

## AI/IoTセンサのしくみを知ろう(応用編)



徳島大学技術支援部常三島技術部門  
技術専門職員 辻 明典 博士(工学)  
E-mail: [a-tsuji@is.tokushima-u.ac.jp](mailto: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 10 / 5 概要，環境設定，配布部品の確認

② 10 / 12 復習

③ 10 / 19 ロボットのモーター1（基本動作）

④ 10 / 26 ロボットのモーター2（応用動作）

⑤ 11 / 9 ロボットのセンサ1  
（距離センサと モーター）

⑥ 11 / 16 ロボットのセンサ2  
（フォトリフレクタ）

⑦ 11 / 30 ロボットの制御1  
（モーター，センサの協調動作）

⑧ 12 / 7 ロボットの制御2  
（ライントレース）

⑨ 12 / 14 ロボットの制御3  
（ライントレース）

# 講座概要

---

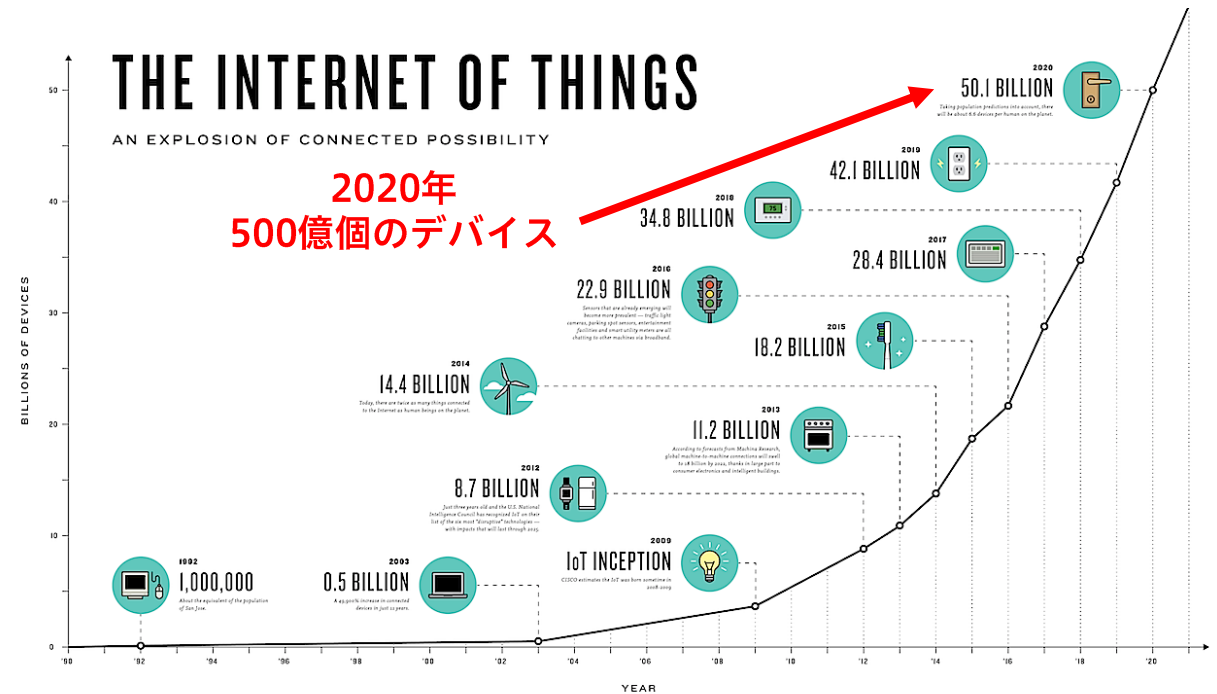
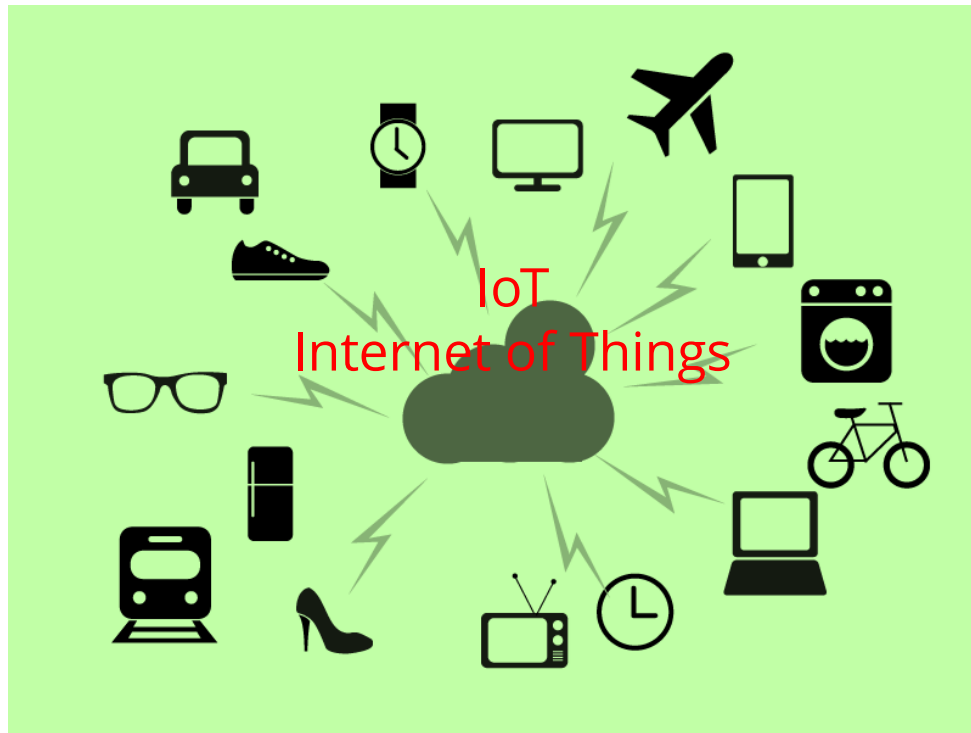
# AI (Artificial Intelligence)

- ▶ 人工知能
  - 言語の理解や推論, 学習, 問題解決などの知的行動を人間に代わってコンピューターに行わせる技術
- ▶ 推論・・・知識をもとに, 新しい結論を得る
  - ゲーム(囲碁, 将棋, オセロなど)
- ▶ 学習・・・情報から将来使えそうな知識を見つける
  - 買い物(おすすを提示)
  - パンを買った人は飲み物も買う
- ▶ 人工知能のアルゴリズム
  - ディープラーニング
  - 遺伝的アルゴリズム
  - 進化計算
  - などなど

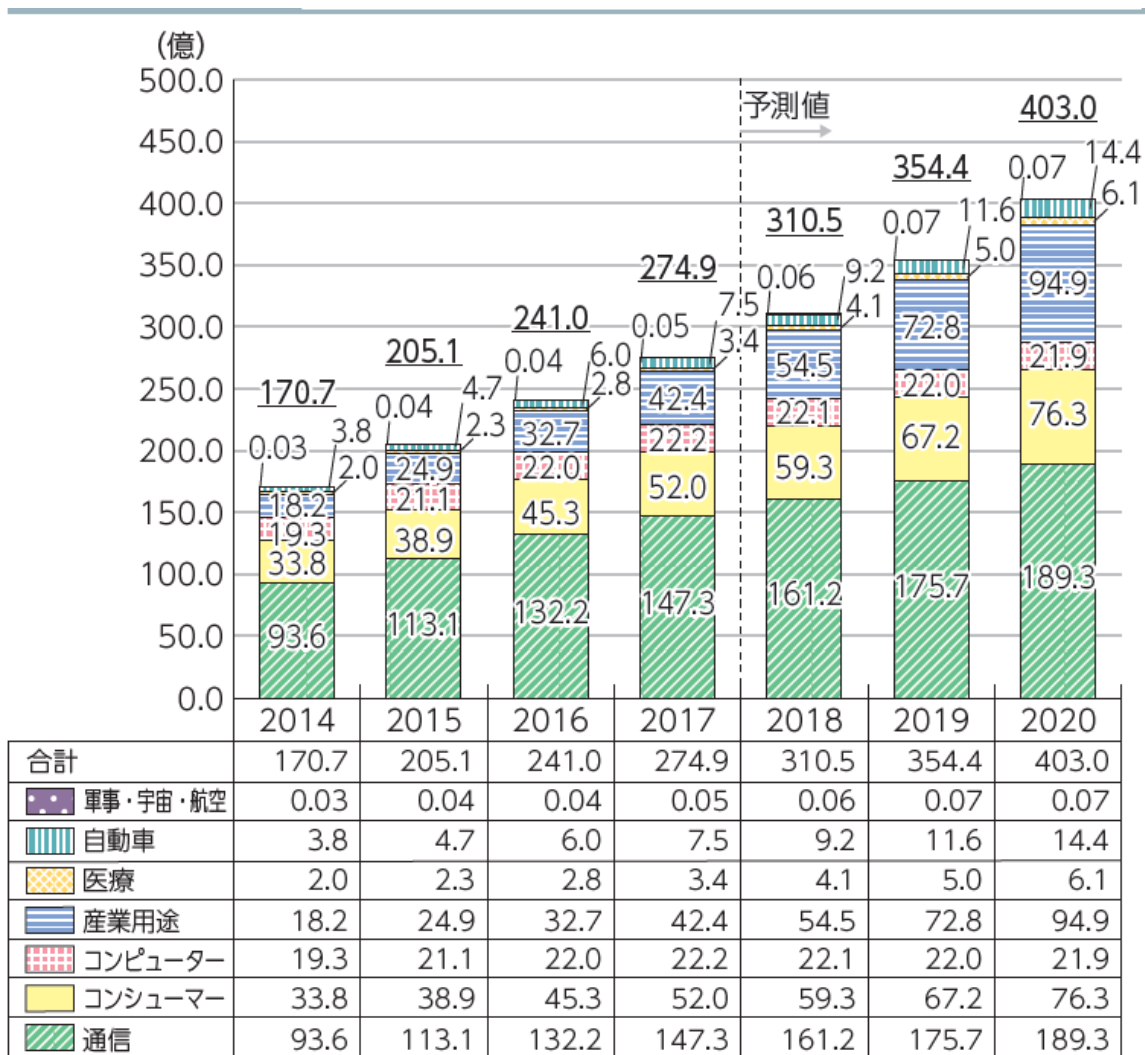
# IoT(Internet of Things)

