

2019年度 人と地域共創センター公開講座(春・夏)

AI/IoTセンサのしくみを知ろう(基礎編)

第9回 無線WiFiをつかう3 & まとめ



徳島大学技術支援部

辻 明典 博士(工学)

E-mail: a-tsuji@is.tokushima-u.ac.jp

講座内容

▶ 講師：辻 明典 (徳島大学技術支援部)

桑折 範彦 (徳島大学名誉教授)

川上 博 (徳島大学名誉教授)

▶ 土曜日：10:00～11:30

▶ 日程：

- ① 5/11 ガイダンス, PC環境設定
- ② 5/18 プログラミングをはじめよう
- ③ 5/25 LED を光らせる(川上先生)
- ④ 6/ 1 温度・湿度をはかる(桑折先生)
- ⑤ 6/ 8 距離をはかる

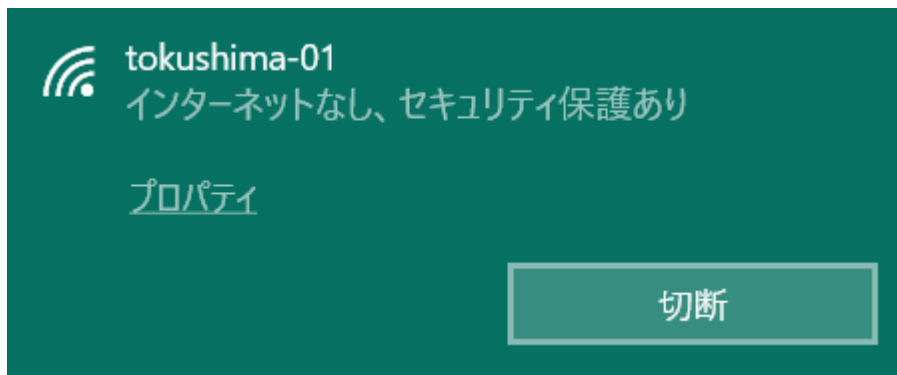
⑥ 6/15 動きをはかる

⑦ 6/22 無線Wi-Fiをつかう1

⑧ 6/29 無線Wi-Fiをつかう2

⑨ 7/ 6 無線Wi-Fiをつかう3
まとめ, 振り返り

