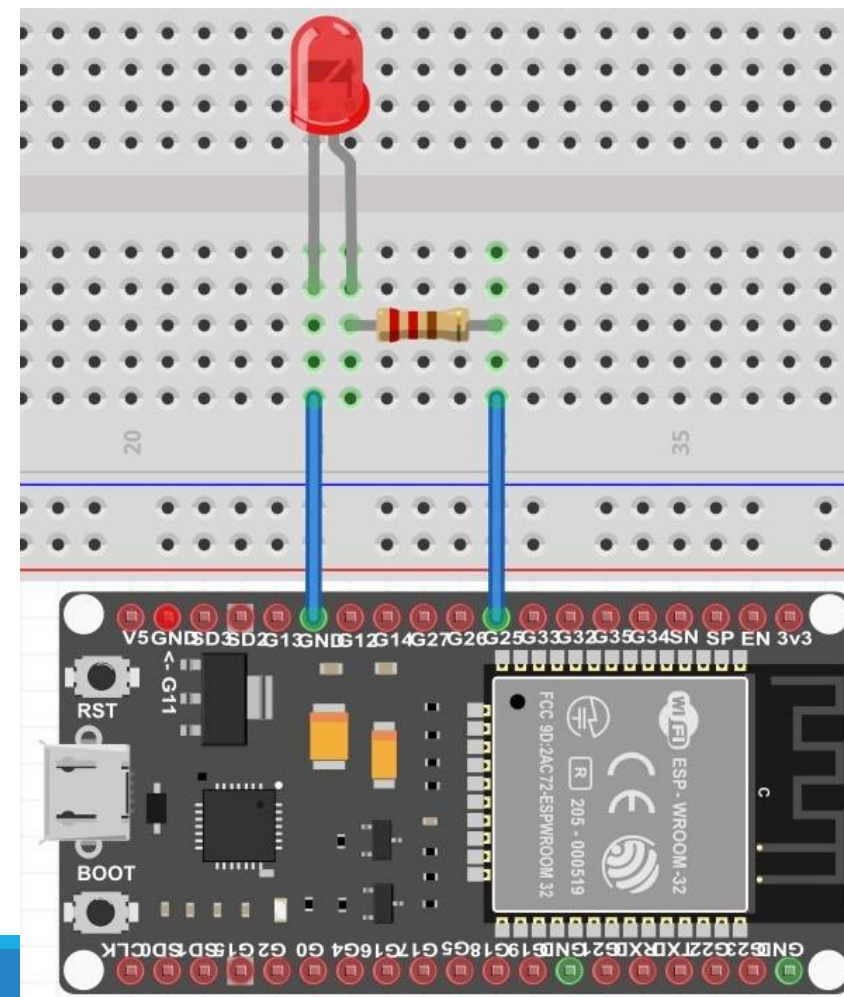
こんな感じで配線します。
あとはプログラムを打っていきます

GPIOピン (デジタル、アナログ)	指定して電気を受け付けたり出力したりする
-----------------------	----------------------

5V, 3V3	電圧(5V, 3.3V)を出力する。
---------	--------------------

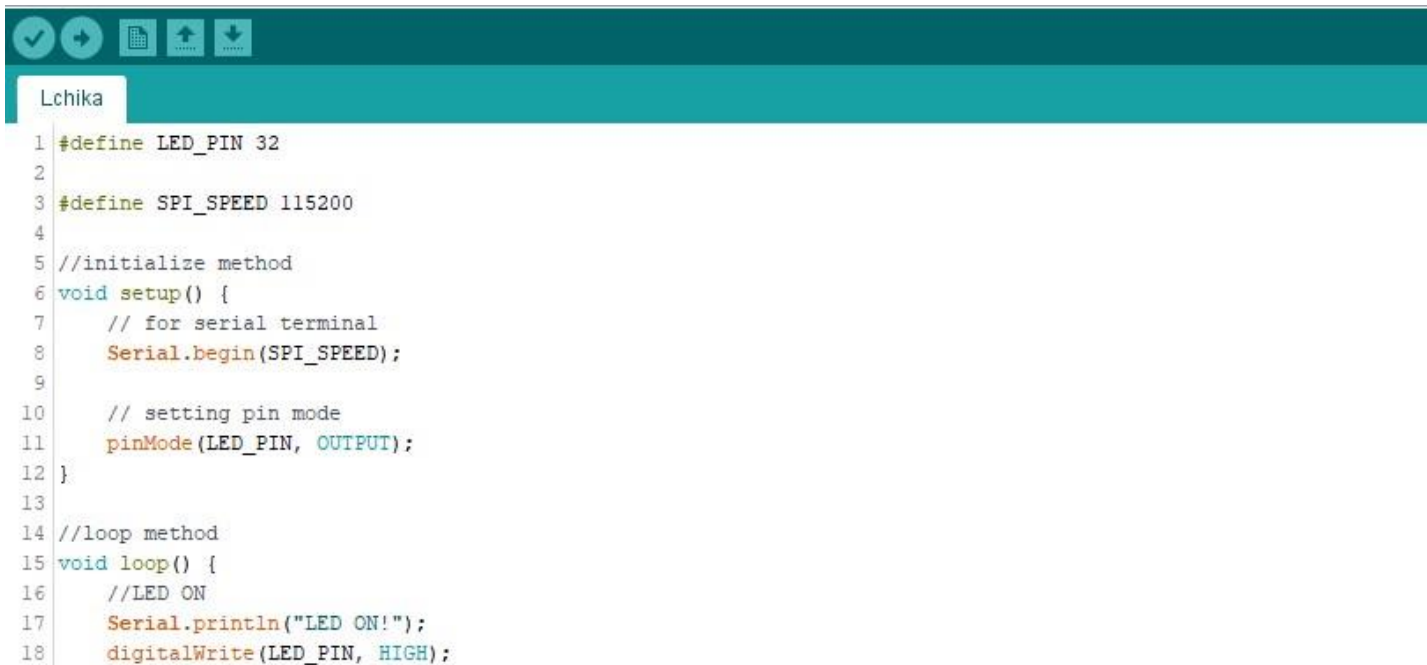
GND	マイナス極の配線部分、電圧(V)の余りをここに垂れ流す
-----	-----------------------------

