

2014年7月28日:草稿
2014年8月17日:初稿

ロボットをつくろうー前編ー

第5回 シリアル通信
2014年8月23日(土)10時～

徳島大学大学院ソシオテクノサイエンス研究部
技術専門職員 辻 明典

連絡先：

770-8506 徳島市南常三島町2-1

TEL/FAX：088-656-7485

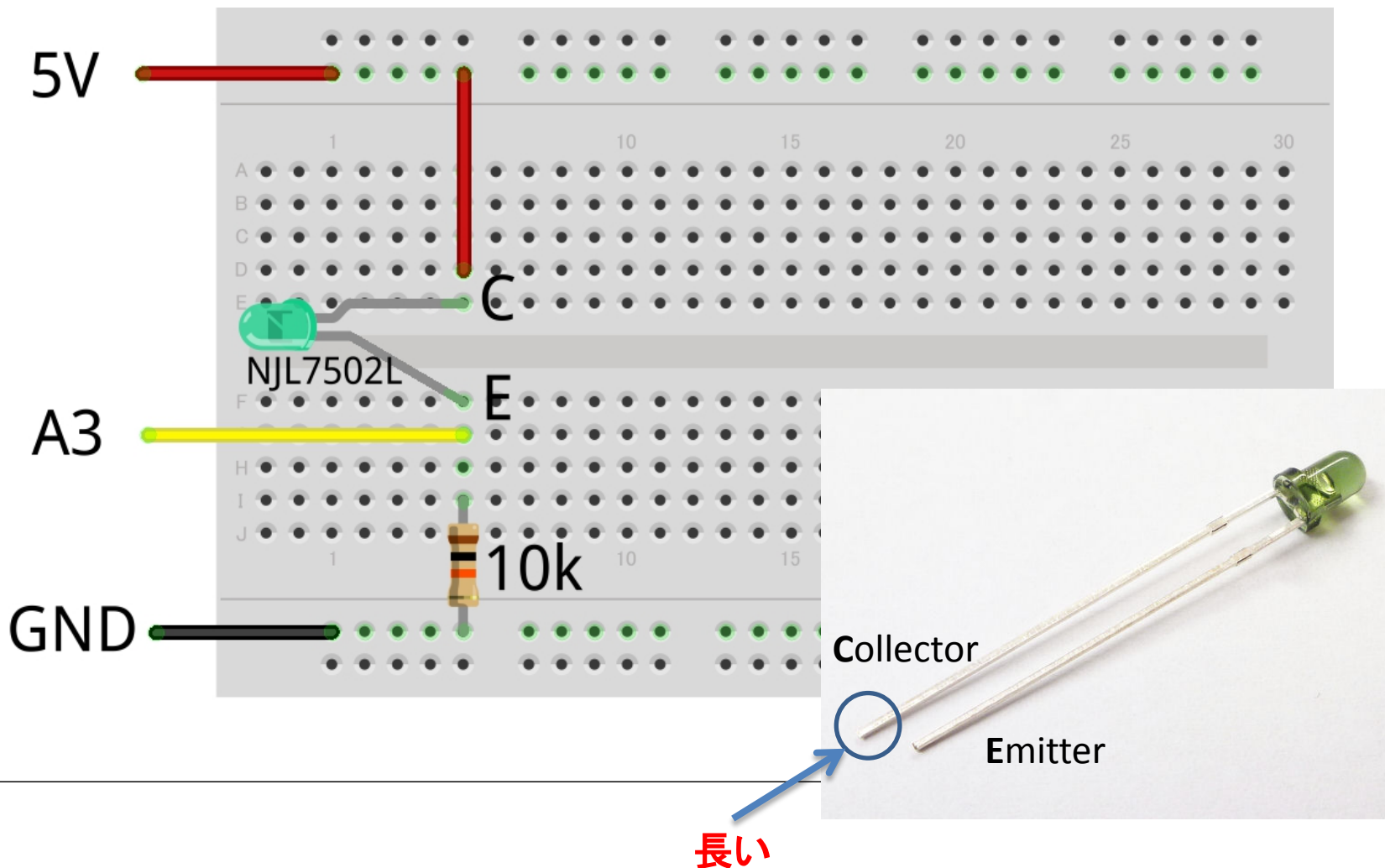
E-mail: : a-tsuji@is.tokushima-u.ac.jp

本日の予定

- 1 ブレッドボード上に回路作成
 - 光センサ
- 2 JJマイコンボード
 - 接続表
- 3 シリアル通信
- 4 シリアル通信のプログラム
 - スイッチの状態を調べる
 - 光の強さを調べる
- 5 シリアル通信の応用
 - 暗くなったらLEDを光らせる
 - スイッチを押してブザーを鳴らす

