プログラミング演習 II

フィジカルコンピューティング

第2回　アナログ入出力とシリアル通信

担当：橋本

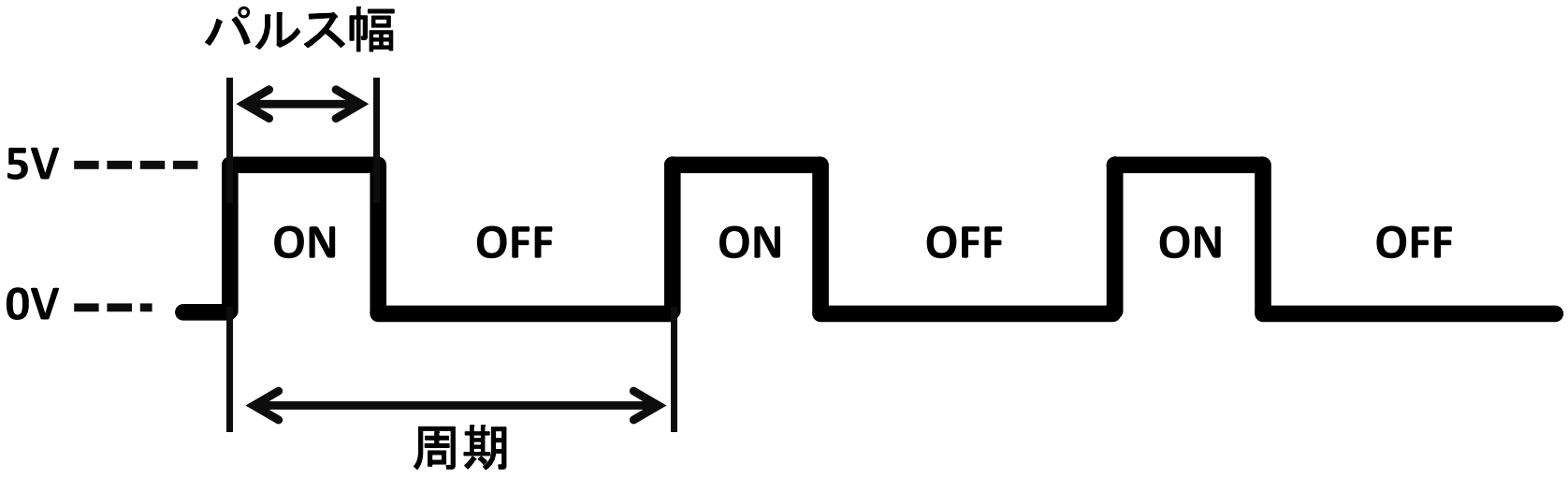
アナログ出力

Arduinoにおけるデジタル出力：LOW か HIGH か（0V or 5V）

Arduinoにおけるアナログ出力：PWM信号（256段階）

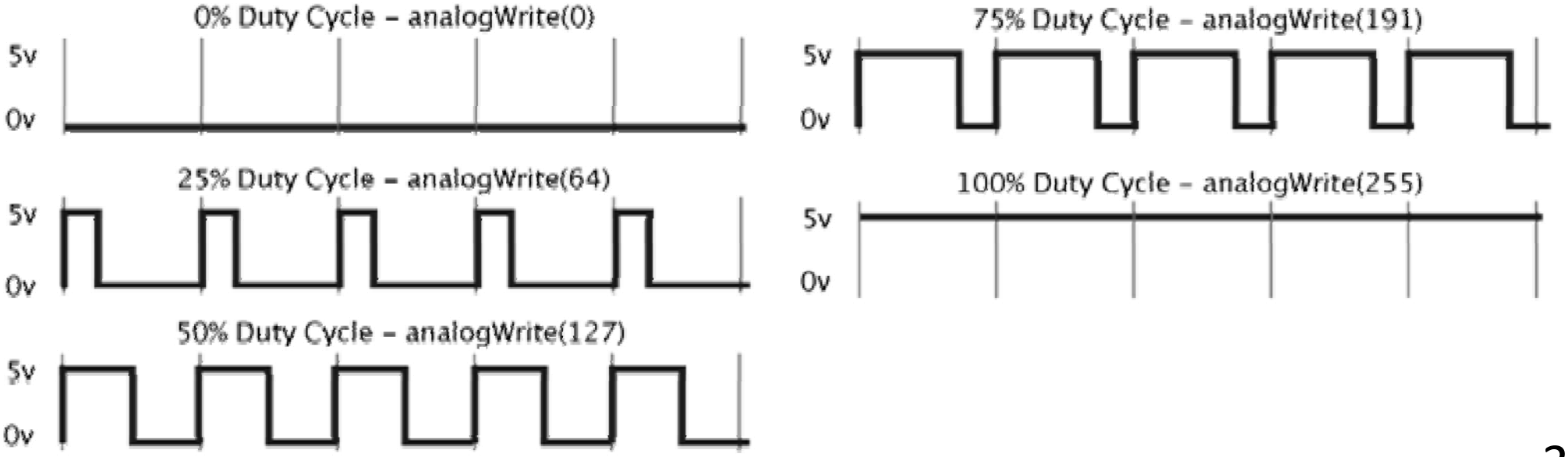
PWM = Pulse Width Modulation（パルス幅変調）

- ON・OFFを高速に切り替える手法
- ON時間とOFF時間の比（デューティ比）を変えてコントロール



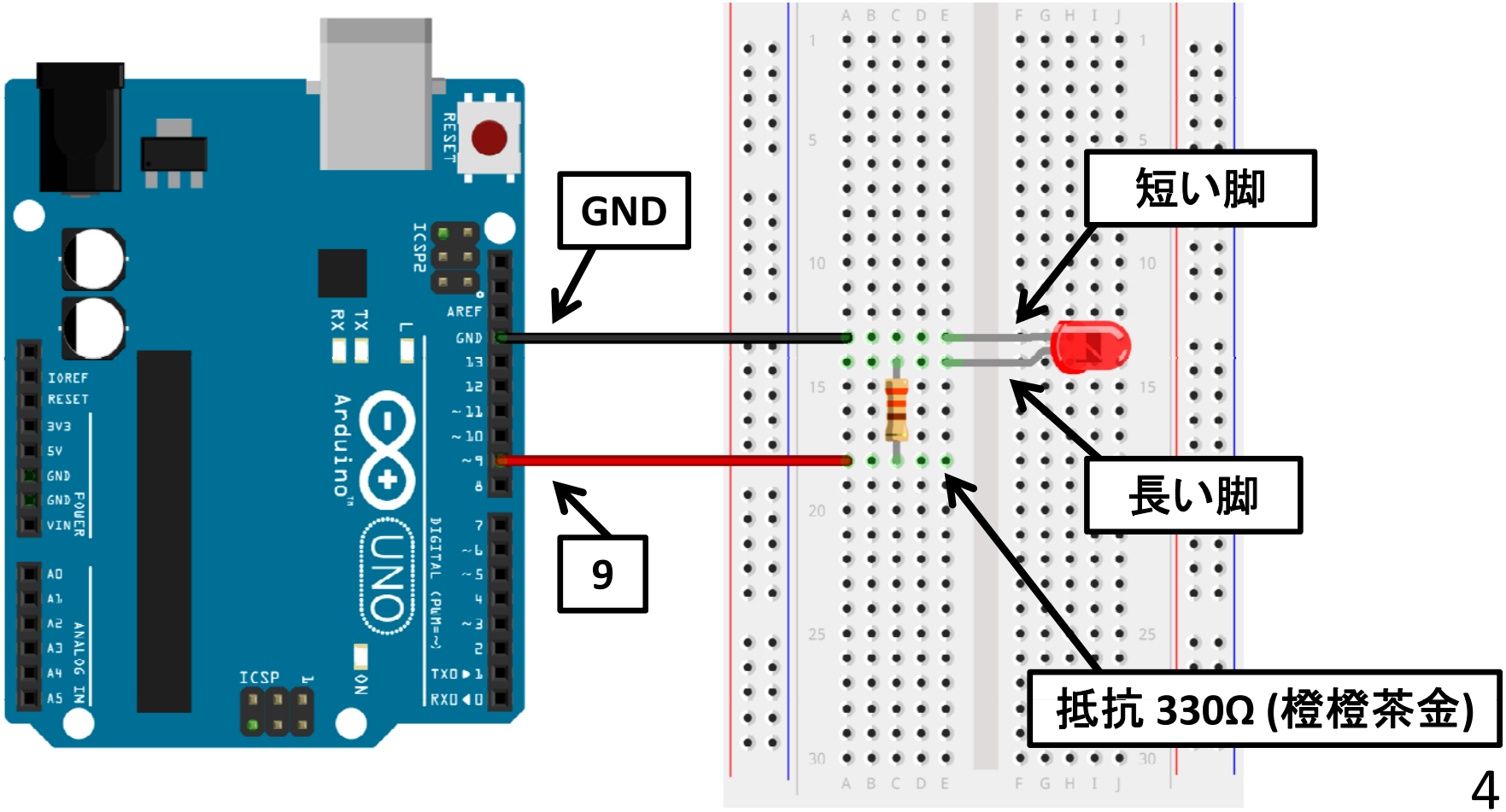