LEDの短いほうをGND、
長いほうを抵抗を間に入れて
GPIOピンの 25番につなぎます



プログラム

Lchika.inoを開いてみて実行してみましょう！

A screenshot of the Arduino IDE interface. The top toolbar shows icons for checking, uploading, and downloading. Below the toolbar, a tab labeled 'Lchika' is active. The code editor displays the following C++ code:

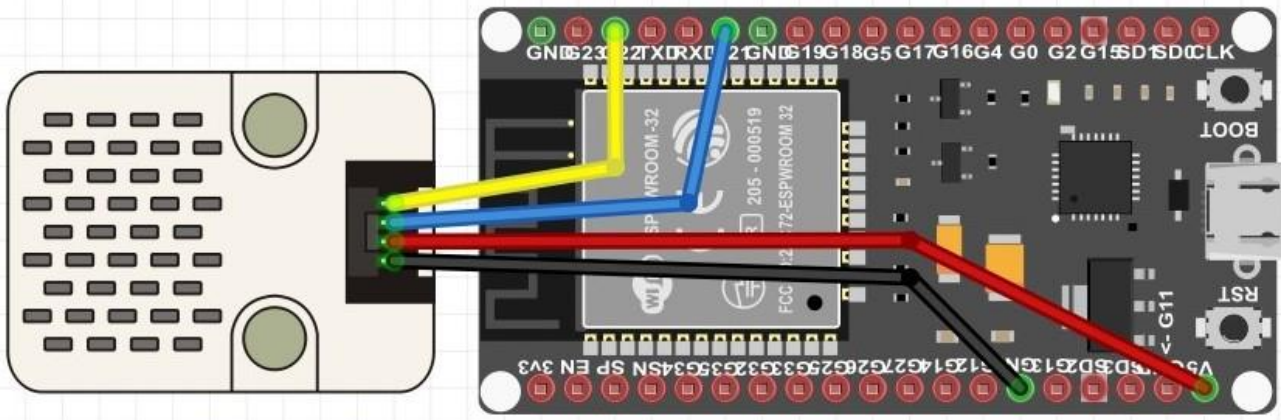
```
1 #define LED_PIN 32
2
3 #define SPI_SPEED 115200
4
5 //initialize method
6 void setup() {
7     // for serial terminal
8     Serial.begin(SPI_SPEED);
9
10    // setting pin mode
11    pinMode(LED_PIN, OUTPUT);
12 }
13
14 //loop method
15 void loop() {
16     //LED ON
17     Serial.println("LED ON!");
18     digitalWrite(LED_PIN, HIGH);
```

LEDが点滅したら成功です！

これでマイコンの世界に入りました！次はセンサーを使ってみましょう

次は環境センサーを使ってみる

次は温度や湿度、気圧を測ってみたいと思います。
これは室内の温度を測って一定に保つために設置したり、台風が来た時に気圧がどうなっているのか測ったり様々な状況で使用できます。
今回はM5stack用に販売されている環境センサーをESP32にぶっ刺して使います。



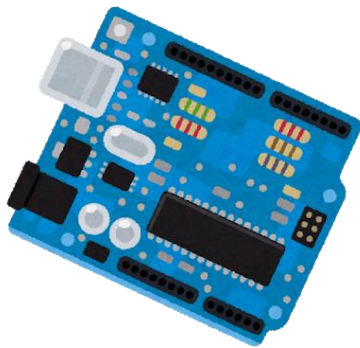
環境センサー	ESP32
GND	GND
5V	5V
SDA	GPIO 21 (I2C SDA)
SCL	GPIO 22 (I2C SCL)