1 ブレッドボード上に回路作成

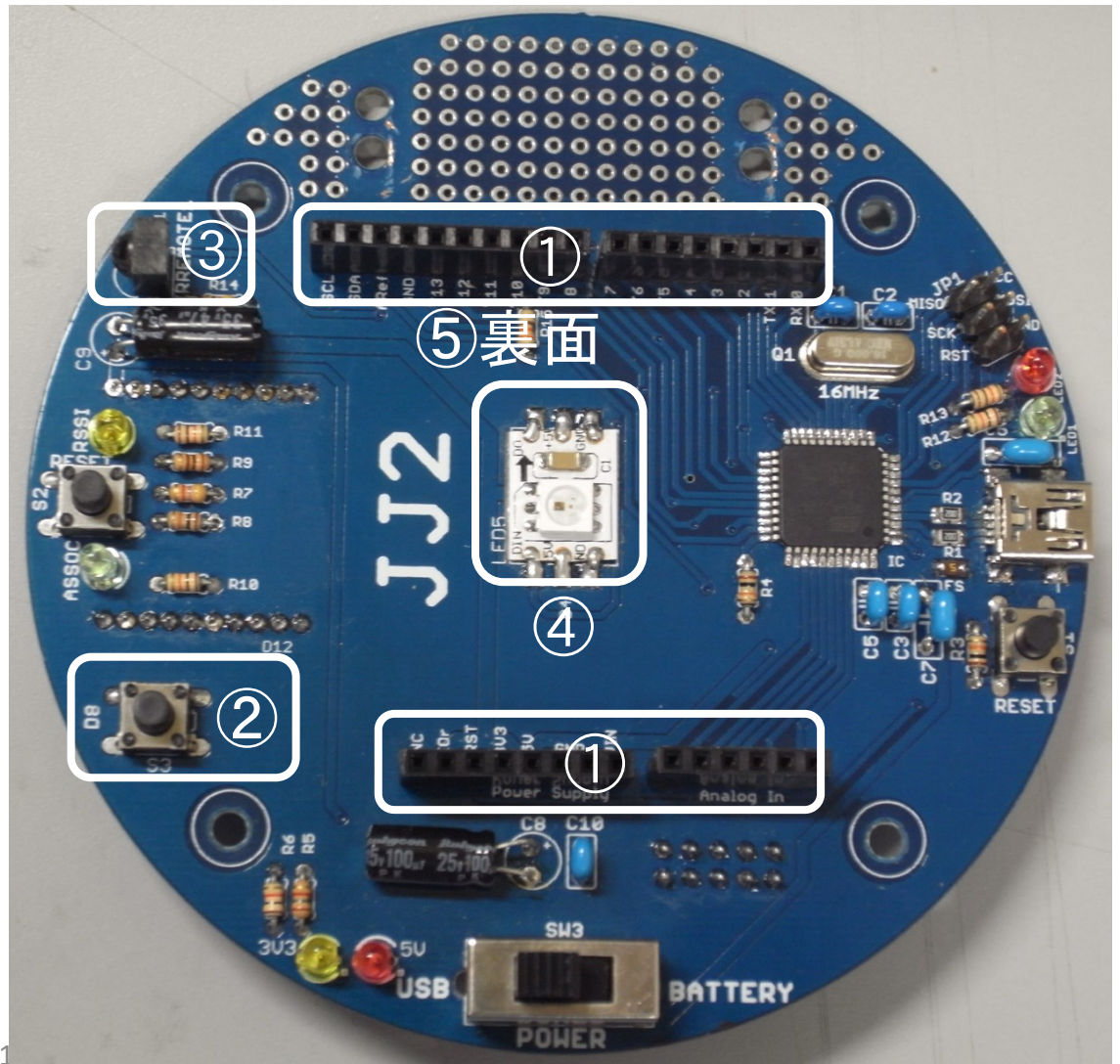
1.1 光センサの回路（準備）

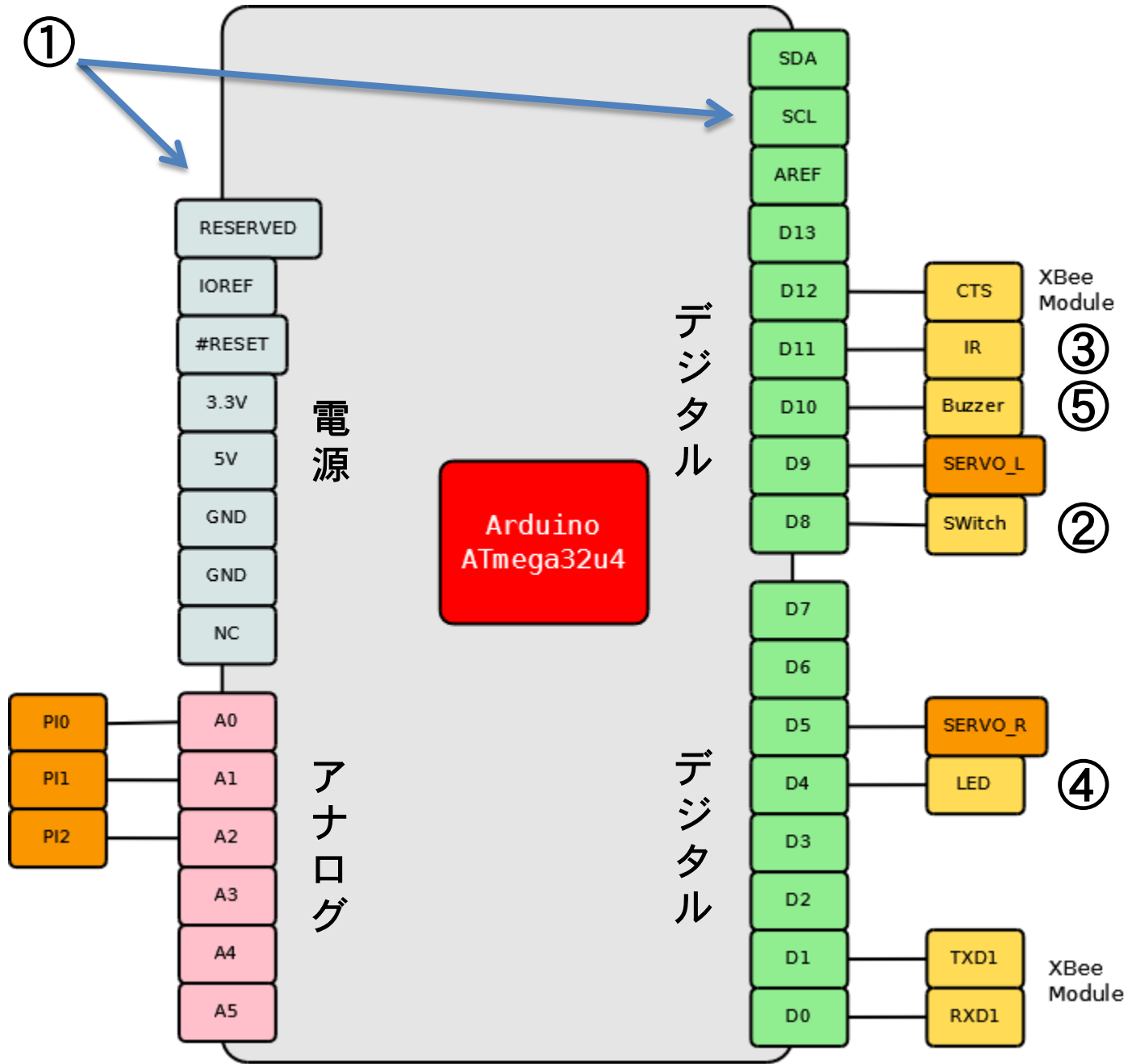


2 JJマイコンボード

2.1 JJマイコンボード

- ① ピンソケット
(Arduino互換)
- ② スイッチ
- ③ リモコン
(赤外線受信センサ)
- ④ フルカラーLED
- ⑤ ブザー
(圧電素子)