トラブルシューティング1



▶ プログラム書き込み前，PCの無線接続を切断

- 無線接続中のままマイコンが起動すると，無線APの起動に失敗する
- PCが無線接続をつづけているため
→ **トラブルシューティング2参照**



- ▶ ブラウザをリロード(再読み込み)
- ▶ 無線を一度切断した後，接続
- ▶ 無線接続エラー時，電源を切って再起動

トラブルシューティング 2

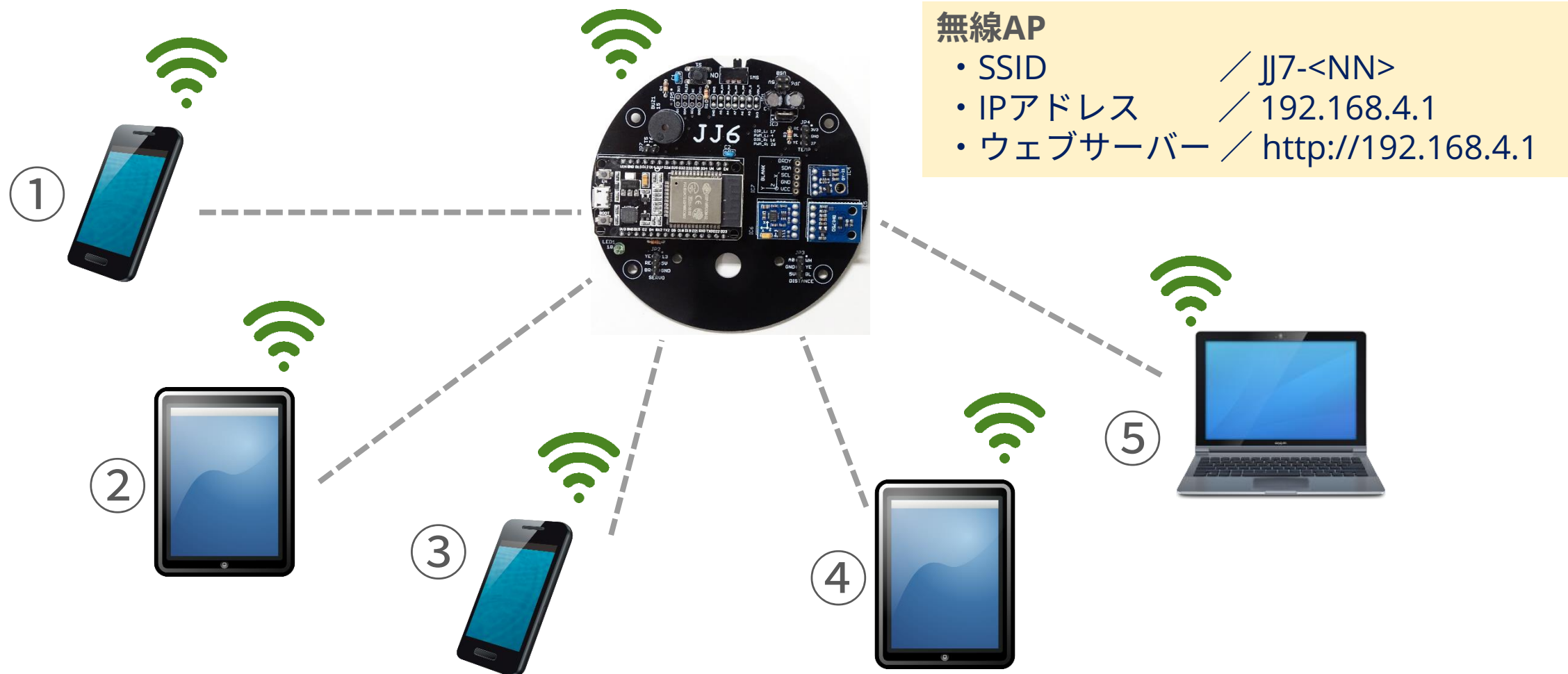
- ▶ **WiFiの初期設定**：無線接続を切断した後，無線APを起動

追加

```
void setup () {  
    . . . .  
    WiFi.disconnect(true);  
    delay(1000);  
    WiFi.softAP(ssid, password);  
    delay(100);  
    WiFi.softAPConfig(ip, ip, subnet);  
    . . . .  
}
```