環境センサーというのはI2Cというもので動いています。I2Cってなに？

I2Cって？

I2C(IICとも)

Inter-Integrated Circuitといい、
クロックに同期させてデータ通信を行う同期式シリアル通信のこと。
マイコン側からのクロックに反応してセンサーの値を返信する方式。
アイ・スクエア・シー(アイツーシー)と呼ぶ。



データください！

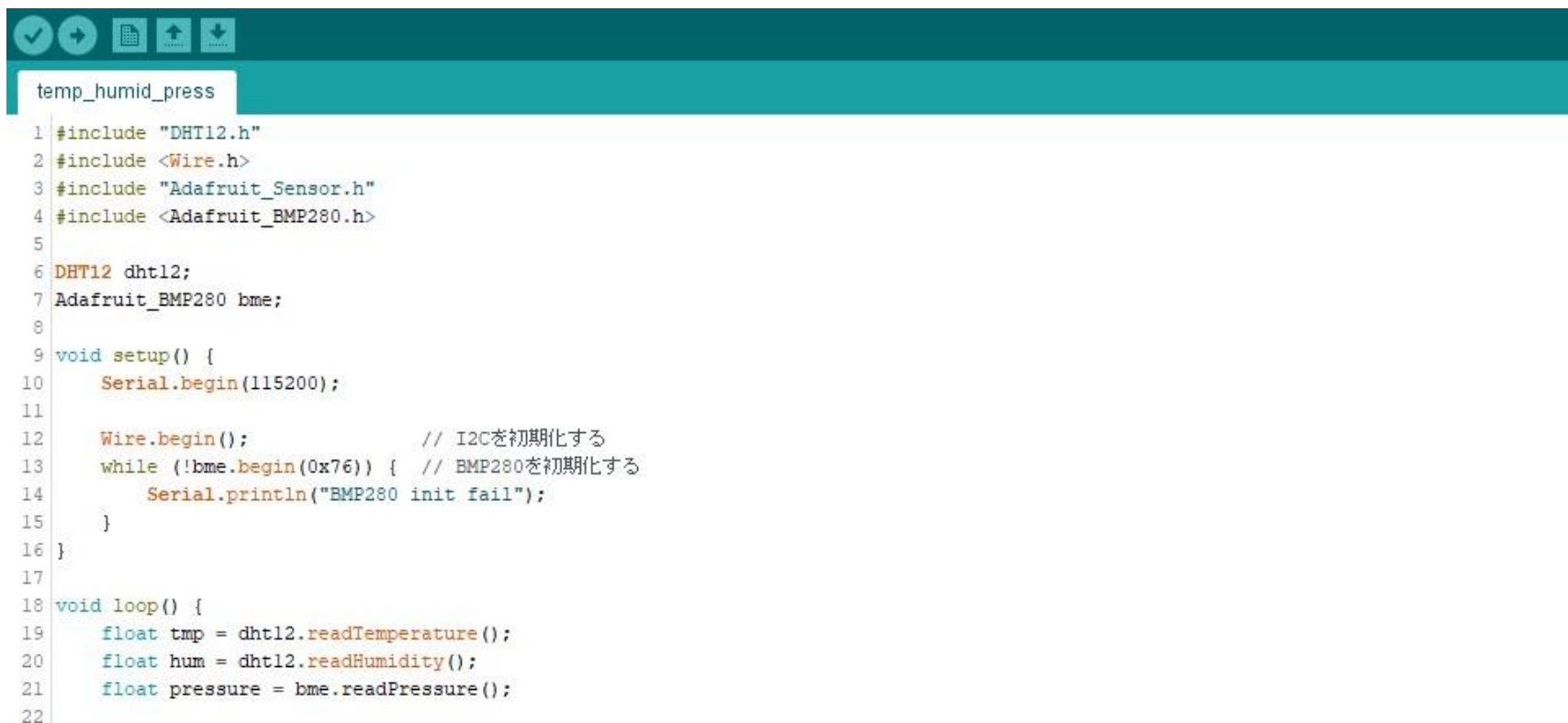
センサーの値



これを使うと
簡単にデータを
取得できる！

プログラムを打ってみる

temp_humid_press.inoを開いてみて実行してみましょう！



```
temp_humid_press
1 #include "DHT12.h"
2 #include <Wire.h>
3 #include "Adafruit_Sensor.h"
4 #include <Adafruit_BMP280.h>
5
6 DHT12 dht12;
7 Adafruit_BMP280 bme;
8
9 void setup() {
10     Serial.begin(115200);
11
12     Wire.begin();           // I2Cを初期化する
13     while (!bme.begin(0x76)) { // BMP280を初期化する
14         Serial.println("BMP280 init fail");
15     }
16 }
17
18 void loop() {
19     float tmp = dht12.readTemperature();
20     float hum = dht12.readHumidity();
21     float pressure = bme.readPressure();
22 }
```