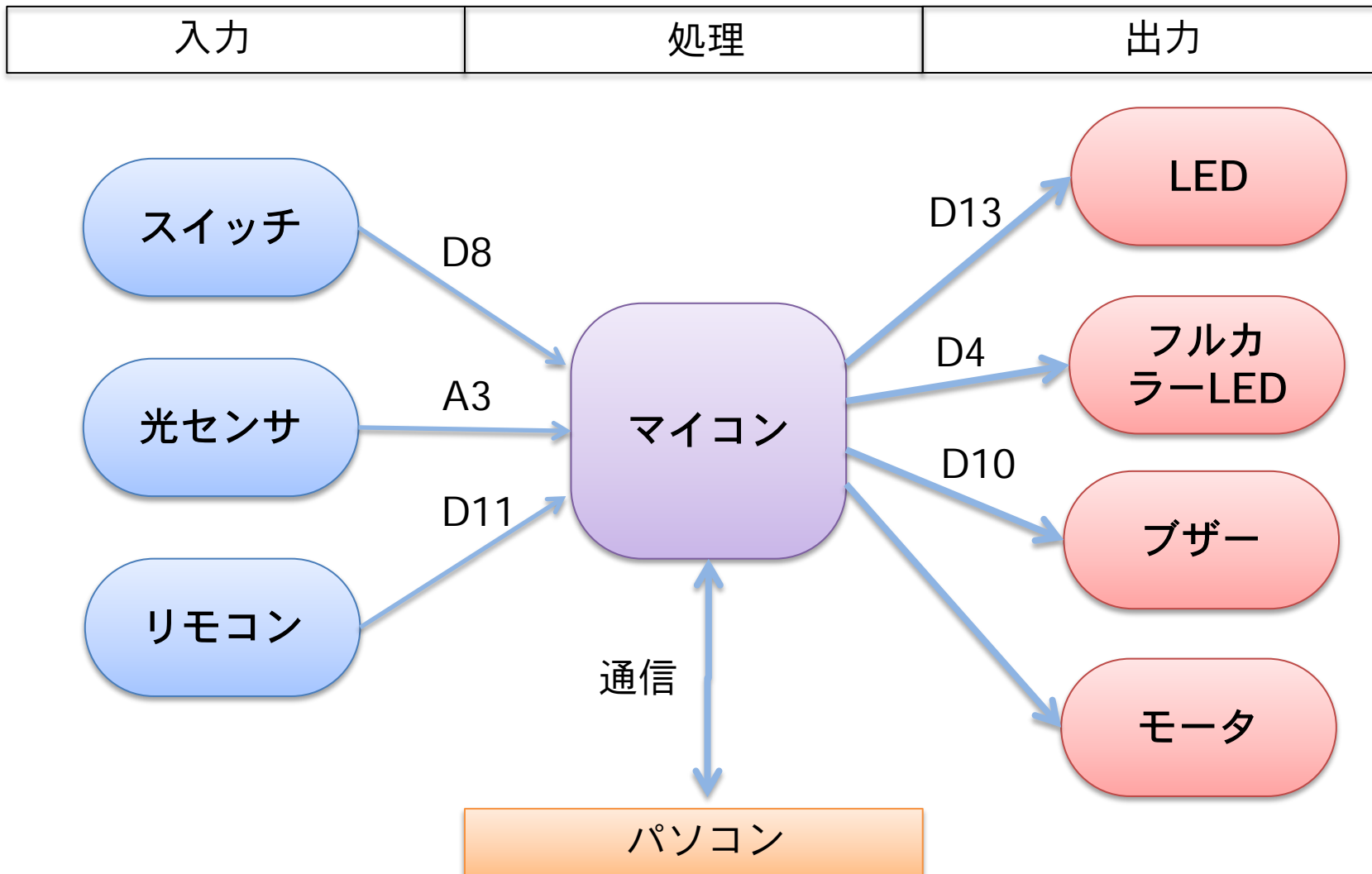
2.2 JJマイコンボードの接続表

Arduinoピン	機能	注釈
4	フルカラーLED	FastLEDライブラリ使用
8	スイッチ	digitalRead使用
10	ブザー	Toneライブラリ使用
11	リモコン	IRremoteライブラリ使用