アナログ出力の命令

- **analogWrite(** ピン番号**,** 出力値 **)**
  - 指定したピンに対してPWM出力する
  - 出力値には 0 ～ 255 を指定する
  - PWM出力できるのは3,5,6,9,10,11ピン



PWMでLEDの明るさを制御する回路

- 「～」マークがついた端子がPWM出力できる端子
- 抵抗は 330Ω ※前回と違うのはより明るくするため



LEDの明るさが滑らかに変化するプログラム

```
void setup() {
   pinMode( 9, OUTPUT );
}

void loop() {
   for ( int i=0; i<=255; i++ ) {
     analogWrite( 9, i );
     delay(10);
   }
   for ( int i=255; i>=0; i-- ) {
     analogWrite( 9, i );
     delay(10);
   }
}
```

9番ピンを出力に設定

0から255まで1ずつカウントアップ

9番ピンでアナログ出力

255から0まで1ずつカウントダウン

9番ピンでアナログ出力

アナログ入力

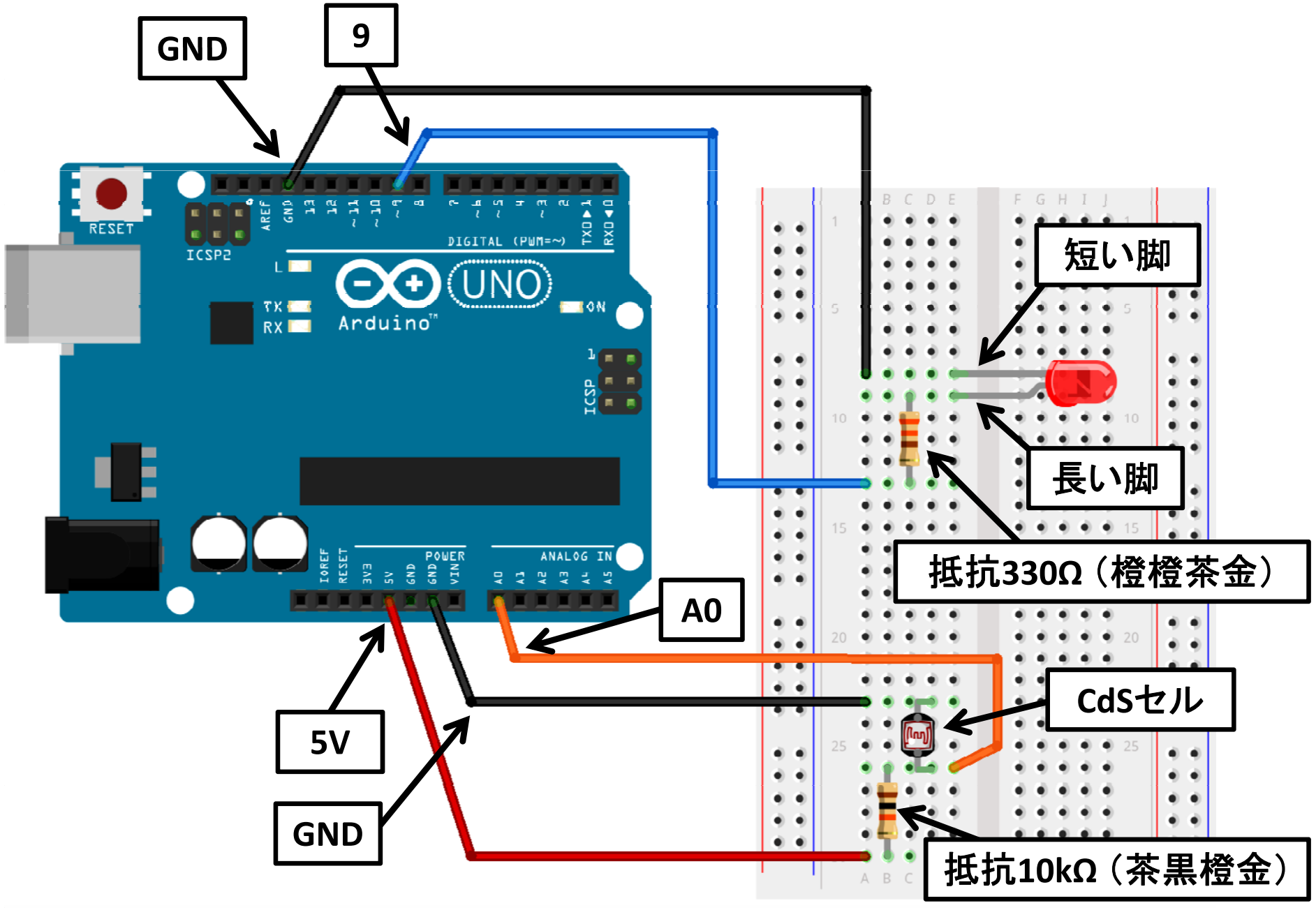
Arduinoにおけるデジタル入力：LOW か HIGH か（0V or 5V）
Arduinoにおけるアナログ入力：0V～5Vを多段階で刻んだ値