(発展)delay関数って？

メモ

マイコンの開発をしていく上で発展的な部分です。
必要なことには必要ですが、ものを作るという点では必ずしも必要な部分ではないので
飛ばしてもらっても構いません。ただ必要な知識ではあるかもしれません。
(これから組み込み系にかかわるのなら)

IoTをやっていく上で実運用的な点をどうしても考えないといけません。
(自分のしたいことを作ってる分には必要はないかもしれませんが…)
通信や、データのとるタイミング、動かし方に無駄がないか...etc

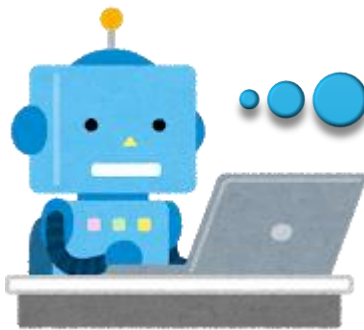


Delay関数というものを使用しているがマイコン的にどうなんだろう・・・
Delay関数ってどういう挙動しているのだろう？電力どうなってる？

(発展)delay関数って？

実はDelay関数は指定した時間までマイコンが数を数えているようになっています。(わかりやすく言うと)なので電力的にはもったいない状態を生み出しています。なのでDeepsleepモードというものを使います。

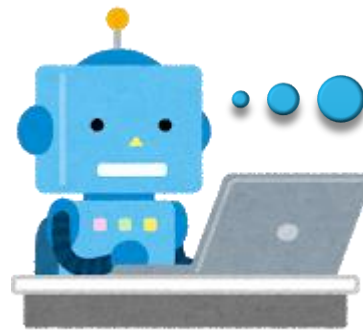
Delay関数



〇〇秒まで
数えっぞ！
1, 2, 3...

電力 大

Deepsleep関数



〇〇秒まで
寝るかあ...

省電力

なので定期的にデータを送るプログラムならDeepsleepモードを使うといいでしょう。
〇〇.inoを実行してみましょう。挙動は変わらないが省電力で動作するはずです。

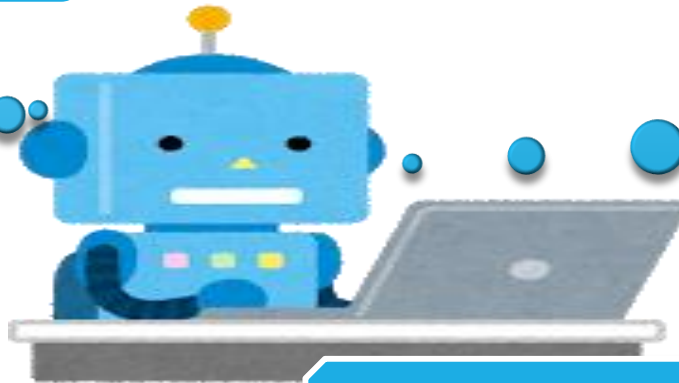
(発展)delay関数って？

Deepsleep関数というものを挙げましたが、delay関数中に動作する必要があるときはどうしましょう？

こういう時はタイマ割込み関数を使用してdelay中にも処理を行えるようにします。

Delay関数

〇〇秒まで
数えっぞ！
1, 2, 3...



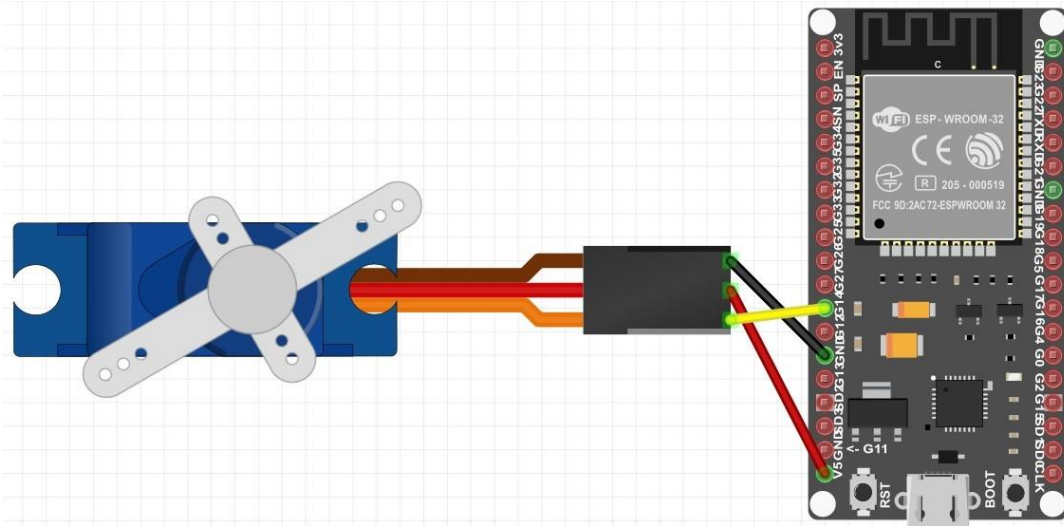
△△秒まで
数えたら
□□をする！

Delayをしながら別の処理ができる

タイマ割込みにはMstimerやTickerというライブラリがあるのでそれを利用します。

次はサーボモータを動かしてみる

サーボモータは車のワイパーのような動きをすることができるモータです
角度を指定して動かすことができます。
まずは接続してみましょう



SG90(サーボモーター)