ブレッドボードに追加

A3	光センサ	AnalogRead使用
----	------	--------------

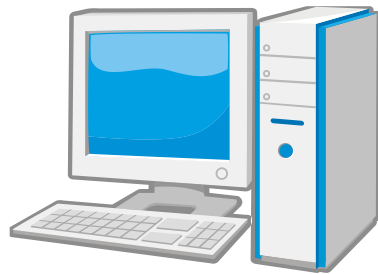
2.3 マイコンの入出力関係



3 シリアル通信

3.1 シリアル通信

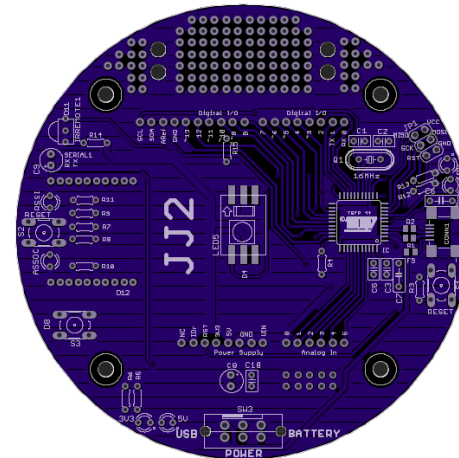
- ◆ パソコンとマイコン間の通信
→データをやりとり



パソコン



シリアル通信



マイコン

- ◆ プロトコル：あらかじめ決めておくこと
通信方式，通信速度，通信方向，データ等

3.2 シリアル通信

◆ シリアル通信



◆ 通信速度

bps : bit per second 1秒間に送れるデータ量

◆ データの形式

8ビット (1バイト) のデータ, スタート・ストップビットで構成

◆ 送受信ポート

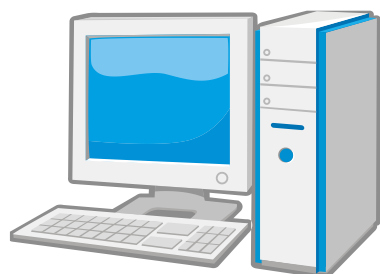
RX : Receiver 受信

TX : Transmitter 送信

4 シリアル通信のプログラム

4.1 スイッチの状態を調べる

JJのスイッチが押されたかどうか？



パソコン



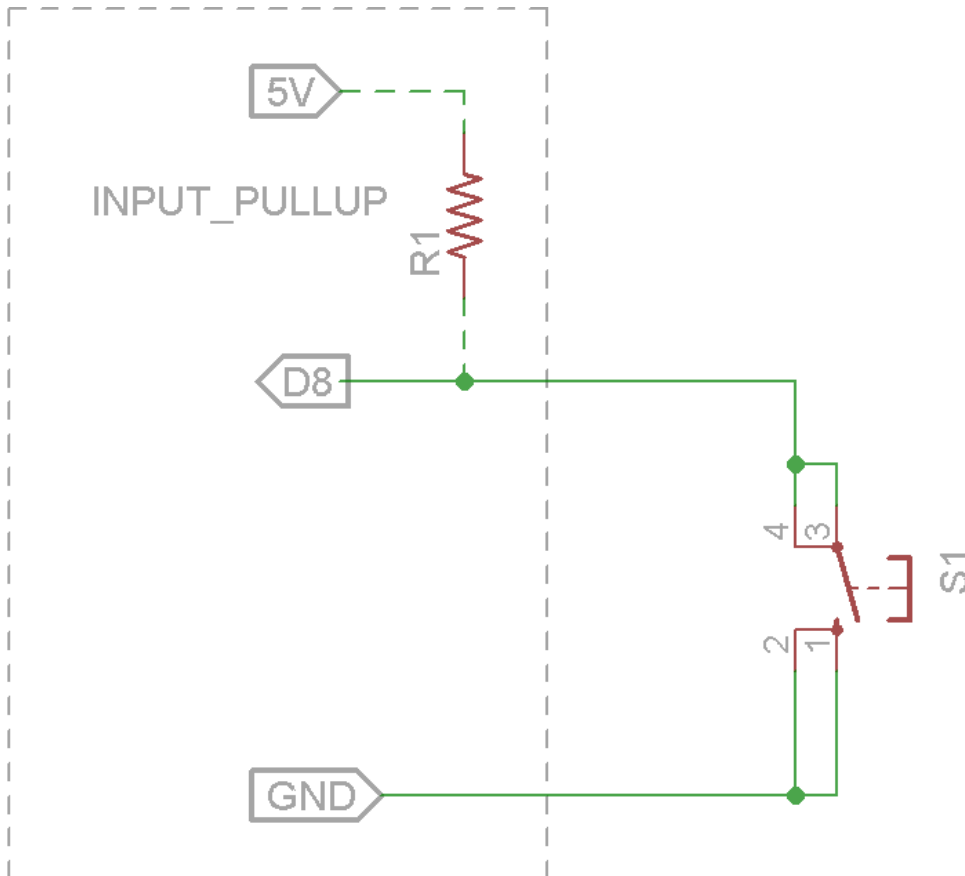
シリアル通信
9600 bps



マイコン
スイッチ (D8)

スイッチの状態

JJマイコン

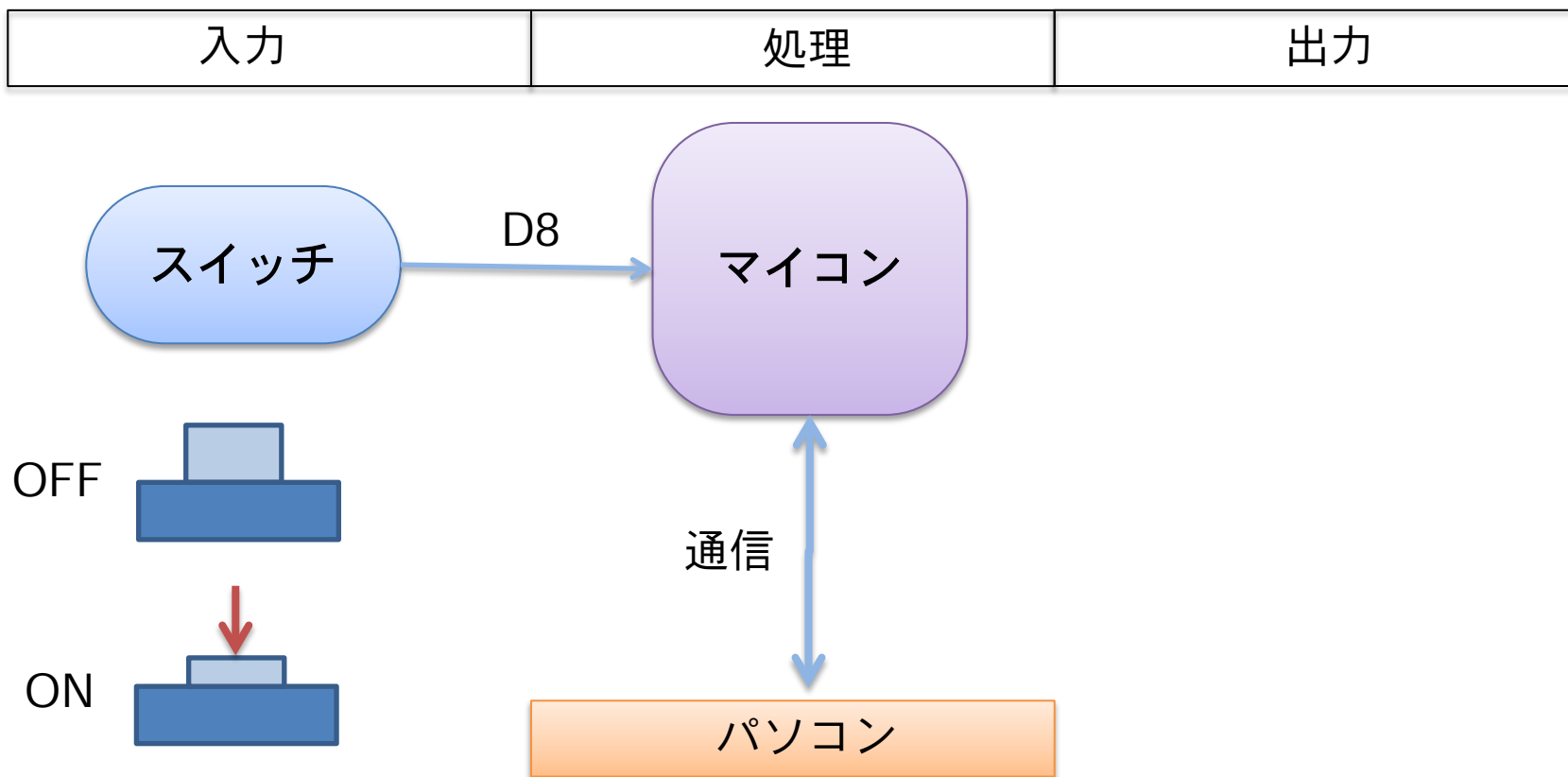


スイッチの状態	D8
ON	0 (LOW)
OFF	1 (HIGH)

デジタル：8番ポート
スケッチにて，INPUT_PULLUPを
定義

スイッチの接続関係

- ◆ スwitchの状態を調べる.