- 入力にはアナログ入力端子 A0～A5 を使う
  - 電圧の変化を**1024段階**で読み取れる
  - 5V/1024 = 4.9mV 刻み

アナログ入力の命令

- **analogRead(** ピン番号 **)**
  - 指定したピンの入力電圧を読み取る
  - ピン番号は 0～5 を指定（A0～A5に対応）
  - int値（0～1023）を返す

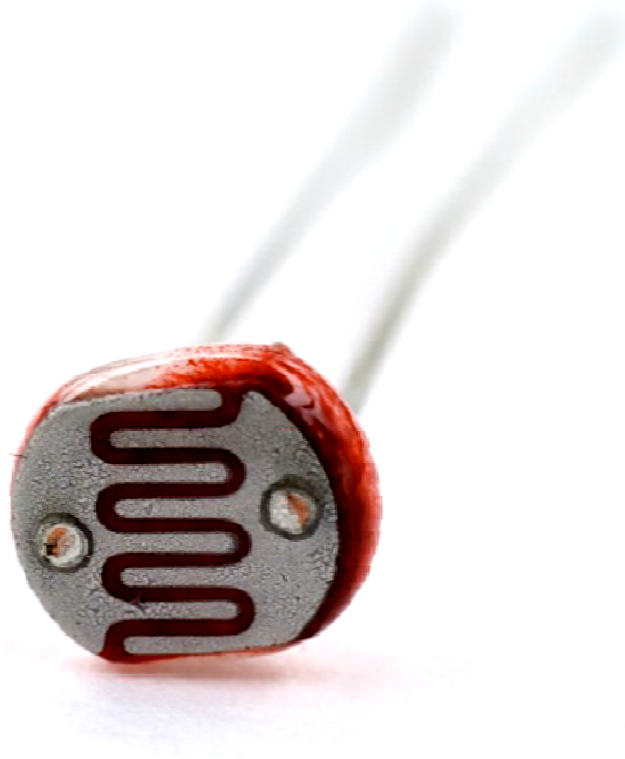
光の強さをセンシングする回路



光センサ（CdSセル）

- あたる光が強くなると電気抵抗が低下する素子
- 硫化カドミウム（CdS）の性質を利用
- フォトレジスタ、光伝導セルとも呼ぶ

抵抗値の変化は、電気回路では
電圧値の変化として読み取る！



周囲の明るさでLEDの点滅周期が変化するプログラム

```
void setup() {
    pinMode( 9, OUTPUT );
}

void loop() {
    int value = analogRead(0);

    digitalWrite( 9, HIGH );
    delay( value );
    digitalWrite( 9, LOW );
    delay( value );
}
```

9番ピンを出力に設定

アナログ入力端子（A0）についてはpinMode()不要

アナログの0番ピンを読み取る

LEDを点滅させる

アナログ入力の値に応じて待ち時間を設定

シリアル通信

- 10010110011…のようなデータ信号を1ビットずつ送る通信方式
- ArduinoとPCを接続して、互いに数値や文字列のデータをやりとりできる
- Arduinoで計測したセンサ情報をPCに送ったり、PCからのコマンド送信によってLEDやモータを制御したりできる