GND

赤線

5V

茶線

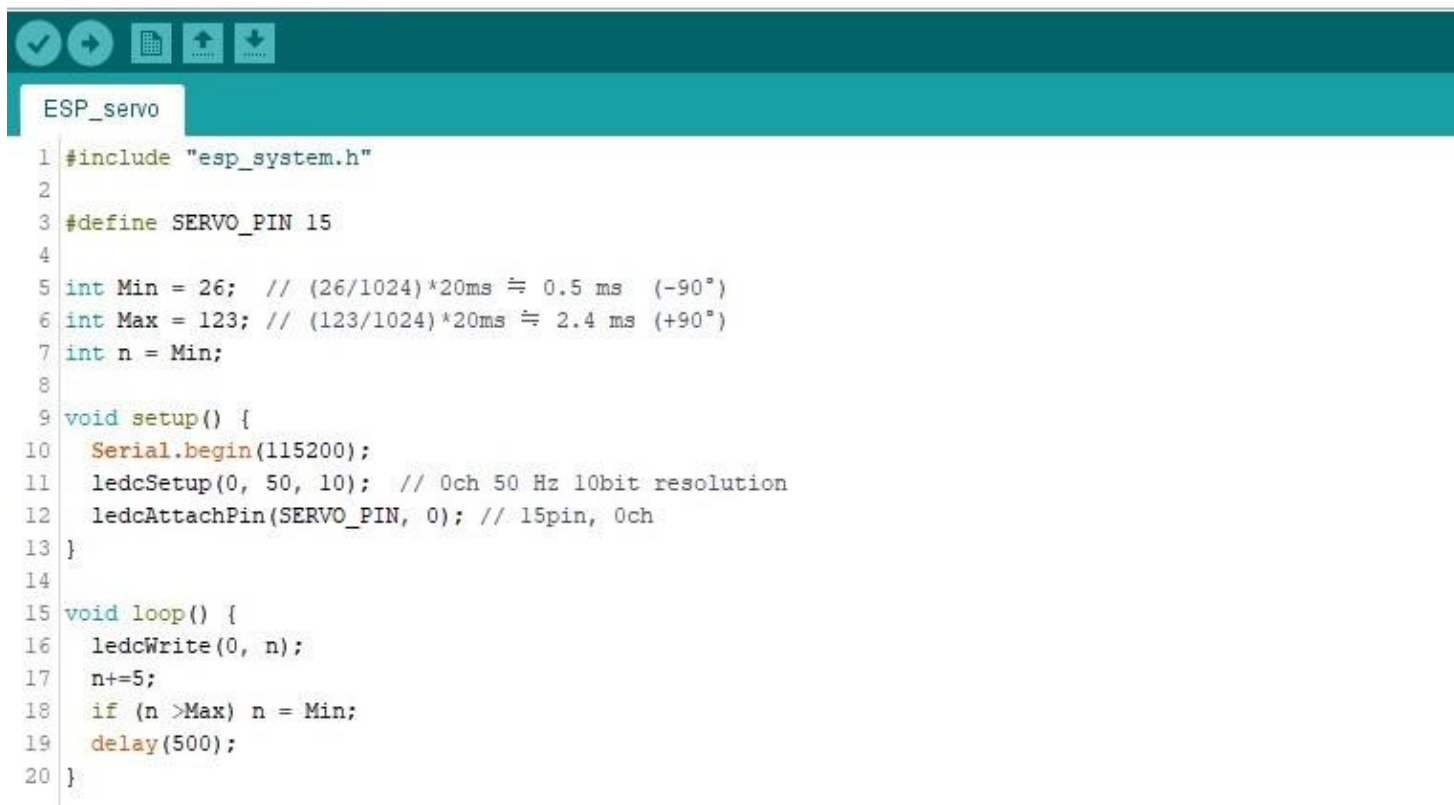
GND

黄線

GPIO 15

プログラムを打ってみる

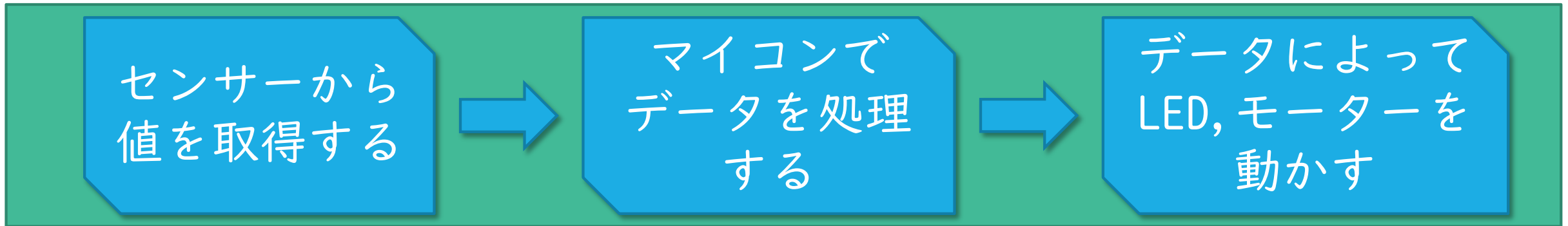
ESP_servo.inoを開いてみて実行してみましょう！

A screenshot of the Arduino IDE interface. The top toolbar shows icons for checking, running, and uploading code. Below the toolbar, a tab labeled 'ESP_servo' is active. The code editor displays the following C++ code:

```
1 #include "esp_system.h"
2
3 #define SERVO_PIN 15
4
5 int Min = 26; // (26/1024)*20ms ≈ 0.5 ms (-90°)
6 int Max = 123; // (123/1024)*20ms ≈ 2.4 ms (+90°)
7 int n = Min;
8
9 void setup() {
10   Serial.begin(115200);
11   ledcSetup(0, 50, 10); // 0ch 50 Hz 10bit resolution
12   ledcAttachPin(SERVO_PIN, 0); // 15pin, 0ch
13 }
14
15 void loop() {
16   ledcWrite(0, n);
17   n+=5;
18   if (n >Max) n = Min;
19   delay(500);
20 }
```

ここまででできたこと

- LEDを制御してみた
- センサーから値を取得した
- サーボモーターを制御してみた



このようなマイコンデバイスの基本的な流れを実現することができるようになりました！

ここまでできたことで実用例(応用してる部分あり)

温度センサー
で部屋の温度
を測る



マイコンで
部屋が暑いのか
どうか調べる



暑かったら
サーボを動かし
エアコンをつける。

人感センサー
で人を
探知する



マイコンで
人が何人目か
数える



100人目の時LEDを
大量に光らせて
お祝いする

距離センサー
で距離を測る



マイコンで
距離が近いか
どうか調べる



近かったら
移動制御を
停止する(自動運転)