スイッチの状態を調べる Ex501A

```
const int SwPin = 8;
```

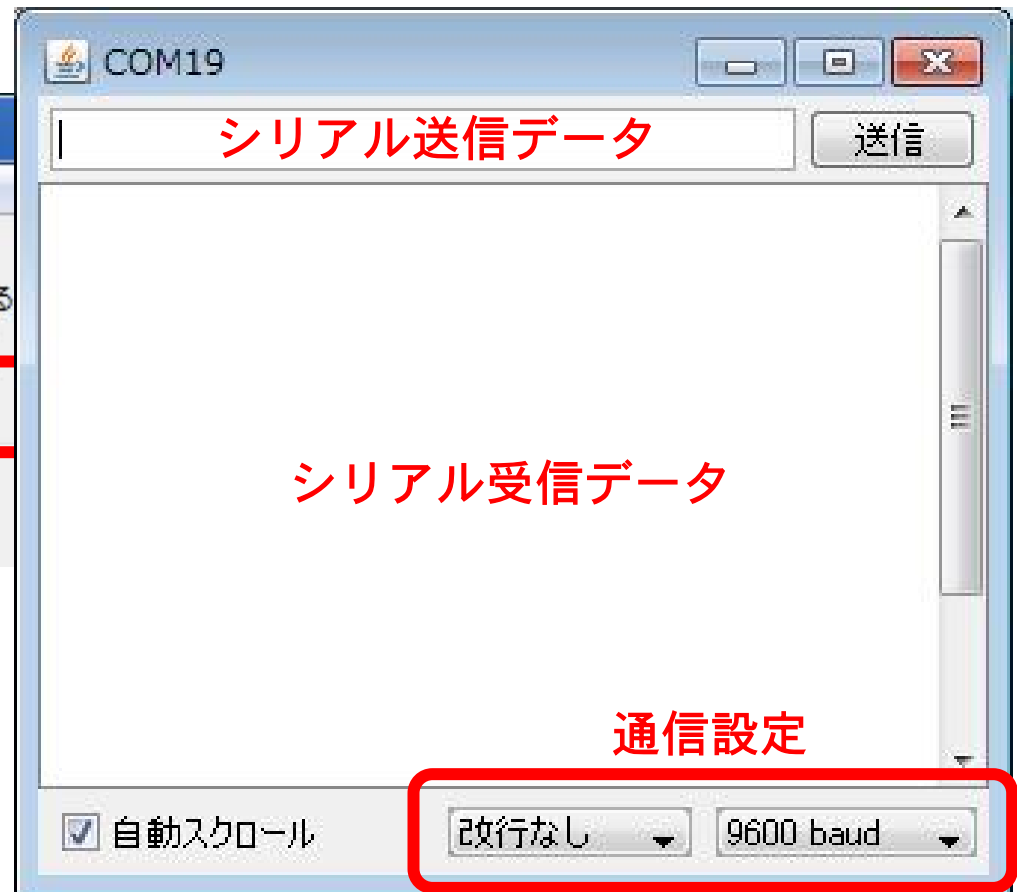
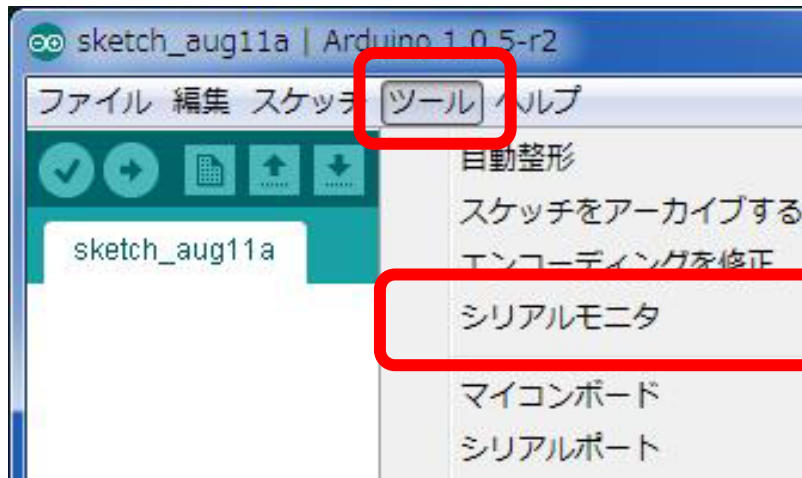
```
void setup() {  
    pinMode(SwPin, INPUT_PULLUP); // 8番を入力に設定  
    Serial.begin(9600); // 通信速度設定  
}
```

```
void loop() {  
    int val = digitalRead(SwPin); // スwitchの状態読み込み  
    Serial.println(val); // パソコンに送信  
    delay(200);  
}
```