シリアル通信に挑戦

- 回路はさっきの光センシングの回路そのまま

【データの送信】（Arduino→PC）

– 光センサの入力値をPCに送信してモニタリング

【データの受信】（PC→Aruidno）

– PCからコマンドを送ってLEDをコントロール

【データの送信】
光センサの入力値を
PCに送信するプログラム

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

void loop() {
    int value = analogRead(0);
    Serial.println(value);
}
```

シリアル通信を開始する
（通信速度：9600bps）

アナログの0番ピンを読み取る

データをPCに送信する

命令の意味

- **Serial.begin(通信速度)**
  - シリアル通信を開始する。通信速度は以下のいずれか

    300bps, 1200bps, 2400bps, 4800bps, 9600bps, 19200bps, 57600bps, 115200bps

    単位: bps = bits per second（1秒間に送るビット数）

- **Serial.print(データ)　… 改行なし**

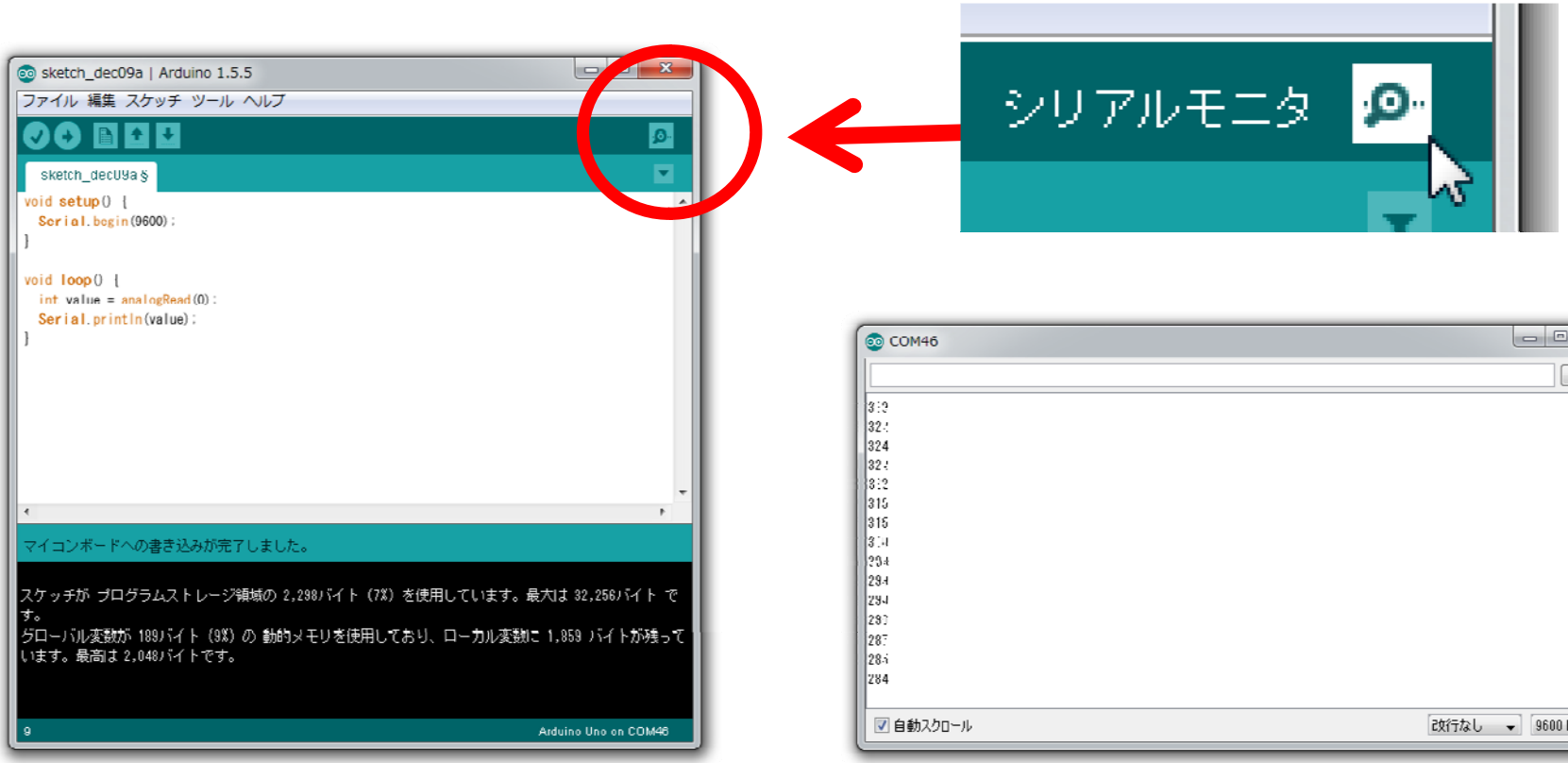
  Serial.println(データ)　… 改行あり
  - 数値や文字列を送れる
  - Arduinoでは「+」による文字列の結合はできないので注意

Serial.print()による文字の出力

- Serial.print(78) - “78”が出力される
- Serial.print(1.23456) - “1.23”が出力される
- Serial.print(‘N’) - “N”が出力される
- Serial.print(“Hello world.”) - “Hello world.”と出力される

データのモニタリング

- ウィンドウ右上にある「シリアルモニタ」というボタンをクリック
- シリアルモニタが起動してデータが流れてくる



曲げセンサ

- 曲げ量に応じて抵抗値が変化するセンサ
- 光センサを取り外して曲げセンサで置き換えてみよう(根元が折れやすいので注意！)
- 曲げに応じて値はどのように変化するだろうか

【データの受信】コマンドでLED制御