次はネットワークにデータをアップデートしてみよう

次はセンサーで取得した値をインターネット上にアップデートし、
可視化を行うことをしてみましょう
可視化を行えることで設置した場所の情報を見ることができます。

例



「台風来るから気圧が変動激しくて片頭痛がつらい…とりあえずどこでも気圧の情報をリアルタイムで見たい…」



「センサー持ち歩くのも面倒だし、かといって気圧を見れないのも嫌」



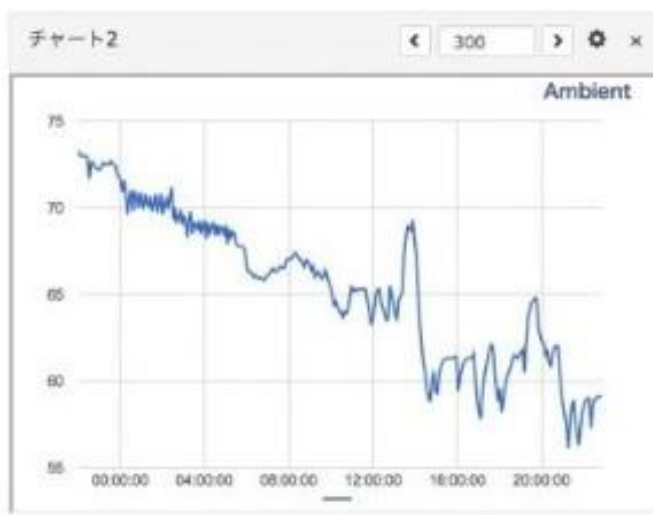
「インターネット上で確認できるようにしよう」

今回はAmbientというサービスを使います。<https://ambidata.io/>でアカウントを登録します。

Ambientとは

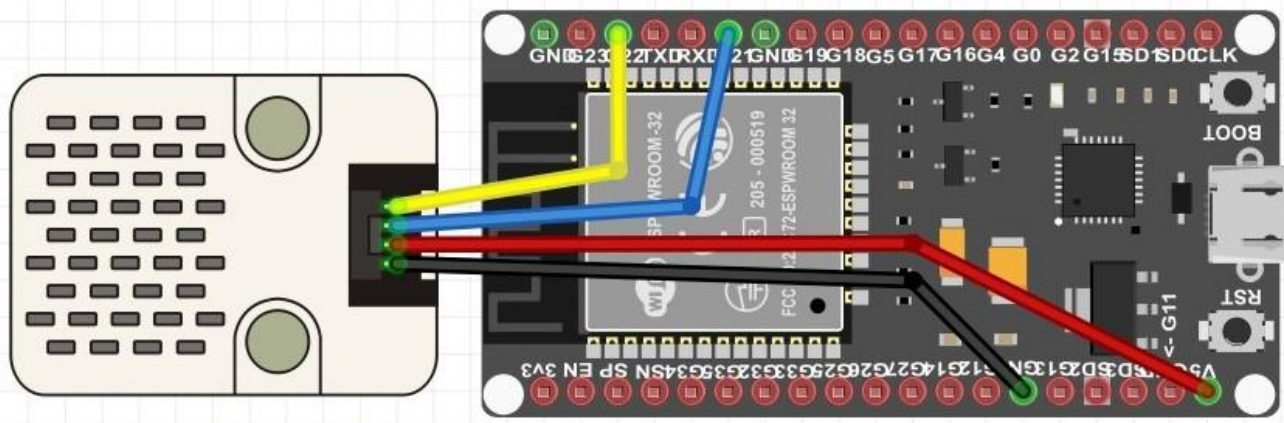
マイコンなどから送られてくるセンサーデータを受信し蓄積、可視化(グラフ)を行ってくれるサイト。

IoTでデータを扱うのに手っ取り早くプロトタイプを作成することができる。



環境センサーをつなげなす

Ambientに送るデータは環境センサーから取得したものとします。
もう一度環境センサーをESP32を接続します。
つなげ方を再掲載します



環境センサー	ESP32
GND	GND
5V	5V
SDA	GPIO 21(I2C SDA)
SCL	GPIO 22(I2C SCL)

環境センサーというのはI2Cというもので動いています。I2Cってなに？

プログラムを打ってみる

Ambient_for_esp32.inoを開いてみて実行してみましょう！

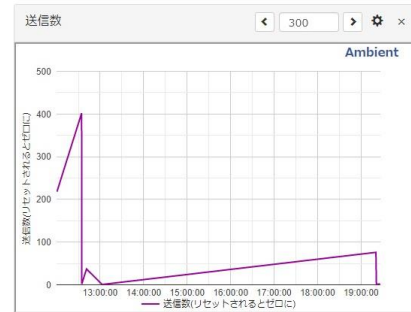
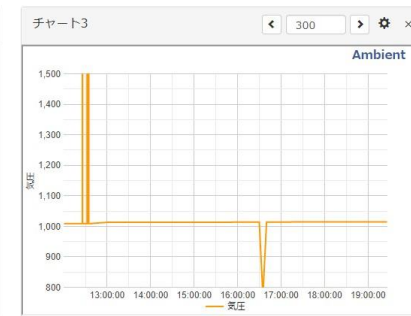