## ▶ IoT・・・Internet of Things (インターネット・オブ・シングス)

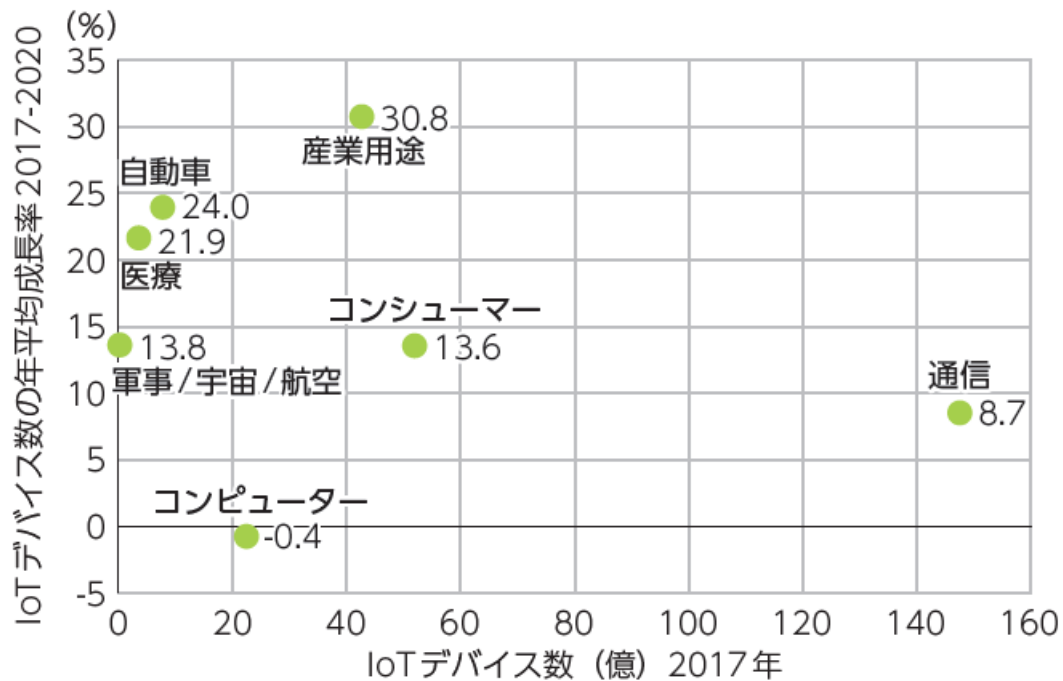
- あらゆるモノをネットワークに接続することにより新たな価値を創出する枠組み
- 2020年にはネットワークに繋がるデバイスは500億個に増加すると考えられている



# IoTの市場



(出典) IHS Technology



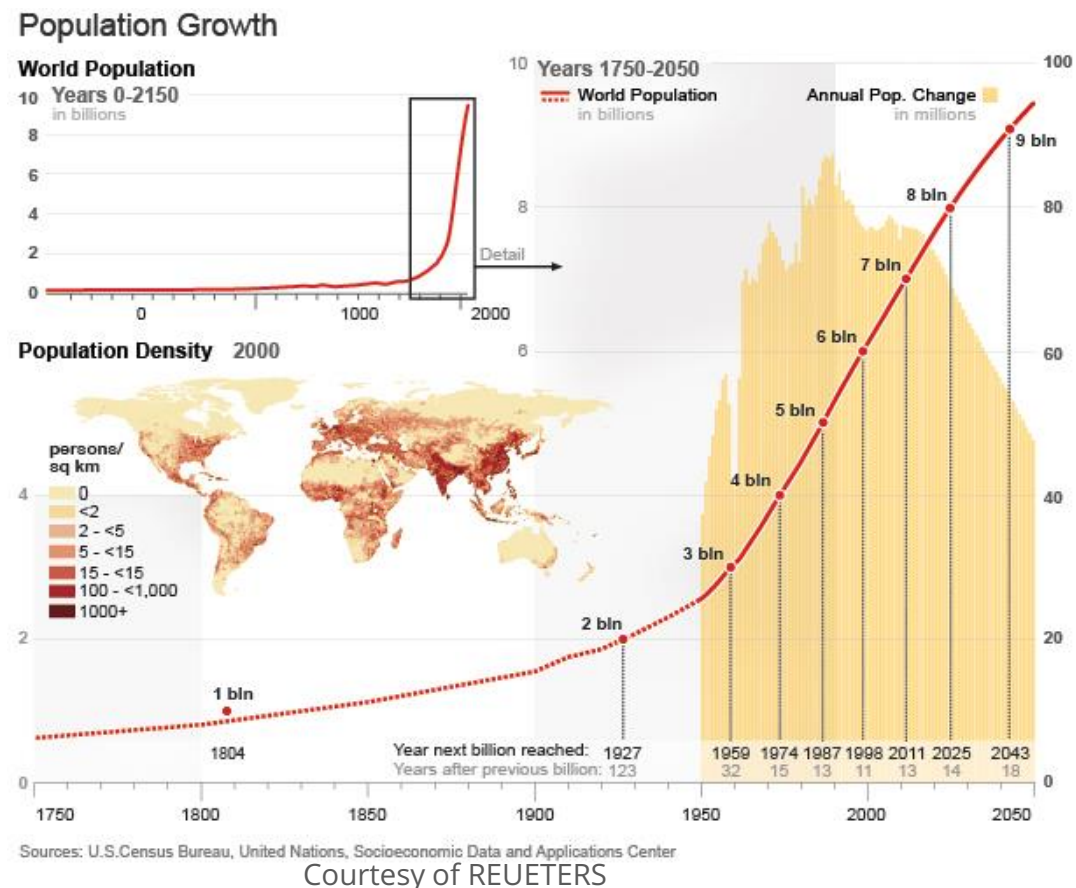
(出典) IHS Technology

# 世界の人口

## ▶ 世界

- 人口増加

## ▶ 人口増加に伴い使用される デバイスも増加

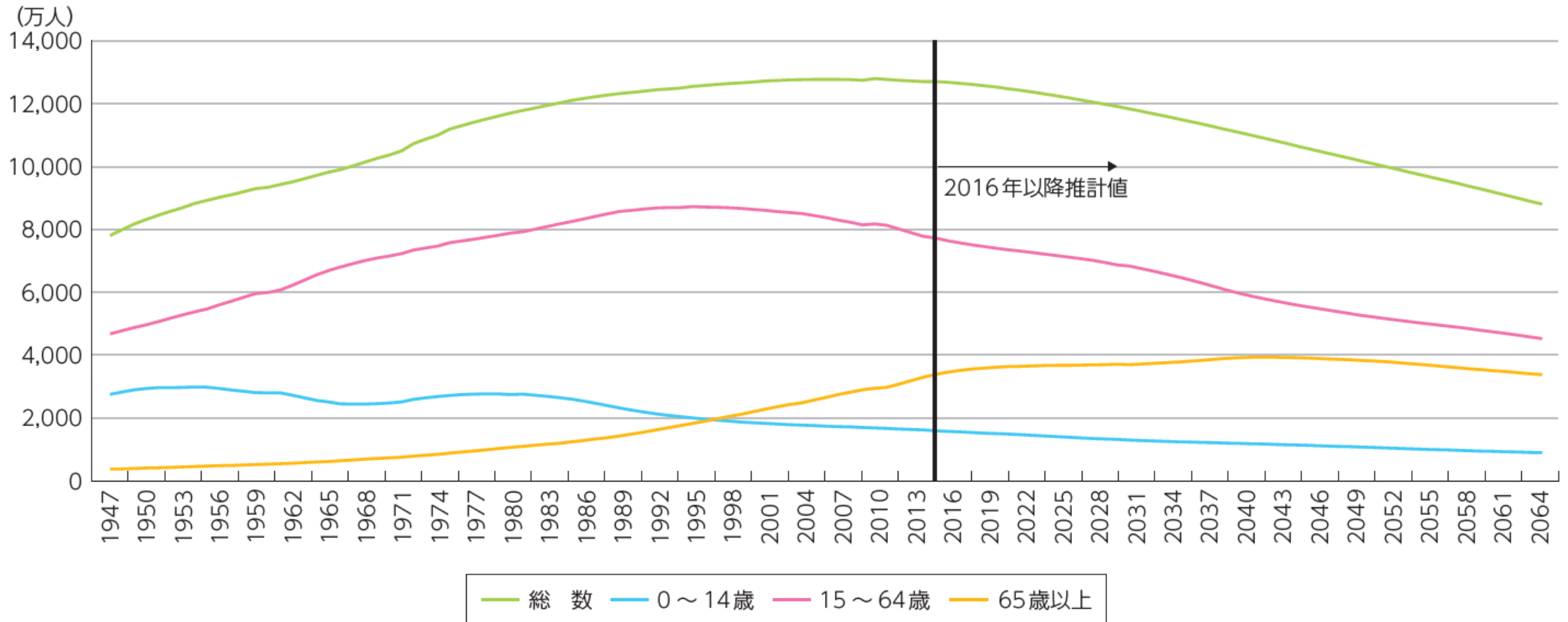


2017年6月21日に国連が発表した「世界人口予測2017年改定版」によると、毎年約8300万人の人口増により、現在76億人の世界人口は、2030年までに86億人、2050年に98億人、そして2100年には112億人に達すると予測されている。2017/06/26 国際連合「世界人口予測・2017年改訂版 [United Nations (2017)].

2019年10月5日(土)

Copyright 2018-2019, Akinori Tsuji, All rights reserved.

# 日本の人口



※ 2018年以降：国立社会保障・人口問題研究所「日本の将来推計人口（平成29年4月）」（出生中位・死亡中位推計）

（出典）2017年まで：総務省「国勢調査」、「人口推計（各年10月1日現在）」  
（総数には年齢「不詳人口」を含み、割合は年齢「不詳人口」を按分補正した人口による。1971年以前は沖縄県を含まない。）