シリアルモニタ

シリアルモニタの起動

- 1) ツール→シリアルモニタ
- 2) 通信状況をモニタ

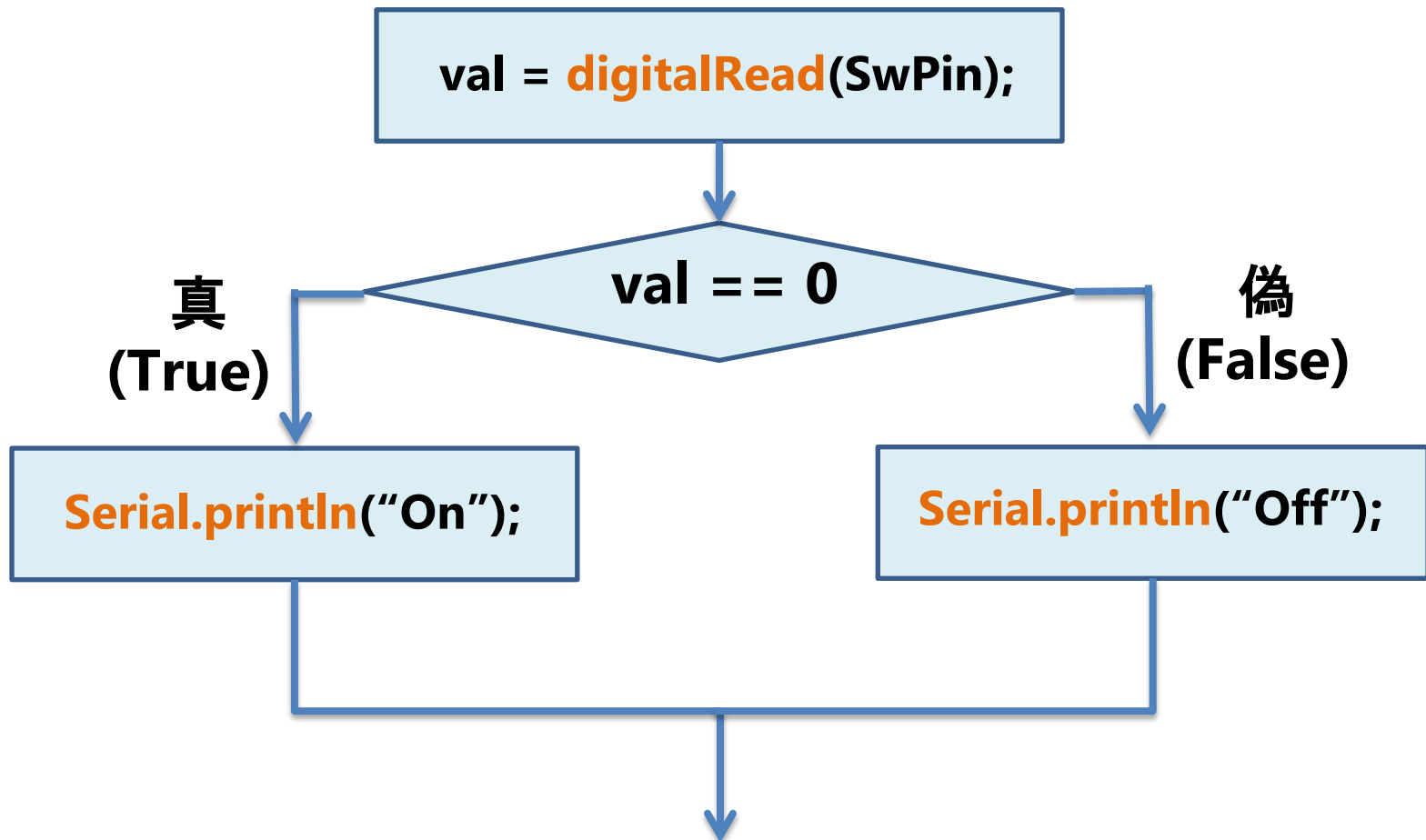


スイッチの状態を調べる(if文) Ex502A

```
const int SwPin = 8;  
void setup() {  
    pinMode(SwPin, INPUT_PULLUP);  
    Serial.begin(9600);  
}  
void loop() {  
    int val = digitalRead(SwPin);  
    if (val == 0) {  
        Serial.println("On"); // スイッチがONのとき“ON”と送信  
    } else {  
        Serial.println("Off"); // スイッチがOFFのとき“OFF”と送信  
    }  
    delay(200);  
}
```

