

```
#!/bin/env -S marp -p
```

# ハードウェア講習 第1回

# index

- 概要説明
- platformio インストール
- Lチカ?
- ブザー

# 概要

- Arduino
  - ボード及びその開発環境
  - C/C++で開発
- ESP32C3
  - 今回のマイコン
  - 安くて高機能
  - Arduino環境で開発可能

# ボード

- ボタン
- LED(NeoPixel)
- ブザー
- 照度計
- 温度計
- 7セグLEDディスプレイ

# マイコン

- 電球に電池を繋ぐと光る
- トランジスタを使って電気で電気を操れる
- 周期的に変化する信号+データ読み書き+演算=計算機
- Micro Controller Unit (MCU)
- 電圧でデジタル信号を出力

# Arduino

## Arduino UNO REV3

- Arduino公式の一番オーソドックスな開発ボード
- Atmel社の8bitマイコンATmega328P

## Arduino UNO REV4

- Renesas社の32bitマイコン

# Arduino

## Arduino言語

- 名前だけ
- 実質C言語

main.cpp

```
#include <Arduino.h>
void setup(){// 電源投入時一度だけ実行
}
void loop(){// 電源が切れるまで繰り返し実行
}
```



## ESP32C3

- Espressif社の32bitマイコンESP32シリーズ
- 廉価版
- 2.4GHz WiFi BLE

# ArduinoIDE

- Arduinoの用意する統合開発環境
- 動作が遅い
- エディタ縛り
- 使い辛い

==> PlatformIOを使う

# PlatformIO

- Python製のマイコン開発環境
- 開発及びコミュニティの活動が活発
- Arduino環境での開発ができる

# PlatformIO Core

PlatformIO Core をインストール

VSCoDe 拡張であるけどボタンポチポチしたくないからcliで入れる

インストール手引きに従って進める

[get-platformio.py](#)をダウンロード

Linuxの人はパッケージマネージャ使うので関係無し

# Python実行環境

## MacOS

- `python3` 既にある
- `alias python='python3'` しておく

## Windows

- PowerShell開いて[scoop.sh](https://scoop.sh)からscoopをインストール
  - 終わったらターミナル再起動
- `scoop install python`

# インストール

get-platformio.pyのある階層に移動して

```
python get-platformio.py
```

終わると環境変数にパスを通すように言われる

# パス通す

## 公式資料

### MacOS

- .profileと.zprofileをよしなにする

### Windows

- メニューで"環境変数"を検索
- 環境変数を編集>上のリストのPathをダブルクリック
- 新規>パスを貼り付け



# 動作確認

pio

# Lチカ

- NeoPixel
- < <https://akizukidenshi.com/catalog/g/g115068/> >
- ESP32 Arduinoのドキュメントが**ある**
- **ソースコード**に `neopixelWrite(pin,r,g,b);` が定義されている
- 8番ピンに接続してある

- 適当なフォルダを用意
- その中にsrcフォルダを作る
- その中にテキストファイルmain.cppを作る

src/main.cpp

```
#include <Arduino.h>
void setup(){neopixelWrite(8,16,16,16);}
void loop(){}

```

- プロジェクトの設定ファイルが必要
- srcフォルダの入っているフォルダにplatformio.iniを作成
- < <https://docs.platformio.org/en/latest/projectconf/index.html> >

platformio.ini

```
[env:main]
platform=espressif32
framework=arduino
board=esp32-c3-devkitc-02
build_flags=
    -D ARDUINO_USB_MODE=1
    -D ARDUINO_USB_CDC_ON_BOOT=1
targets=upload
monitor_speed=115200
```

- 一ボードとPCをUSBで接続
- `pio run` でコンパイル書き込み

- 点滅させる
- `delay(ms)` で待機

src/main.cpp

```
#include <Arduino.h>
void setup(){}
void loop(){
    neopixelWrite(8, 16, 16, 16);delay(1000);
    neopixelWrite(8, 0, 0, 0);delay(1000);
}
```

# ブザー

- 圧電スピーカ
- < <https://akizukidenshi.com/catalog/g/g104118/> >
- 電圧をかけると変形する素子(セラミック)
- 10番ピンに接続してある

- `tone(pin, freq, dur)`
- [Arduino言語のドキュメント](#)にある
- 周波数の計算
  - 440: ラ
  - $x2$ : 1オクターブ上
  - $xpow(2, 1/12)$ : 1音上



src/main.cpp

```
#include <Arduino.h>

void note(int8_t n){
    tone(10, 440*(pow(2, n/12.+1)), 100);
}

void setup(){}
void loop(){
    note(3);note(5);note(10);
    neopixelWrite(8, 16, 16, 16);
    delay(1000);
    note(10);note(5);note(3);
    neopixelWrite(8, 0, 0, 0);
    delay(1000);
}
```

おわり