# AI/IoTによる将来課題解決

## ▶ 日本

- ・ 人口減少が続く

## ▶ 平成30年4月現在

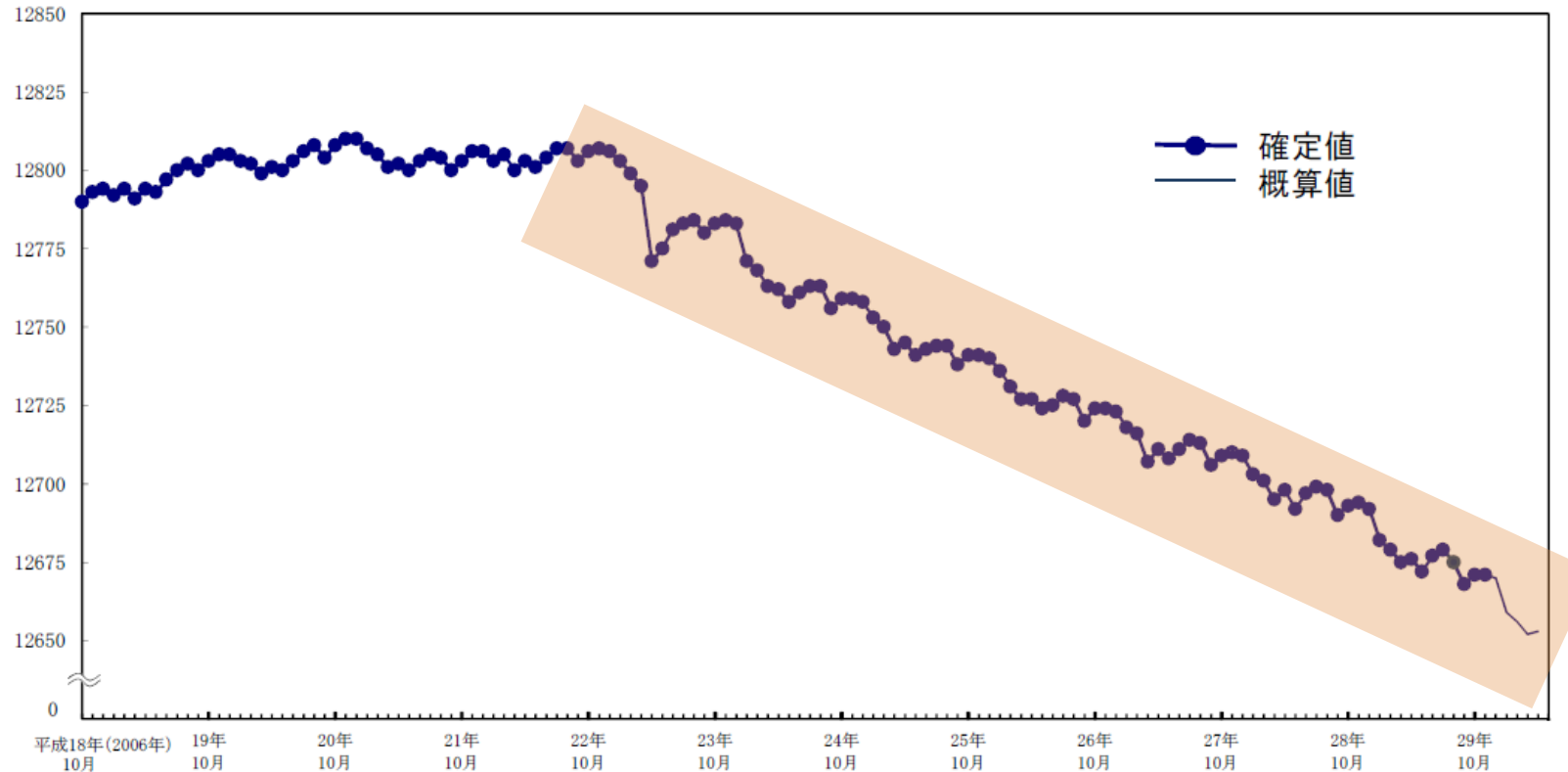
- ・ 1億2643万人

## ▶ 年率, 約20万人毎減少

## ▶ 将来的課題

- ・ 労働力不足
- ・ 経済規模縮小
- ・ 社会保障費拡大
- ・ 少子高齢化
- ・ など・・・

## ▶ AIやIoTなどの技術革新で解決



総務省統計局調べ

# ロボットの構成要素

## センサ

周囲の状況を感じ  
**認知**

人の五感に相当  
・視覚, 聴覚, 触覚  
味覚, 臭覚

×

## コンピューター

状況に応じて  
**判断**

人工知能(Artificial Intelligence)  
機械学習(Machine Learning)  
ニューラルネットワーク  
(Neural Network)  
ディープラーニング  
(Deep Learning)

×

## アクチュエーター

状況に応じた  
**行動**

AC・DCモーター  
サーボモーター  
スピーカー

×

## ネットワーク

情報の**共有**  
通信範囲の**拡大**  
無線通信, 有線通信

# AI/IoT, ロボットに必要な知識

- ▶ 電気・電子回路, モーター, センサ, プログラミング, 機械・筐体設計  
    **コンピューター+ネットワーク**

その他にも . . .

- ▶ 人文科学 . . . 文学・哲学, 世界史, 日本史, 古文など
- ▶ 自然科学 . . . 数学, 物理, コンピュータ, ネットワークなど
- ▶ 社会科学 . . . 社会学, 心理学, 経済学, コミュニケーション論など
- ▶ 語学 . . . . 英語, ドイツ語, フランス語, 中国語など
- ▶ スポーツ, アート・デザイン, ボランティア, 地域活動

→ 応用範囲が広いため, 対象に合わせた**複合的な知識の習得**が必要

# アクティブラーニング教材

- ▶ 学習者自身が課題の発見や解決に向けて主体的に学ぶ体験型の学習方法
- ▶ 目的：知識や技術の習得，学習意欲の向上
- ▶ 工学系の例
  - 電気回路や電子回路の座学は難しい
  - 教科書の理論やシミュレーションでは理解不足
- ▶ 本講座では，
  - 実際の回路を用いて実践的な演習を行う
  - ノートパソコンがあれば**何処でも**実習ができる
  - **実際に動作させて**確認できる
- ▶ 経験的に学習することで，より一層知識を深められる
  - センサ，電気回路，電子回路，無線通信，LED，モーター，プログラミングなど

# 講座を始める前に・・・

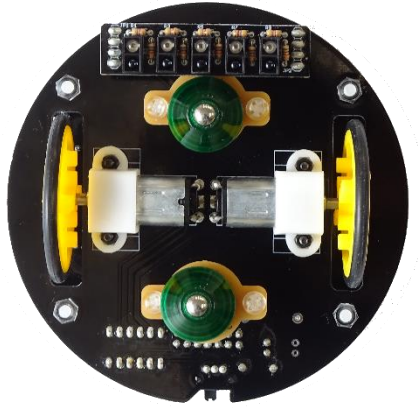
## ▶ 諸注意

- 講師の注意事項をお守り下さい。
- 講座で使用するマイコンボードやセンサは学習用です。装置に組み込んだり改造などを行わないで下さい。
- マイコンボードやパーツ（部品），電池等は，各自の責任の下で管理を行ってください。
- 演習中は，けがの無いよう安全に気をつけてください。
- 動作中の基板を金属などの導電性のものに近づけないでください。基板や部品がショートして破損します。
- マイコンボードに異常（発熱，発火，煙など）を感じたときには，すぐに電源を切り使用を中止してください。
- マイコンボード上の部品を手で触らないでください。静電気等により破損の原因になります。
- 電池の極性（プラス・マイナス）を間違えたり，指定以外の電池を使用しないで下さい。発火や液漏れなどの危険性があります。
- LEDが発光中，光源を直視しないでください。

# 配布部品一覧

	名称	規格	個数	備考
1	モーター基板	モーターボード	1	
2	フォトリフレクタ基板	フォトリフレクタボード	1	取り付け済み
3	ナット	3mm	4	
4	電池	アルカリ単3電池	4	

# 配布部品一覧



1. モーター基板



2. フォトリフレクタ基板  
(取り付け済み)

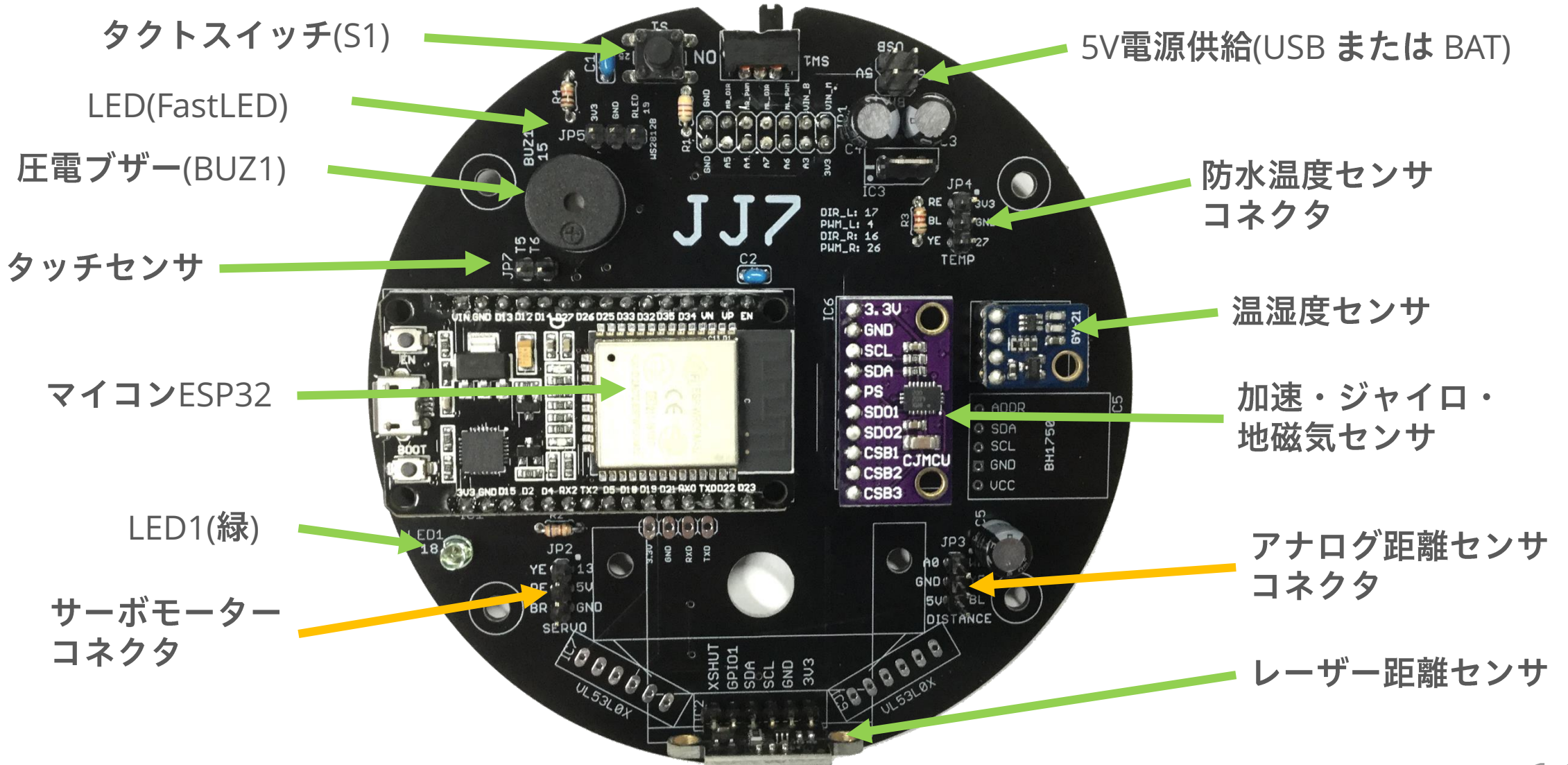


3. M3ナットx4



4. 単3アルカリ電池4本

# マイコンボード(JJ7)





# モーター基板

電源スイッチ(SW2)