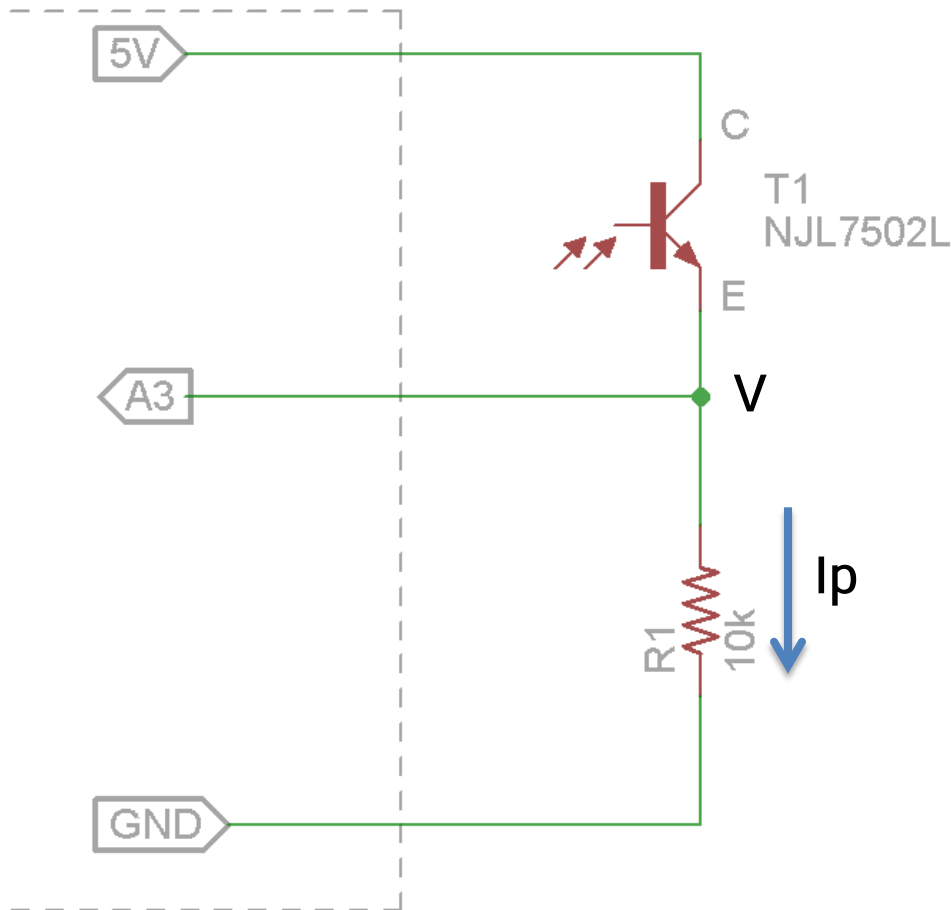
if文

if文: 変数が条件を満たすかどうか



4.4 光の強さを調べる

JJ2



光センサの状態	A3
光の強さ	0 ~ 1023

アナログ：3番ポート

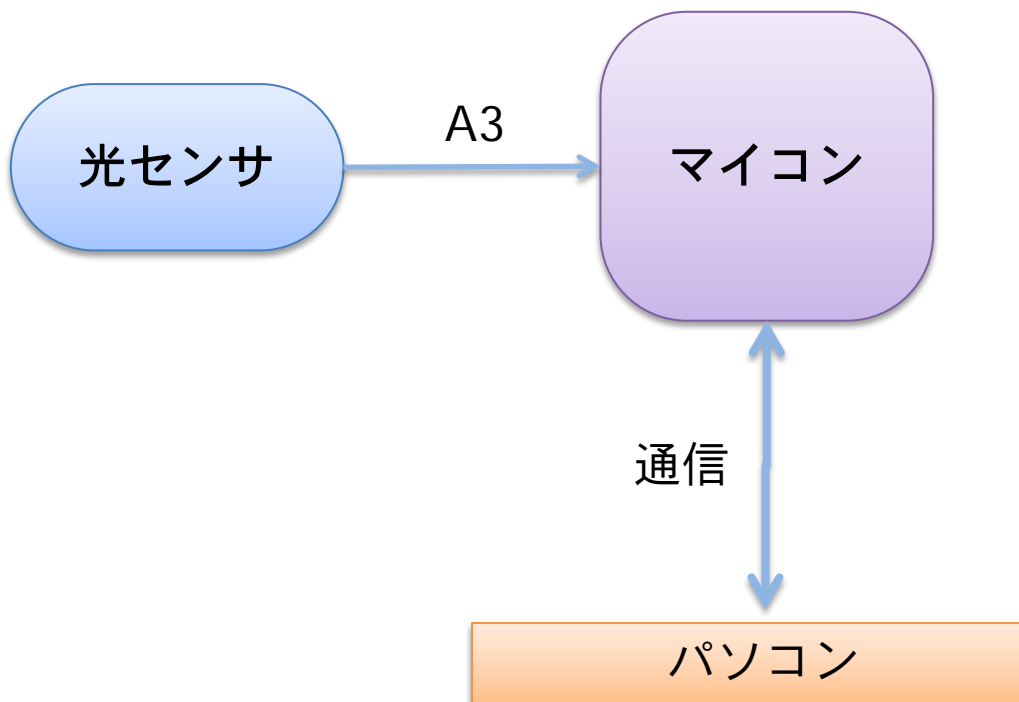
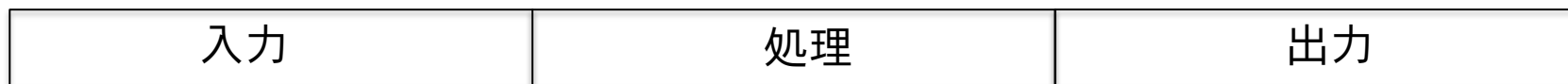
センサに光があたると電流 I_p が流れる

$$V = R1 \times I_p \text{ [V]}$$

値：光が強い→値が大きい
光が弱い→値が小さい

光センサの接続関係

- ◆ 光の強さを調べる.



光の強さを調べる Ex503A

```
const int lightPin = A3;
```

```
void setup() {  
  Serial.begin(9600);  
}
```

```
void loop() {  
  int val = analogRead(lightPin); // センサの値を読み込む  
  Serial.println(val);  
  delay(200);  
}
```

光の強さを調べる(if文) Ex504A

```
const int lightPin = A3;
```

```
void setup() {  
  Serial.begin(9600);  
}
```

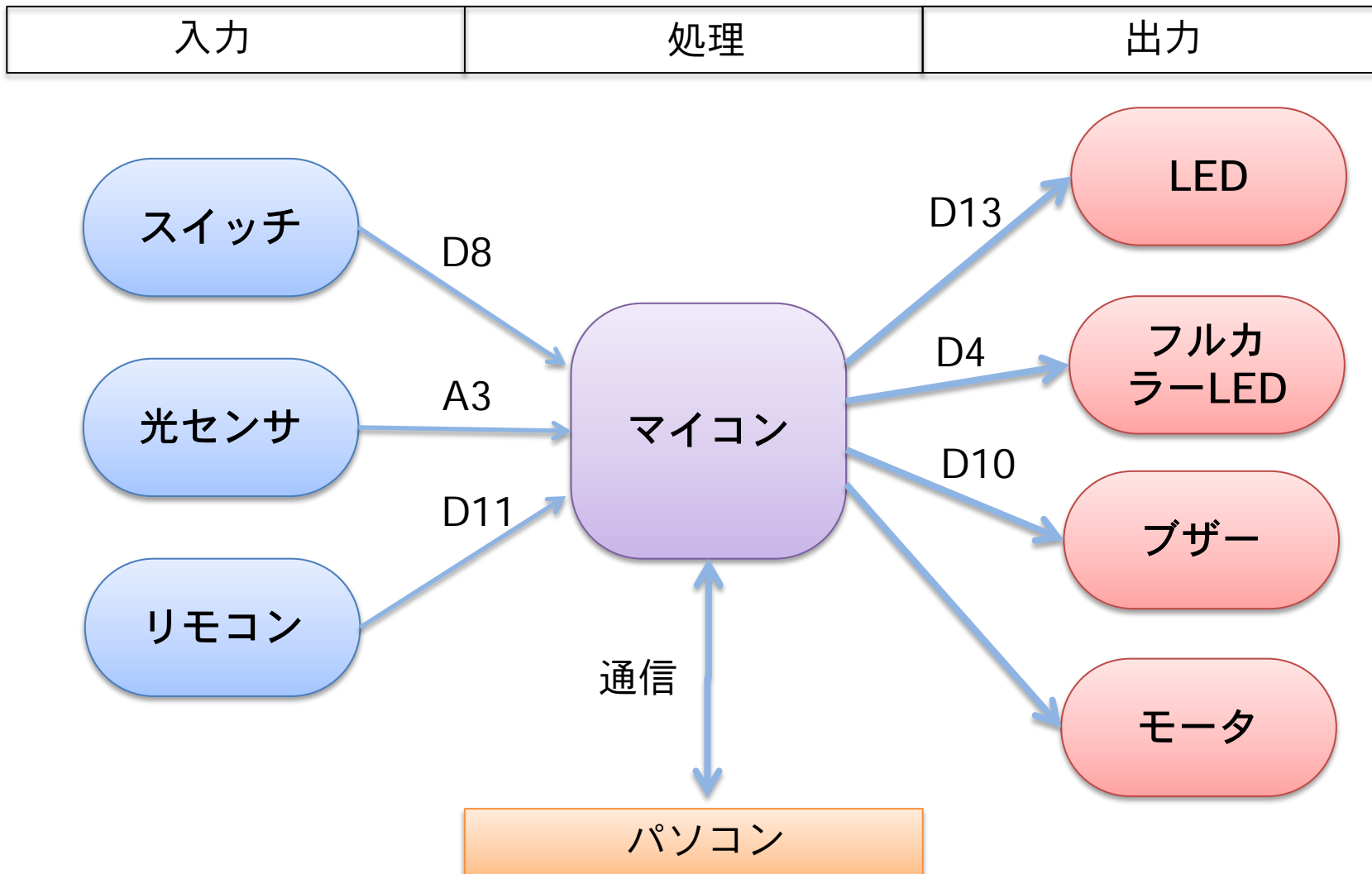
```
void loop() {  
  int val = analogRead(lightPin);  
  if (val < ) {  
    Serial.println("LED On");  
  } else {  
    Serial.println("LED Off");  
  }  
  delay(200);  
}
```