トラブルシューティング3

▶ クライアント同時接続数：**5台まで**



トラブルシューティング 4

▶ 使用するブラウザ

- WebSocket(※)に対応したバージョン



> 10



> 13



> 11



> 12.10



> 6.0

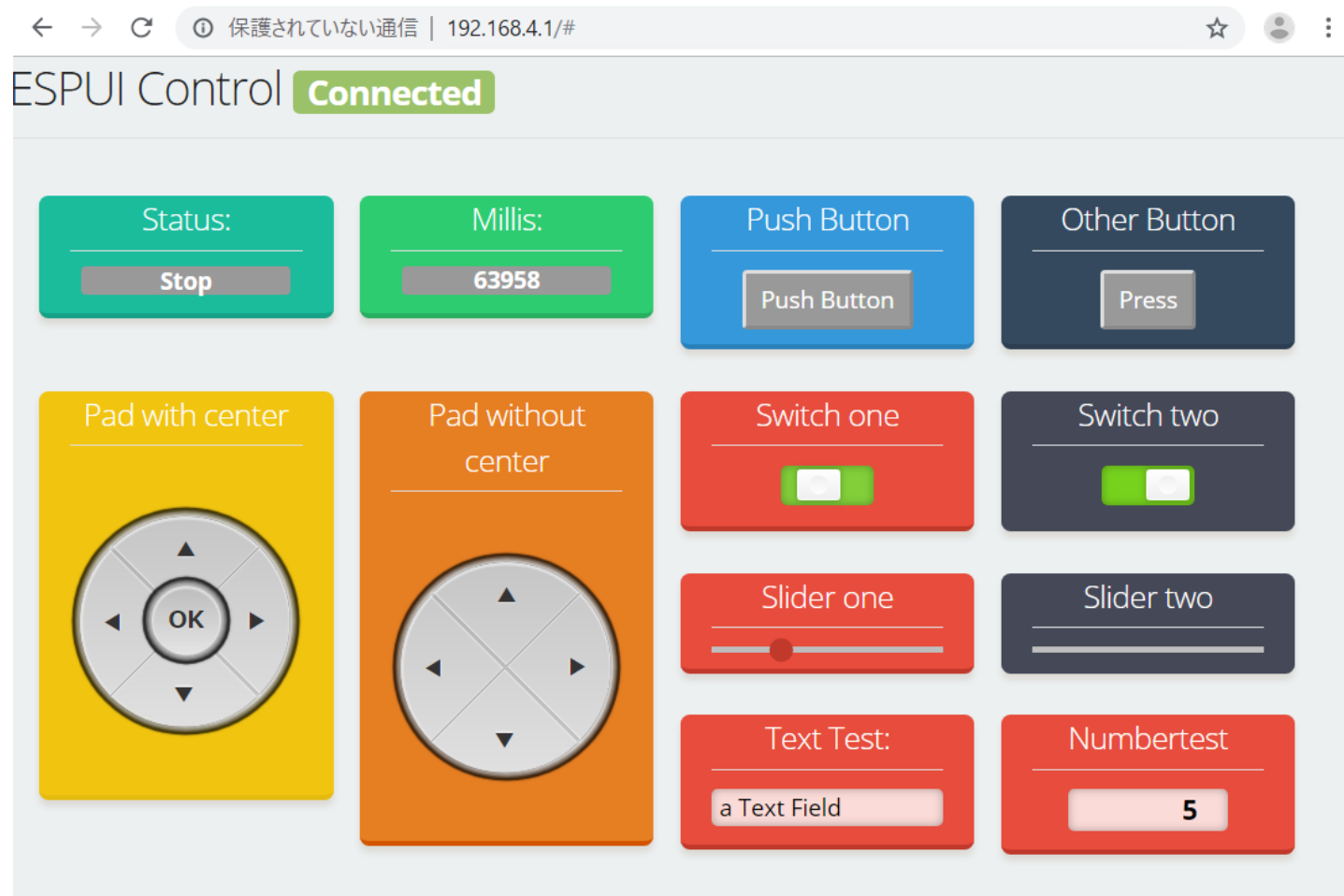


> 18

※ WebSocket | Web上で、簡易に双方向通信を行うしくみ

ESPUIライブラリ

▶ espguiを参照



イベントハンドラー

- ▶ イベントが発生したときに呼ばれる関数
 - メリット | イベントがないときは何もしなくても良い

イベント登録

- ボタン
Button
- テキスト
Text
- スライダー
Slider
- など

イベント発生!

- ボタンが押された
- ボタンを放した
- テキストが入力された
- スライダーが動いた
など

何のイベントか??

イベントハンドラー

```
void ButtonPush()  
void ButtonRelease()  
void TextInput()  
void SliderChange()
```

イベントに対応した関数を実行

Ex0901 | テキスト入出力



- ▶ テキスト入力
 - 文字をキーボードから入力
 - Enterキーを押す
- ▶ テキスト出力
 - テキスト入力した文字列を表示

イベントハンドラー

```
ESPUI.text(  
  "テキスト入力", // 表示ラベル  
  &textCall,     // イベントハンドラー  
  COLOR_ALIZARIN, // 背景色  
  "");          // 初期値(何もなし)
```

```
String inputText;  
void textCall(Control sender, int type) {  
  inputText = sender.value; // 入力した文字をコピー  
  ESPUI.print("テキスト出力", inputText);  
}
```