モータードライバ

ギア付きDCモーター

キャスター

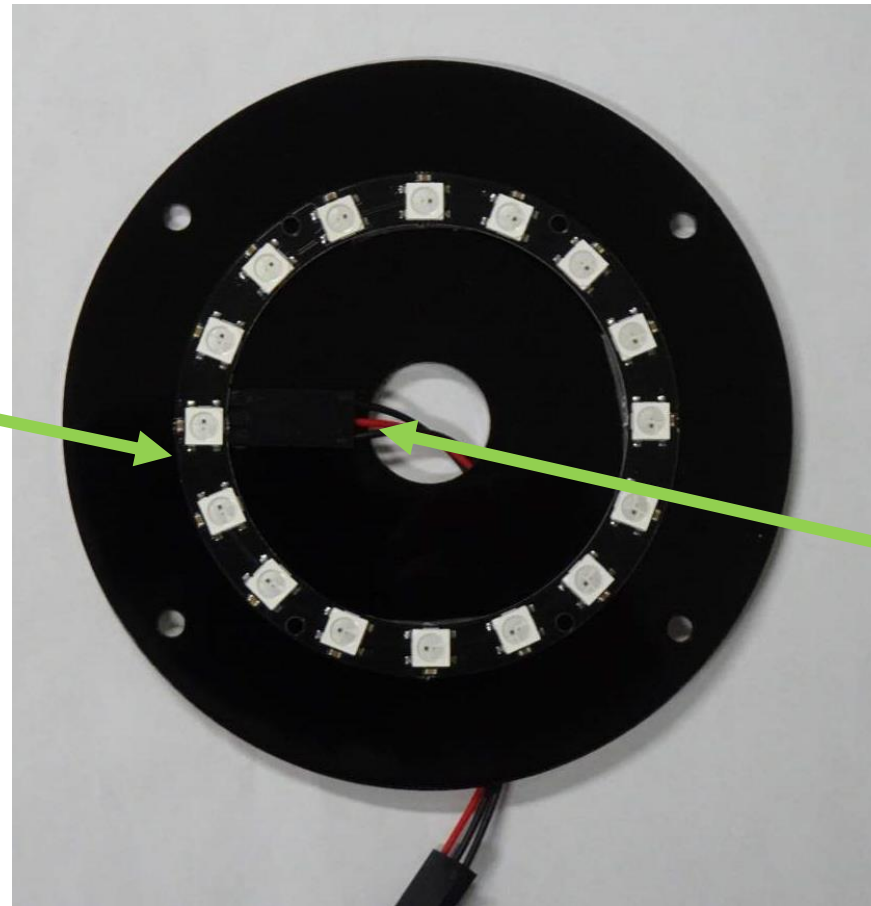
電池ボックス  
(単3x4)

タイヤ(40mm)