シリアルモニタで調べた値を入れる

暗くなったらLEDを点灯

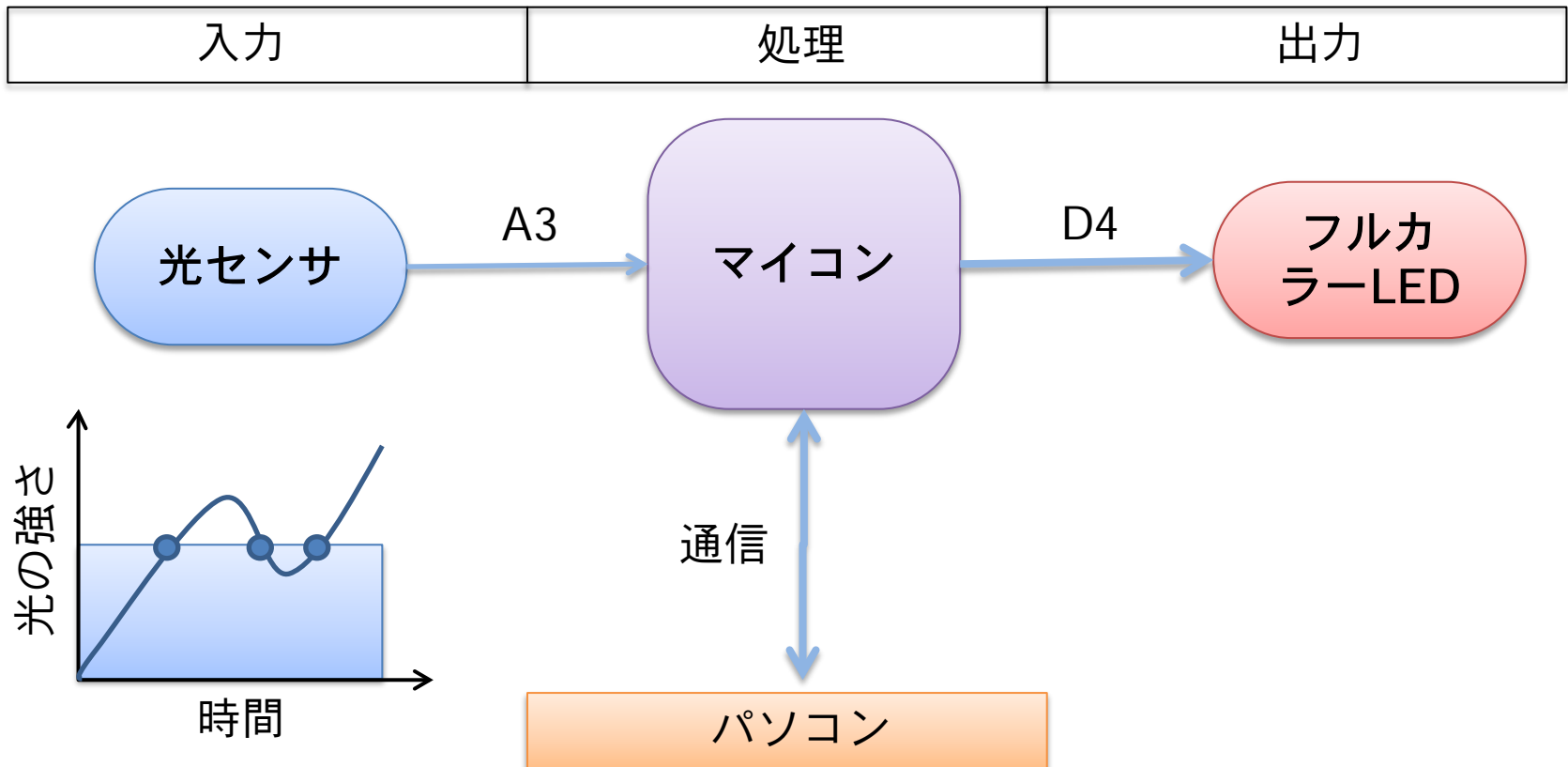
5 シリアル通信の応用

5.1 組み合わせを考える



5.2 光センサとLED

- ◆ 暗くなったらLEDを点灯する
→ 光センサの明るさを調べてLEDを点灯・消灯する



暗くなったらLEDを点灯（光センサとLED） Ex505A

```
#include "FastLED.h"
```

```
const int pinLight = A3;
```

```
//フルカラーLEDの設定
```

```
const int pinLED = 4;
```

```
CRGB led[1];
```

```
void setup() {
```

```
  FastLED.addLeds<WS2811, pinLED, GRB>(led, 1);
```

```
}
```

<次ページに続く>

暗くなったらLEDを点灯（光センサとLED） Ex505A

```
void loop() {  
  int val = analogRead(pinLight);  
  if (val < 200) { // 光センサの明るさを調べる  
    led[0] = CRGB(255, 255, 255); // LED点灯  
  } else {  
    FastLED.clear(); // LED消灯  
  }  
  FastLED.show(); // LED表示更新  
  delay(200);  
}
```

5.3 組み合わせを考える

1. スイッチが押されたら、LEDを点灯する.
2. スイッチが押されたら、ブザーを鳴らす.
3. 暗くなったら、ブザーを鳴らす.

応用：

1. スイッチが押されたら、LEDを5秒点灯した後消灯する.
2. スイッチが押されたら、ブザーを3秒鳴らす.
3. 暗くなったら、LEDを4秒点灯する.
4. 暗くなったら、ブザーを1秒鳴らす.

懐中電灯：スイッチのON・OFFにより、LEDを点灯・消灯する.

センサライト：暗くなったら、LEDを一定時間点灯する.

Arduino→ヘルプ→リファレンス を参照