Ex0902 | 距離センサの設定

- ▶ 距離センサのプログラム
 - 検知距離の最小値，最大値の設定

```
ESPUI.number(  
  "検知距離(最小)",           // 表示ラベル  
  &numberCall_distmin,       // イベントハンドラー  
  COLOR_ALIZARIN,           // 背景色  
  dist_min,                  // 初期値(0)  
  0,                          // 最小値(mm)  
  500);                       // 最大値(mm)
```

イベントハンドラー

```
void numberCall_distmin(Control sender, int type) {  
  String data = sender.value;  
  dist_min = data.toInt(); // 数値に変換  
}
```



Ex0903 | LEDリモコン2

- ▶ スピードアップとダウン
- ▶ スピード調整の変数 sp
 - $sp = \{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$
- ▶ 1000を sp で割る
 - $\text{delay}(1000 / sp)$

```
void loop() {  
  for (int i = 0; i < NUM_LEDS; i++) {  
    leds[i] = CHSV(h, 255, 255);  
    FastLED.show();  
    FastLED.delay(1000 / sp);  
    fill_solid(leds, NUM_LEDS, CRGB(0,0,0)); // 消灯  
  }  
}
```



Ex0904 | LEDリモコン3

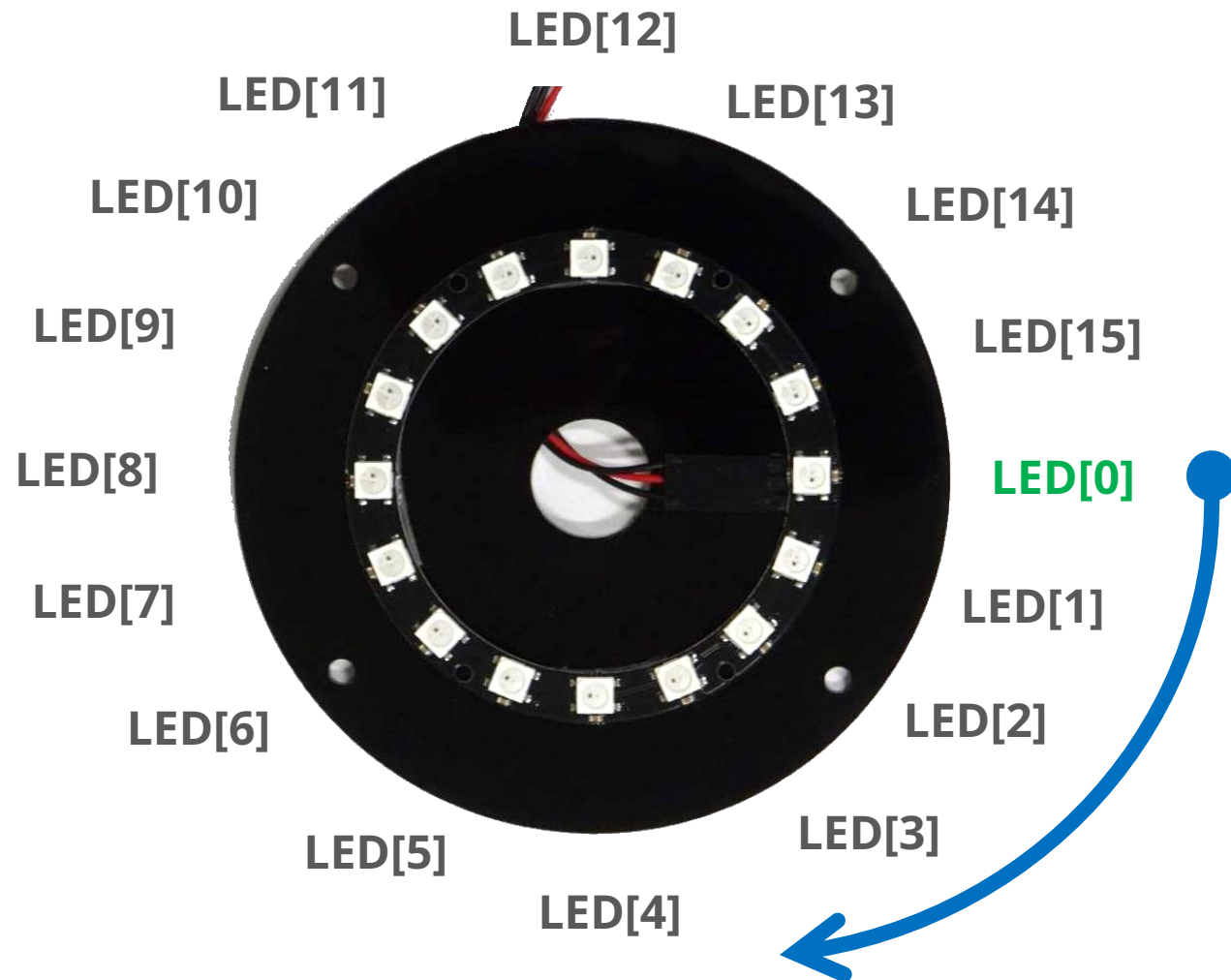
- ▶ LEDを点灯したまま回転
 - LEDの消灯を削除

```
void loop () {  
  for (int i = 0; i < NUM_LEDS; i++) {  
    leds[i] = CHSV(h, 255, 255);  
    FastLED.show();  
    FastLED.delay(1000 / sp);  
  }  
}
```