フォトリフレクタx5

# LED基板

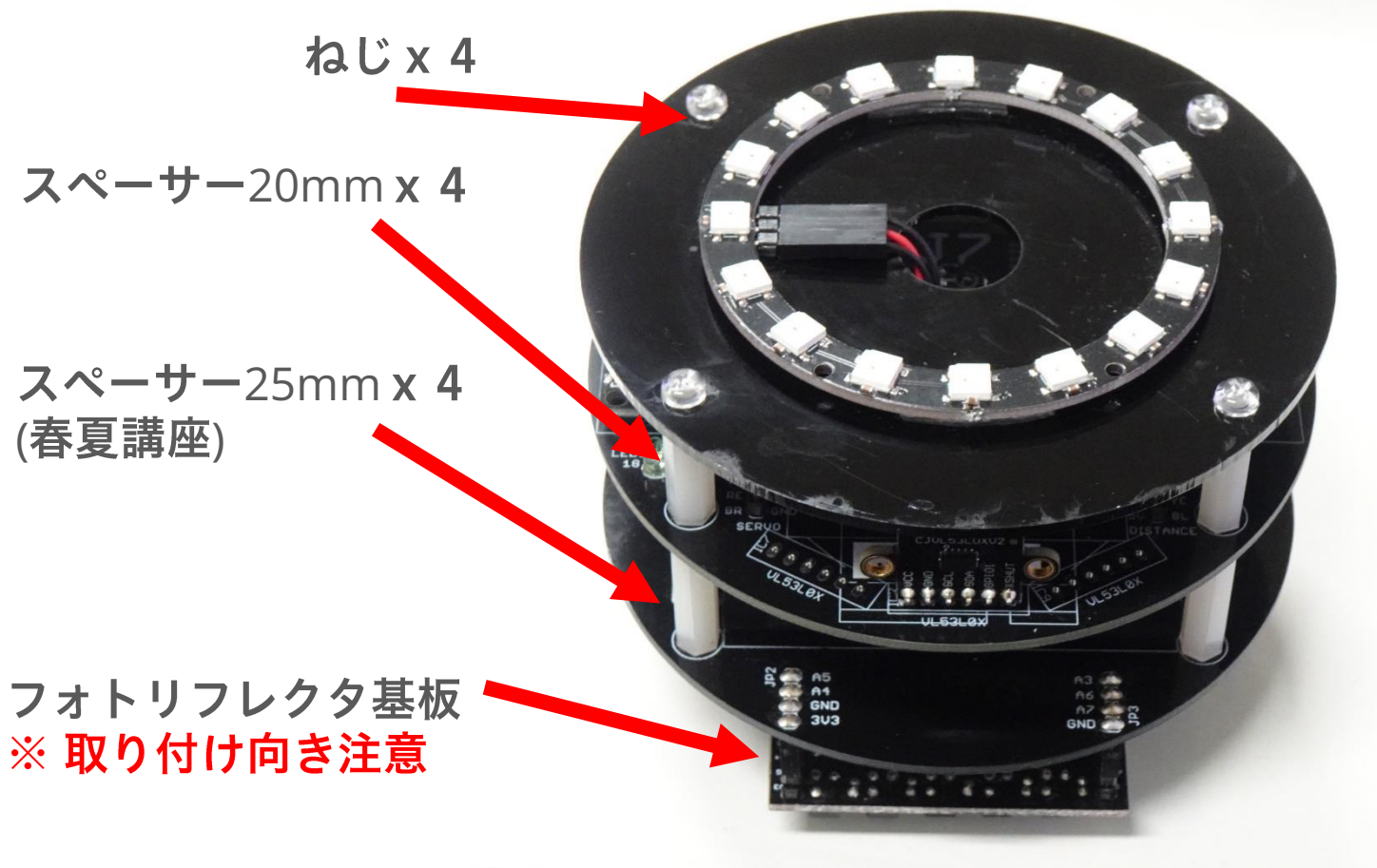
マイコン内蔵LED 16個  
WS2812B



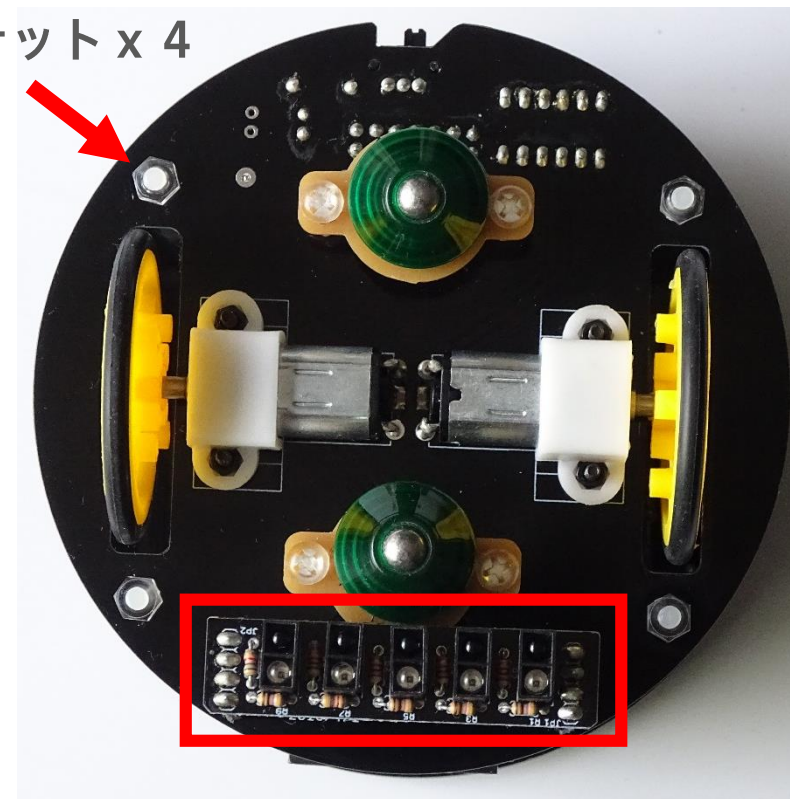
JP5  
赤 : 3V3  
黒 : GND  
黒 : RLED

LED拡張端子へ

# ロボットの組み立て

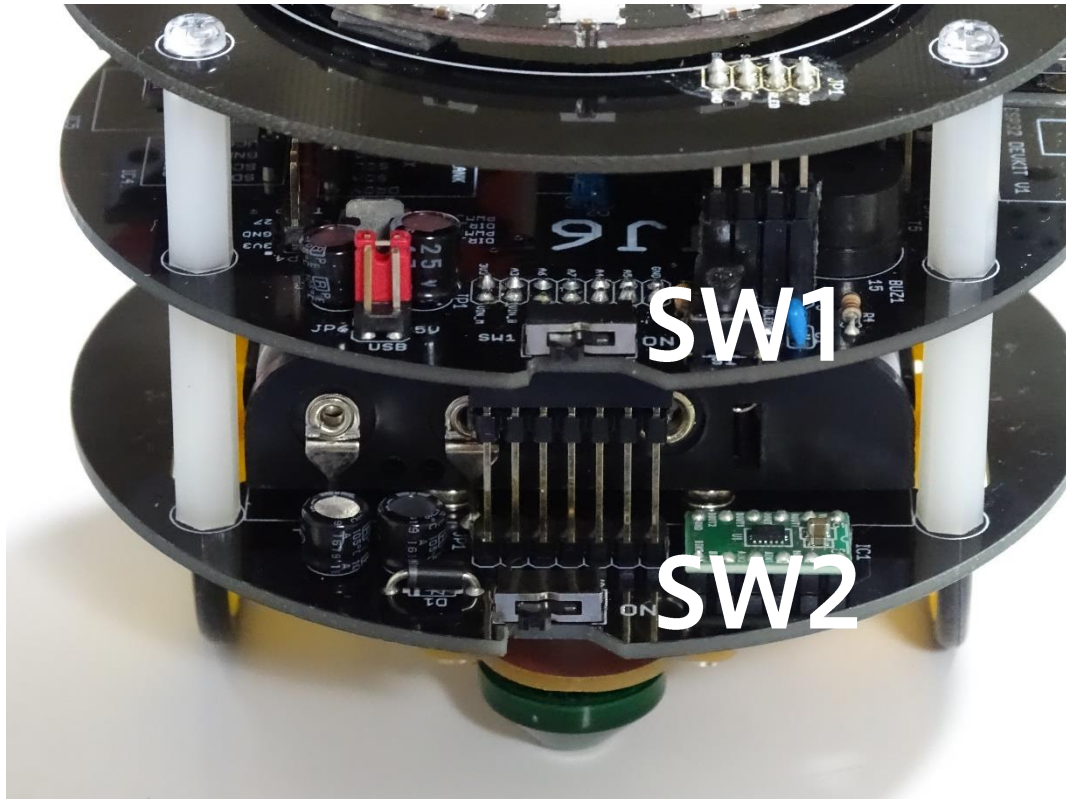


M3ナット x 4



フォトリフレクタ基板  
※ 取り付け向き注意

# 電源スイッチ(SW1, SW2)



プログラムを書き込むとき (USBより電源供給)  
**SW1,SW2の両方をOFF (推奨)**

ロボット単体で動作 (電池より電源供給)  
**SW2をONした後, SW1をON**

- ロボットの動作(プログラム) 停止  
SW1をOFF
- ロボットの再起動  
SW1をOFFした後, SW1をON

ロボットの電源を切る  
**SW1,SW2の両方をOFF**

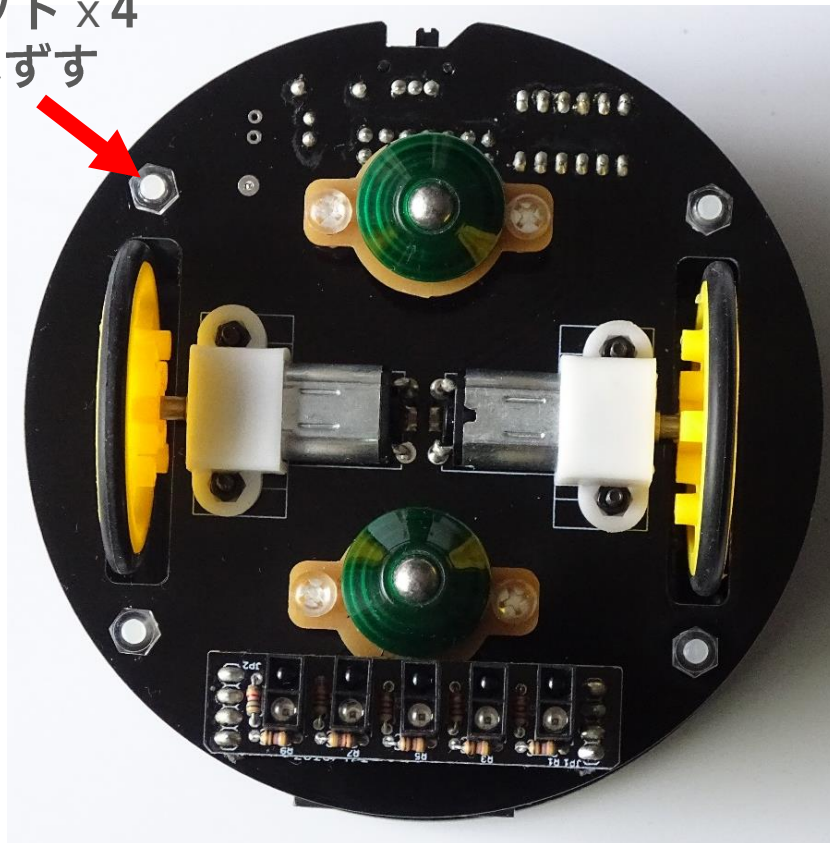
**※スイッチがONになっていると電池が消耗します。**

# 電池の交換方法

## ▶ 電池4本・・・アルカリ乾電池，またはニッケル水素電池(充電電池)

① 電源スイッチを切る

② ナット×4  
をはずす



③ ゆっくりとコネクタをはずす

コネクタを曲げないようにまっすぐに引き抜く

④ 電池を交換する

※ 電池の極性（プラス・マイナス）を間違えないよう  
確認します。

電池の種類：アルカリ電池，ニッケル水素電池が  
使用できます。

繰り返し使える充電式の電池がおすすめです。

例) パナソニック エネループ 充電器セット  
単3形充電電池 4本付き スタンダードモデル  
K-KJ53MCC40

# マイコンの開発環境のセットアップ(初回のみ)

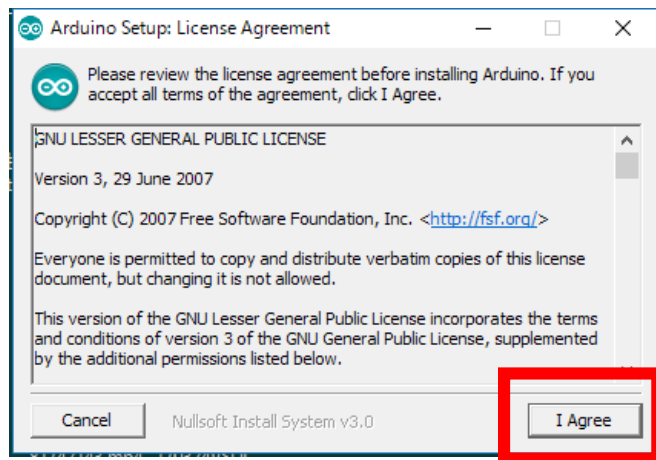
## ▶ USBドライバのインストール

- フォルダ：driver
  - CP210x\_Windows10
  - CP210x\_Windows7\_8
  - CP210x\_WindowsXP
- CP210xVCPInstaller\_x64.exe または
- CP210xVCPInstaller\_x86.exeをダブルクリック
  - x64: 64bit, x86: 32bit システム→バージョン情報, コントロールパネル→システム