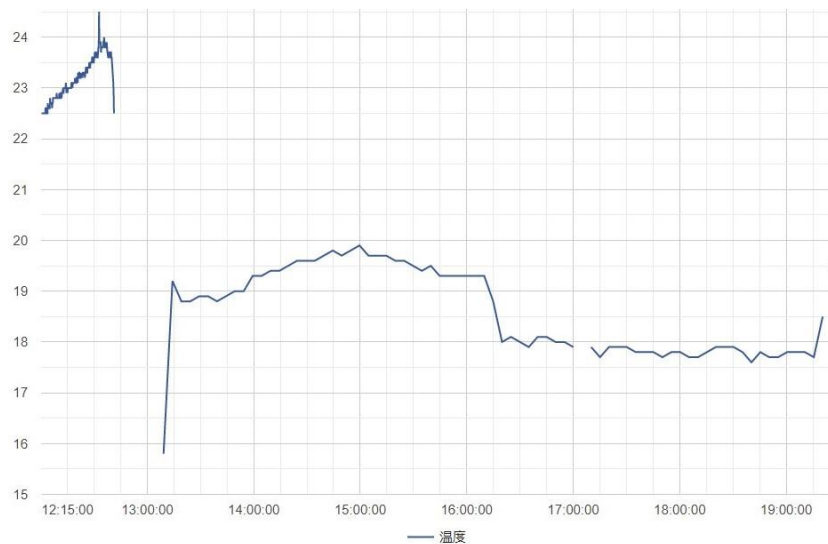
Ambient_for_esp32

```
1 #include "DHT12.h"
2 #include <Wire.h>
3 #include "Adafruit_Sensor.h"
4 #include <Adafruit_BMP280.h>
5 #include "Ambient.h"
6
7 DHT12 dht12;
8 Adafruit_BMP280 bme;
9
10 #define uS_TO_S_FACTOR 1000000 /* Conversion factor for micro seconds to seconds */
11 #define TIME_TO_SLEEP 6 /* Time ESP32 will go to sleep (in seconds) */
12
13 WiFiClient client;
14 Ambient ambient;
15
16 const char* ssid = "ssid";
17 const char* password = "password";
18
19 unsigned int channelId = 100; // AmbientのチャンネルID
20 const char* writeKey = "writeKey"; // ライトキー
21
22 void setup() {
23     Serial.begin(115200);
24
25     Wire.begin(); // I2Cを初期化する
26     while (!bme.begin(0x76)) { // BMP280を初期化する
27         Serial.println("BMP280 init fail");
28     }
```