補足：関数 fill_solidは、すべてのLEDを同じ色に
するとき使用



LEDの配置



- **LED用の配列**

CRGB leds[NUM_LEDS];

- **色の指定**

CRGB(Red, Green, Blue)

CHSV(Hue, Saturation, Brightness)

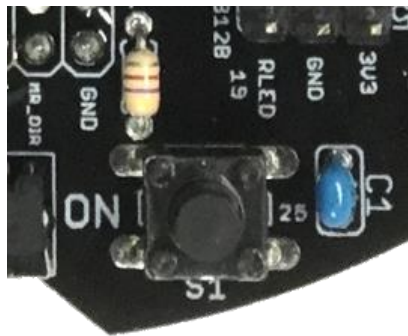
考えてみよう

- ▶ Ex0901 | 別のクライアントから同じ画面を見してみる
- ▶ Ex0902 | 検知距離の最小値，最大値を変えてできること
- ▶ Ex0903 | LED[0]->LED[15], LED[15]->LED[0], 振り子のように光らせる
 - ヒント : `for (int i=NUM_LEDS-1; i > 0; i--)`
- ▶ Ex0903 | 真ん中ボタン付きに変えて，一時停止ボタンを作る
- ▶ Ex0904 | スライダーに置き換えてみる

- ▶ これまでに使用したセンサやLED，ボタン，ESPUIなどを組み合わせてなにか役に立つものを作ってみる

付録：スイッチによる外部割り込み

- ▶ 外部割り込み・・・ハードウェアによる割り込み
 - ・マイコンのポートに直結されたハードウェアの状態変化
 - ・割り込み要因：スイッチ，センサ，IOポートなど



スイッチ押す

ハードウェアによる
割り込み発生



割り込みサービスルーチン
(ISR: Interrupt Service Routine)

```
void IRAM_ATTR sw_event() {  
  led_state = !led_state;  
}
```

Ex09a : スイッチによる外部割り込み

```
const int LED_PIN = 18;
const int SW_PIN = 25;
// LEDの状態
volatile byte led_state = HIGH; //点灯

void setup() {
  pinMode(LED_PIN, OUTPUT);
  pinMode(SW_PIN, INPUT_PULLUP);
  attachInterrupt(
    SW_PIN,
    sw_event,
    FALLING);
}
```

- **割り込み条件**
割り込み関数 `attachInterrupt`内
FALLING . . . 押したとき
RISING . . . 放したとき
CHANGE . . . 押したとき/放したとき
HIGH/LOW . . . HIGHの状態/LOWの状態

```
void loop() {
  digitalWrite(LED_PIN, sw_state);
}
```

```
// 割り込みサービスルーチン
void IRAM_ATTR sw_event() {
  led_state = !led_state;
}
```


まとめ

▶ この講座について



感想



課題



改善



?

次回・・・

講座お疲れさまでした。

秋冬講座は**10月より**始まります。

またの受講をお待ちしています。