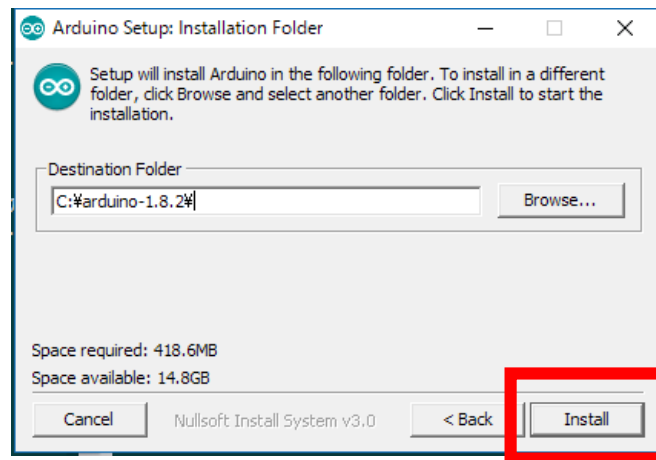
## ▶ Arduinoのインストール

- arduino-1.8.5-windows.exeをダブルクリック

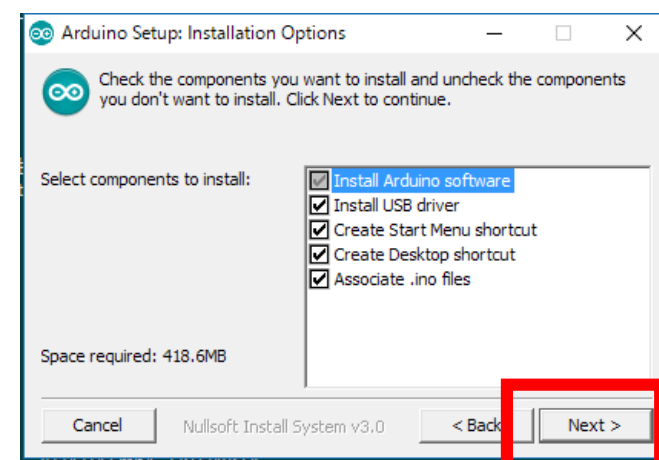
# Arduinoのインストール



ライセンスの同意：I Agree



インストール先：Install



ドライバ等のインストール：Next

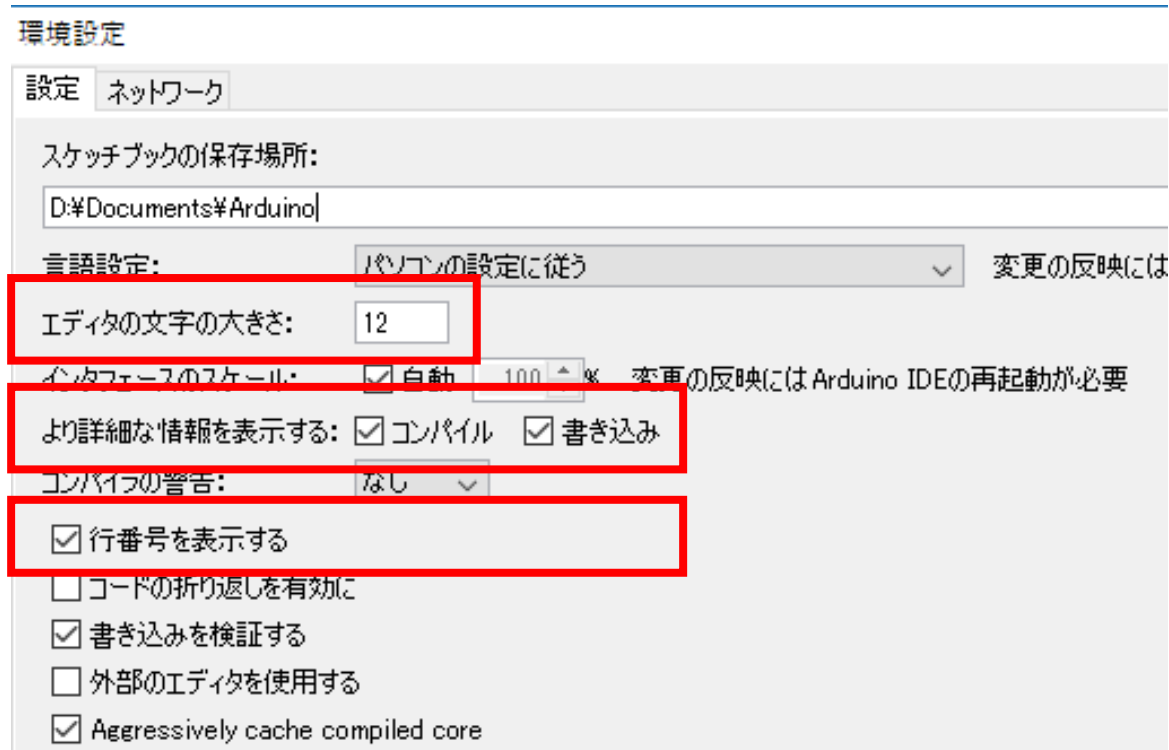


デスクトップにArduinoを  
起動するアイコンができる

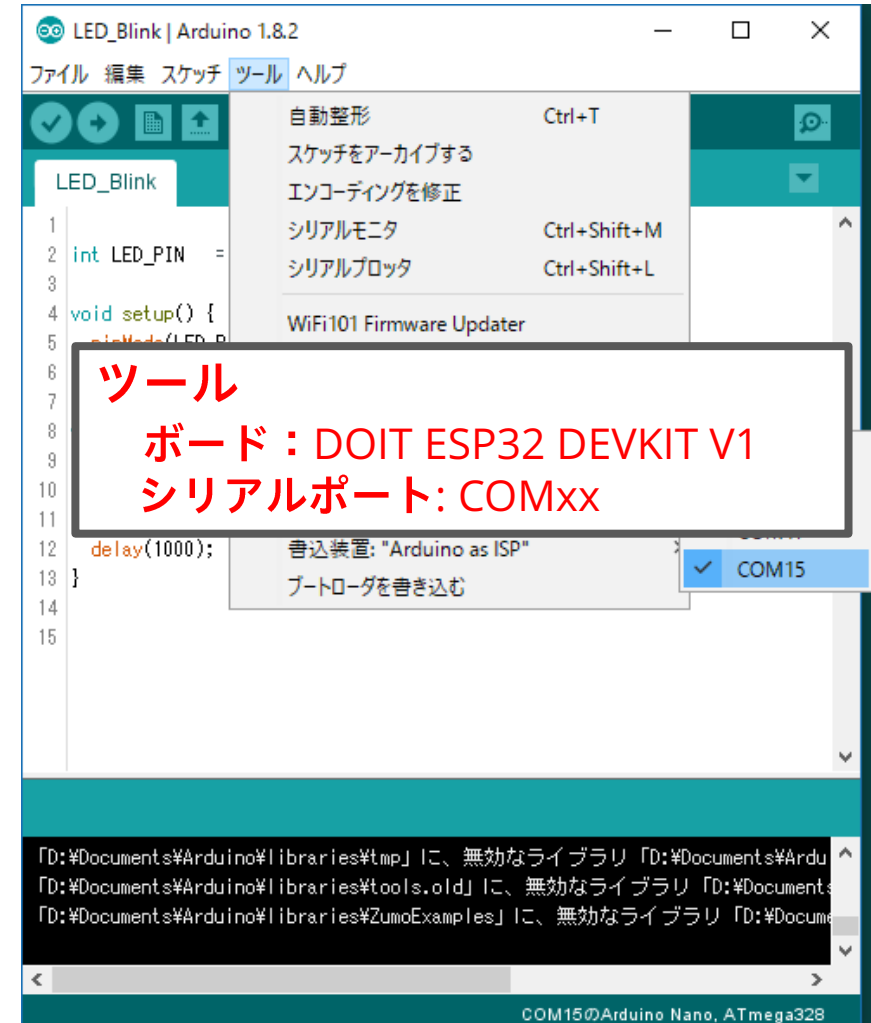
# Arduinoの初期設定

## ファイル→環境設定

- ① エディタの文字の大きさ：見やすい大きさに変更
- ② コンパイル，書き込み：両方にチェック
- ③ 行番号を表示する。：チェック



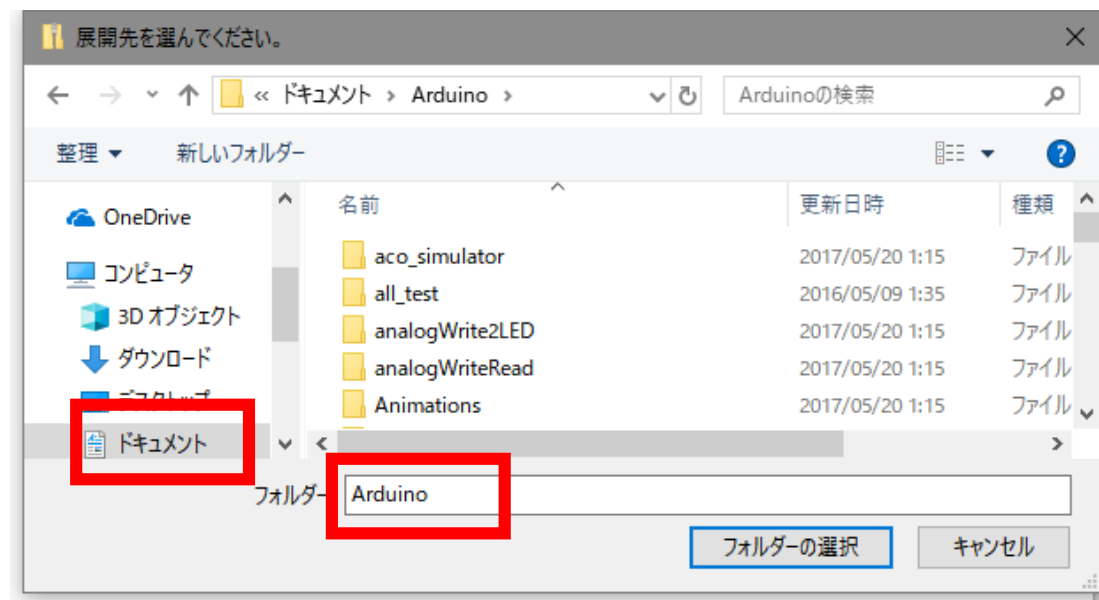
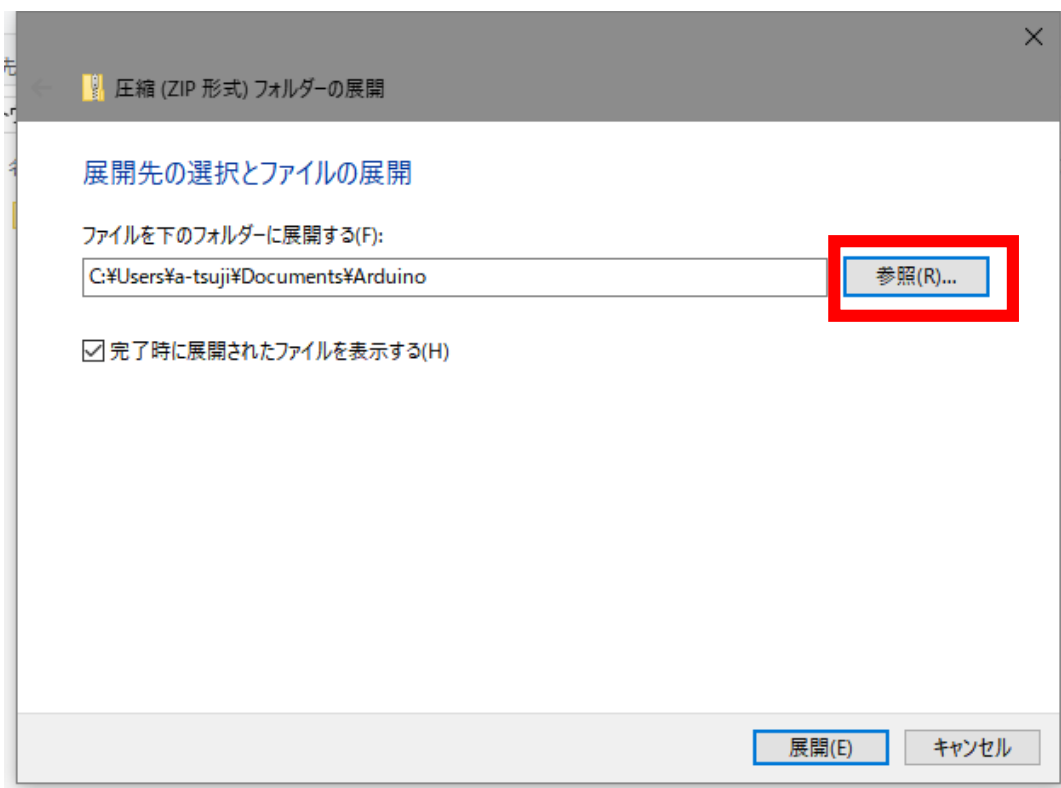
## ④ マイコンボードの設定





# マイコンの開発環境のインストール

- ▶ Arduinoを閉じる
- ▶ フォルダのinstall.zipを右クリックして全て展開
  - 展開先を, **ドキュメント¥Arduino**とする。



# マイコンESP32

- ▶ IoT対応マイコン (Espressif System Ltd.)
- ▶ WiFi無線 (IEEE 802.11 bgn), Bluetooth 4.2
- ▶ デュアルコア Tensilica LX6
- ▶ SRAM: 520KB
- ▶ 電源電圧: 2.2V~3.6V
- ▶ 静電容量タッチインターフェース: x 10
- ▶ 32kHzオシレータ: x 1
- ▶ GPIO x 21, UART x 3, SPI x 2, I2S: x 2 ADC: x 16  
DAC: x 2, I2C: x 2, SDIO: マスター/スレーブ
- ▶ 工事設計認証(技適)番号: 211-161007



# Arduino

## ▶ Arduino(あるでゅいーの)

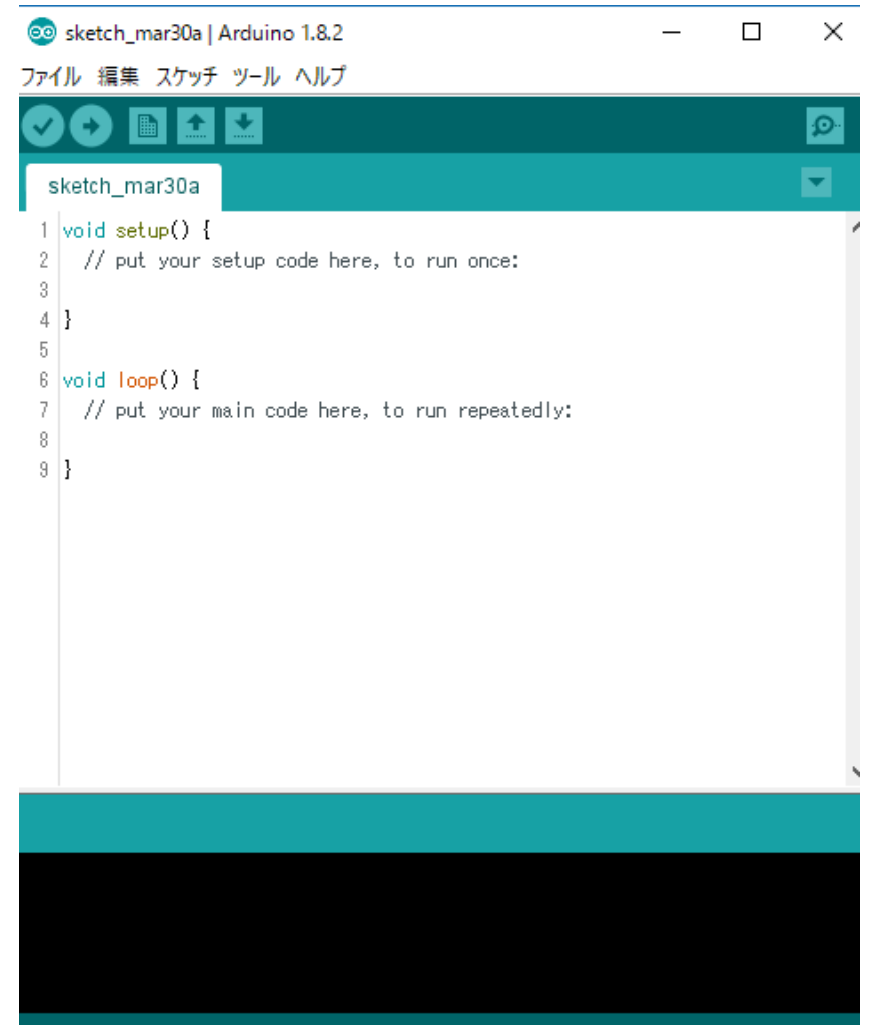
- 初心者でも簡単にマイコンの開発ができる環境
- C/C++言語
- プログラムのことをスケッチという
- プロトタイプ (試作品) 製作
- 開発コミュニティが活発