Ambientを見ている

グラフを見るのですが、ある程度データがたまってからデータを見たほうがグラフになっているので家に帰って実行して一日後ぐらいにグラフを見てみましょう。

IoTって感じになっているはずです。



これで以上になります！

(おまけ)豊富なセンサー類

今回は環境センサー(温度、湿度、気圧)を使用しましたが、センサーというものはものすごい数存在しています。いくつか面白そうなものを紹介します。

- フォトリフレクタ

ものが近づいたら反応します。ゲーセンのメダルの計数機などに使われています。

- 人感センサー

人が通った時に反応します。通過した人の数を数えるときに使えます。

- 距離センサー

対象との距離を測ることができます。おもちゃの車につけて距離が来たら停止するというギミックなどに利用されます。

- 角速度センサー

角度を計測することができます。Nintendoswitchのジャイロ操作(スプラトゥーンなど)に利用されています。

- 心拍センサー

血中の酸素濃度なども図ることができます。運動する際につけておくとどれだけの負荷がかかっているのかモニタリングできます。

(おまけ)様々なアイデア

IoTやマイコン系モノづくりが昔と比較して触りやすい時代になっています。
少しだけ個人的に面白いネタを紹介します。

- 指紋認証で開くママチャリ

<http://netgeek.biz/archives/150094>

指紋認証を使ってママチャリのカギの代わりをするというもの

- PLENbit

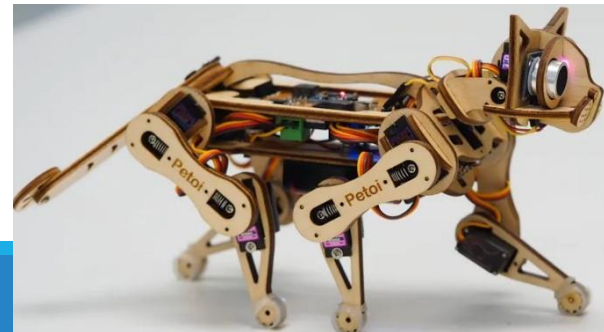
https://twitter.com/PLEN_Project

m5stackやmicrobitなどを使って教育用ロボットを作成するプロジェクト

- Petoï project

https://create.arduino.cc/projecthub/RzLi/petoi-nybble-944867?ref=platform&ref_id=424_trending__&offset=118

Arduinoを用いた猫のロボット。かわいい。



(おまけ) マイコンもAIの時代？

最近では小型のマイコンでもK210チップ搭載のマイコンやJetsonなどと呼ばれるAI(深層学習)と呼ばれるものを実行することができるマイコンが出てきています。少しだけ紹介します。



- M5stickV

K210という軽量の深層学習のモデルを動かすことができることが売りのマイコン。
物体検出や軽度な自動運転などで利用されがち
同会社からカメラ部分に特化したUnitVも最近販売開始(日本まだだけど・・・)

- Jetson series

NvidiaというGPUを発売している会社が発売したGPUが搭載されたマイコン。
Ubuntuで動作するので比較的操作は簡単。
Jetracerと呼ばれる自動運転やK210よりは深い層の学習モデルを動かすことができる。
GPUを利用してグラフィックス的な操作に利用している方も・・・



(おまけ)マイコンを使用したAIの活用例

実用例を紹介します

- Brownie(M5stickV)
M5stickVで作成された物体認識プログラム。数種類の物体を検出して検出できたらAndroidのこれでそれが何か教えてくれます。
<https://github.com/ksasao/brownie/blob/master/README.ja.md>
- JetRacer(Jetson nano)
JetsonnanoのデモとしてNvidiaが公開した自動運転プログラム。
写真を100枚程度(だったはず)撮影してそれを学習、学習内容に沿って運転をする。
<https://github.com/NVIDIA-AI-IOT/jetracer>
- Openpose(Jetson)
動画像から骨格を推定して表示したりできるオープンソースプログラム
実際にはCPUでも動く。でもマイコンでも動かせる。



終わり

- お疲れさまでした！
- 長い時間付き合っただけでありがとうございました。
- 一応就活でも組み込み系できるぞって言えるくらいのことがしたつもりです。
(僕もそういう感じで言ったことがありますw)
- マイコン系が楽しいなって思ってもらえたらうれしいです。
- NT金沢等モノづくりイベントもいろいろあります！ぜひ見に行ってください！
(実は石川県はそういうイベントそこそこあると思ってます。)
- アイデアはTwitterなどいろいろなところから手に入れると面白いと思います。
- (何々をマネするから初めても全然いい！始めることに意味あります。)