<https://create.arduino.cc/projecthub>

Arduino UNO



Arduino Mega



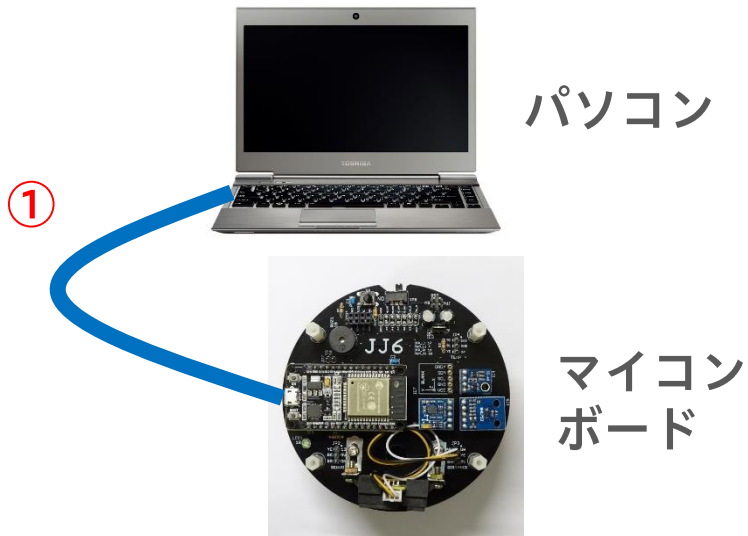
# マイコンのプログラム手順

① パソコンとマイコンボードを  
USBケーブルで接続

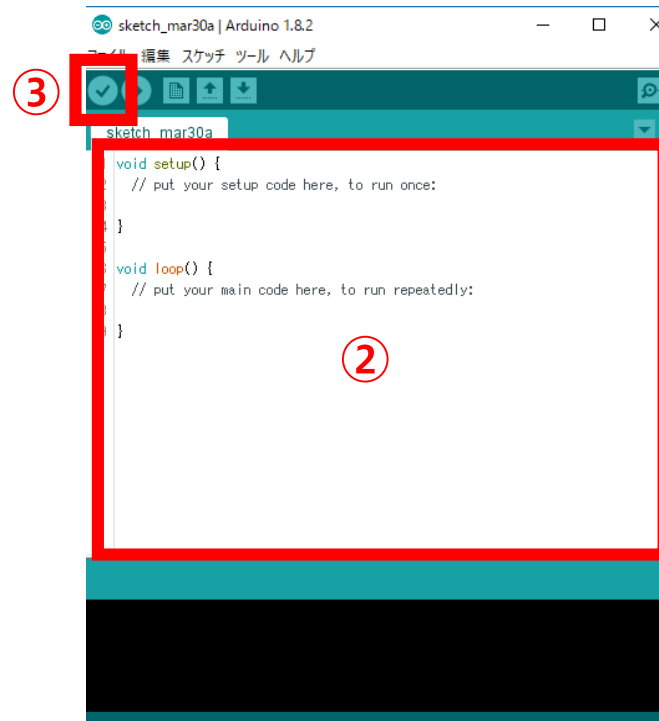
② スケッチ(プログラム)の作成

③ スケッチの検証

④ プログラムの書き込み



マイコンボードの電源LED(赤)が  
点灯していることを確認

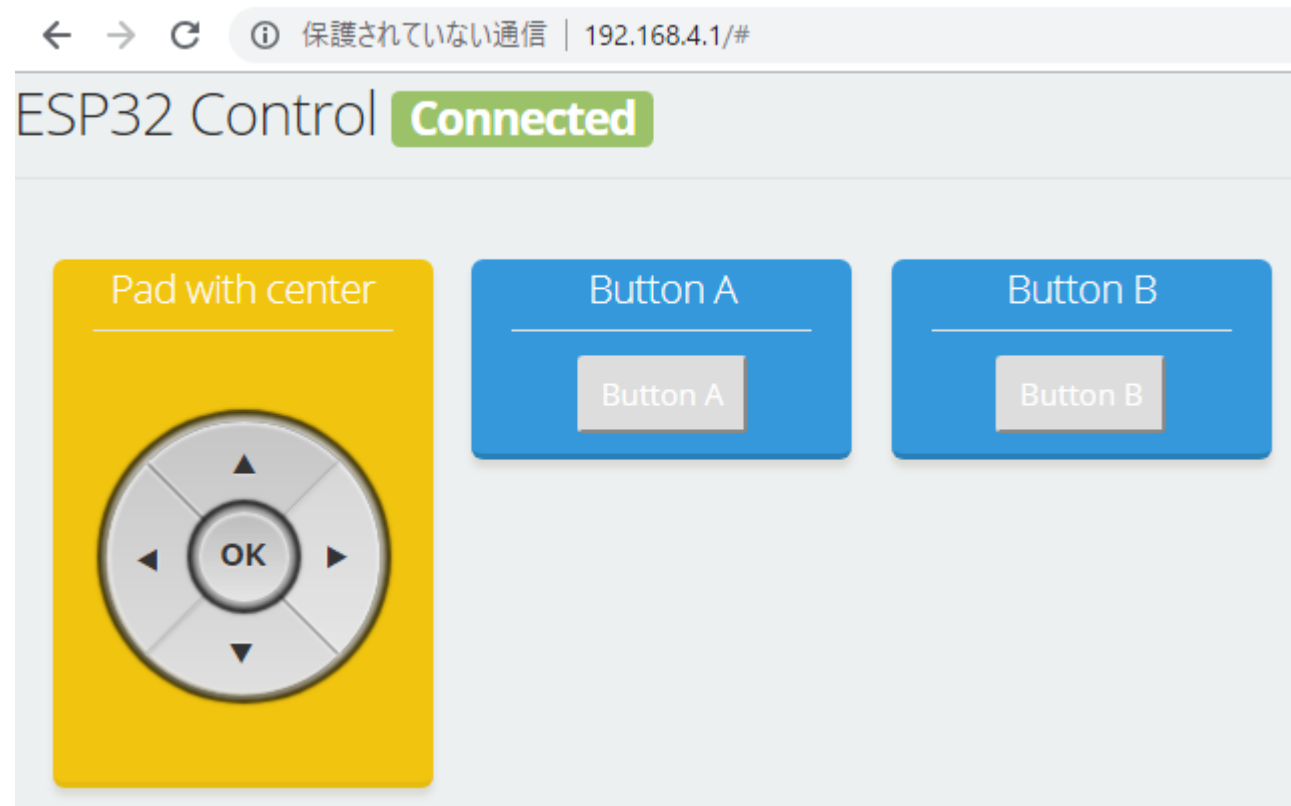


# IoT体験(LED)



# IoT体験(ロボットの動作確認)

- ▶ マイコンボードにUSBケーブルを接続 (電源供給)
- ▶ WiFiの一覧からSSID「**jj7-nn**」を選択して接続 (nn=01,02,・・・)
  - ・ パスワードを入力
- ▶ ブラウザを起動
- ▶ URL欄に「**192.168.4.1**」を入力
- ▶ リモコンのページの表示
  - ・ ロボットを操作



# LEDの点滅 (プログラム)

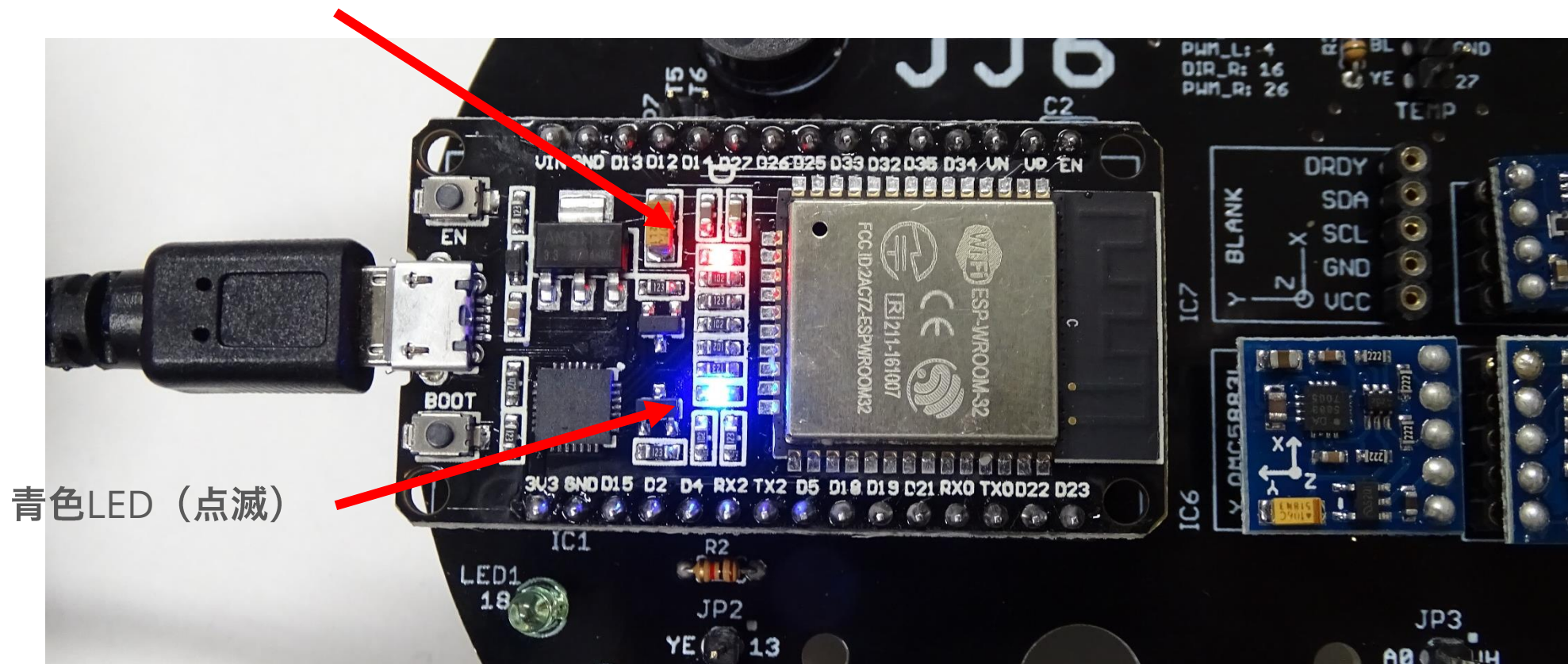
## ▶ Arduinoのメニュー：ファイル→スケッチ例→01.Basics→Blinkを開く

```
24 // the setup function runs once when you press reset or power the board
25 void setup() {
26     // initialize digital pin LED_BUILTIN as an output.
27     pinMode(LED_BUILTIN, OUTPUT);
28 }
29
30 // the loop function runs over and over again forever
31 void loop() {
32     digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
33     delay(1000); // wait for a second
34     digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
35     delay(1000); // wait for a second
36 }
```

# スケッチの実行

- ▶ 青色LEDが1秒毎に点灯，消灯を繰り返す

赤色LED（電源）



青色LED（点滅）



# Ex0101 | LEDの点滅(1秒ごと)

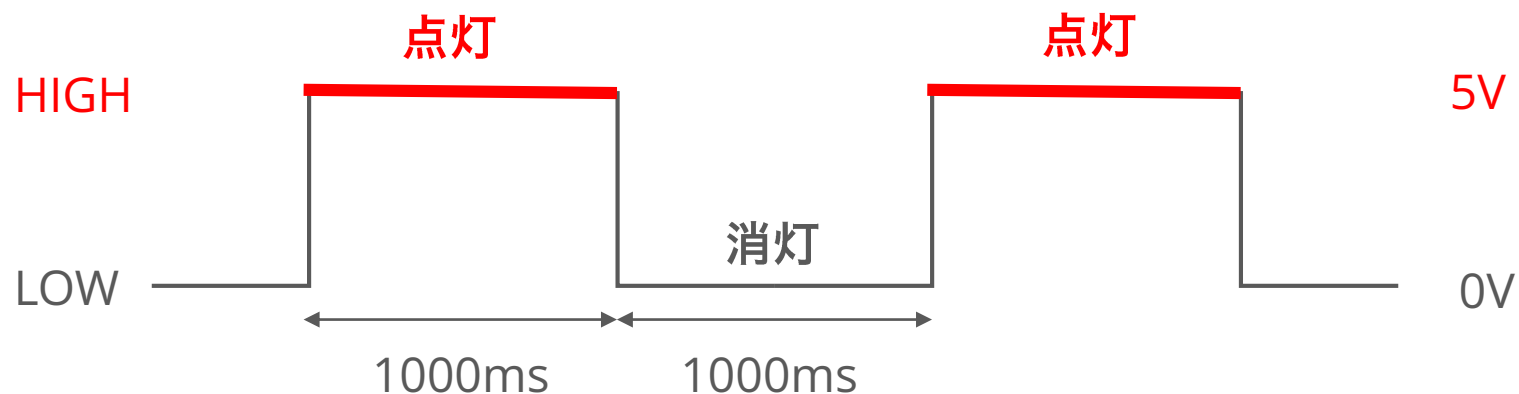
```
// Ex0101 LEDの点滅(1秒毎)
const int LED_PIN = 2;

void setup() {
  pinMode(LED_PIN, OUTPUT); // 出力
}

void loop() {
  digitalWrite(LED_PIN, HIGH); //点灯
  delay(1000); // 1000ms待ち
  digitalWrite(LED_PIN, LOW); //消灯
  delay(1000); // 1000ms待ち
}
```

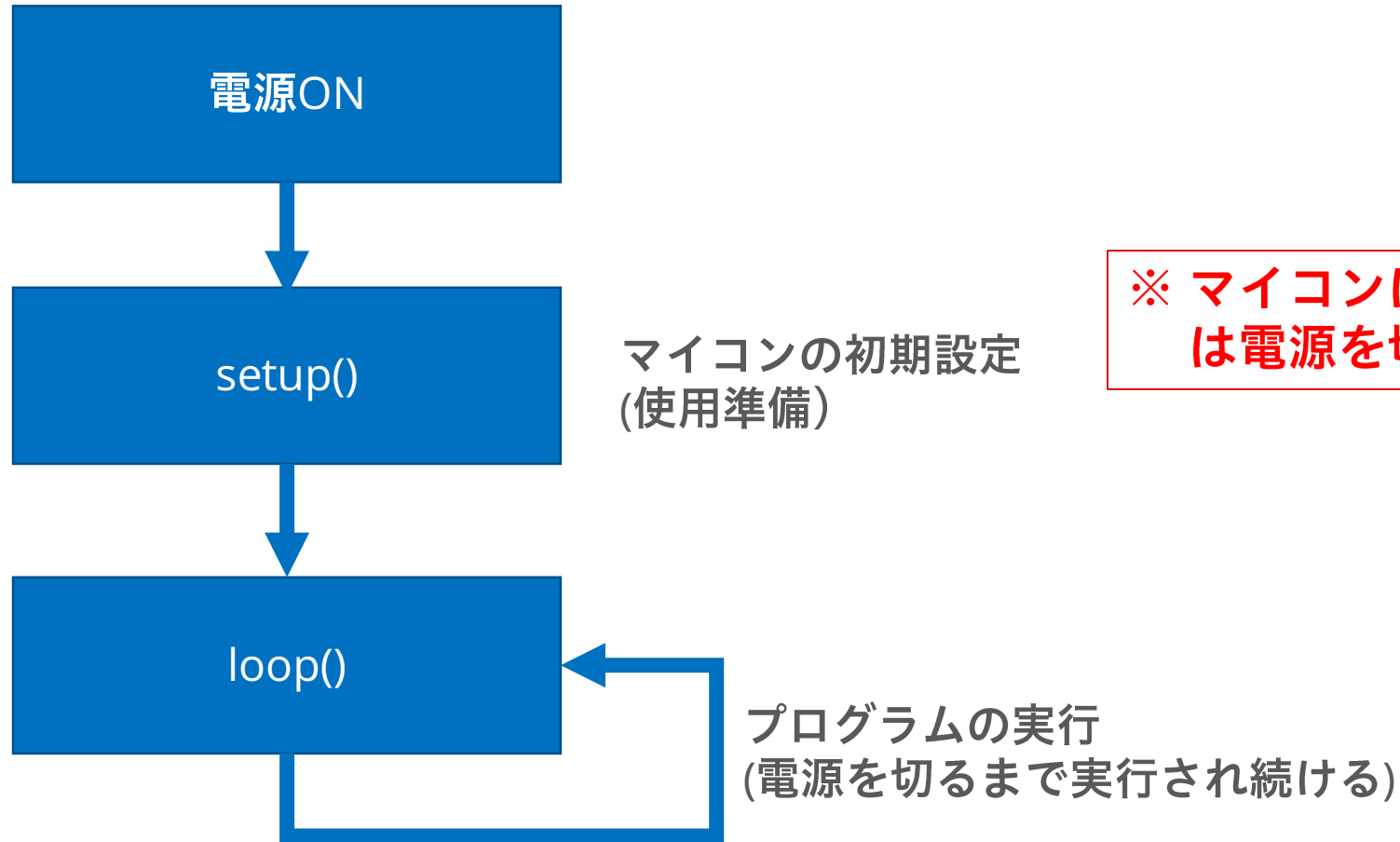
## スケッチの各関数の意味

- ▶ `setup()` . . . マイコンの初期設定(一度だけ実行)
- ▶ `loop()` . . . 無限ループ(繰り返し実行)
- ▶ `pinMode` . . . ポートの入出力設定
- ▶ `digitalWrite` . . . ポートのHIGH,LOW出力
- ▶ `delay` . . . ms単位の遅延(待ち)
  - 1000 ms = 1 s



# プログラムの実行

- ▶ フローチャート・・・プログラムの流れを説明した図



## 課題(Ex01)

- Ex0101をもとに、次のようにLEDを点滅させてみましょう。

Ex0102: 青色LED(2)の点滅間隔を500ms毎にON, OFFを繰り返す

Ex0102a: 青色LED(2)を1000msON,1000msOFFを3回繰り返した後,  
500msON,500msOFFを3回繰り返す

Ex0102b: Ex0102aを forループを使う

Ex0103: 緑色LED(18)の点滅間隔を1000ms毎にON, OFFを繰り返す

Ex0103a: 緑色LED(18)の点滅間隔を500ms毎にON, OFFを繰り返す

Ex0103b: 緑色LED(18)を1000msON,1000msOFFを3回繰り返した後,  
500msON,500msOFFを3回繰り返す

Ex0103c: Ex0103bを forループを使う

Ex0104: 青色LED(2)と緑色LED(18)の点滅間隔を1000ms毎にON, OFFを繰り返す

## ▶ 講座資料(スライド, サンプルスケッチ等)

- <http://cms.db.tokushima-u.ac.jp/DAV/person/S10767/Share/R01Robot>
- スケッチ例の使い方
  - Example01.zipをダウンロード
  - ファイルを右クリック: 「すべて展開」
  - 参照: 「ドキュメント¥Arduino」を選択して「展開」

← 圧縮 (ZIP 形式) フォルダの展開

展開先の選択とファイルの展開

ファイルを下のフォルダに展開する(F):

C:\Users#a-tsuji\Documents

参照(R)...

完了時に展開されたファイルを表示する(H)

展開(E)

キャンセル

The screenshot shows a web browser window with a yellow header bar. The breadcrumb path is "CMS > IDX > 徳島大学 > 工学部 > 知能情報工学科 > 辻 明典 > Share". The user is identified as "【個人】 辻 明典 / Tuji, Akinori" with a sub-path "(Share/H29Sensor)". A red-bordered notification box contains the text: "《EDB/CMSからの重要なお知らせ》 (August 30, 2015更新) お知らせを見る". Below this is a "コンテンツエリア" section with a table of links for different user types. At the bottom, a "コンテンツ" table lists "Parent Directory" and "Lecture01/".

場所の情報とサーバ切替 (ヘルプ)	
一般閲覧用	: <a href="http://cms.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/">http://cms.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/</a>
学生閲覧用 (学内)	: <a href="http://cms-ldap.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/">http://cms-ldap.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/</a> <a href="#">切替</a>
学生閲覧用 (学外)	: <a href="https://cms-ldap.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/">https://cms-ldap.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/</a> <a href="#">切替</a>
教職員閲覧・登録 (ID&Pass)	: <a href="https://cms.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/">https://cms.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/</a> <a href="#">切替</a>
教職員閲覧・登録 (EDB/PKI)	: <a href="https://cms-pki.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/">https://cms-pki.db.tokushima-u.ac.jp/DAV/person/S10767/Share/H29Sensor/</a> <a href="#">切替</a>

Name	Last modified	Size	Description
Parent Directory		-	このフォルダはインターネットからアクセス可能です。 (→詳細)
Lecture01/	15-May-2017 12:44	-	

## 付録：開発環境リンク

### ▶ Arduino

- Arduino-1.8.10 : <http://www.arduino.cc/>

### ▶ ESP32開発環境

- <https://github.com/espressif/arduino-esp32>

### ▶ USBドライバ

- ダウンロード先
  - CP210x: <https://jp.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>

### ▶ ライブラリ

- DallasTemperature: 温度センサ用
  - <http://www.arduino-libraries.info/libraries/dallas-temperature>
- SparkFun\_HTU21D: 温湿度センサ用
  - [https://github.com/sparkfun/SparkFun\\_HTU21D\\_Breakout\\_Arduino\\_Library](https://github.com/sparkfun/SparkFun_HTU21D_Breakout_Arduino_Library)
- OneWire: 1線通信用
  - <http://www.arduino-libraries.info/libraries/one-wire>