```
void setup() {
   Serial.begin( 9600 );
   pinMode( 9, OUTPUT );
}

void loop() {
   if ( Serial.available() > 0 ) {
     int data = Serial.read();
     if ( data == 'a' ) {
       digitalWrite( 9, HIGH );
     }
     else if ( data == 'b' ) {
       digitalWrite( 9, LOW );
     }
   }
}
```

シリアル通信を開始する

9番ピンを出力に設定

データがきているかチェック

データを1バイト読み込む

データが「a」だったら
LEDをONにする

データが「b」だったら
LEDをOFFにする

命令の意味

- **Serial.available()**
  - 受信したデータのバイト数を返す
  - 1以上の値を返せばデータが着ていることになる
  - Arduinoは一度に128バイトまで保持する

- **Serial.read()**
  - 受信したデータの最初の1バイトを返す
  - 戻り値は int型

PCからのデータの送信

- シリアルモニタを起動する
- 入力欄にコマンド（「a」または「b」）を入力して送信ボタンを押す（もしくはエンターキー）
- 「a」を送った時にLEDがONになり、「b」を送った時にLEDがOFFになる

（補足）文字「a」を送るということ

- 文字にはそれぞれ「文字コード」と呼ばれる識別番号が割り当てられている。

  （例）「a」は97、「A」は65、「1」は49

  参考：http://www9.plala.or.jp/sgwr-t/c_sub/ascii.html

- 「a」の文字コード（97）を2進数で表すと「01100001」

- この1/0の並びをON/OFFの電気信号に置き換えて伝送している。

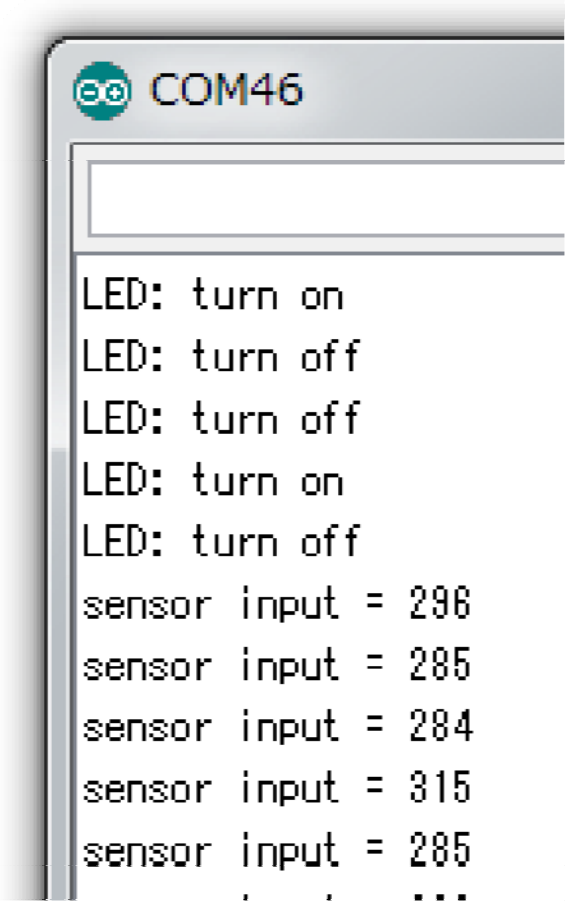
（補足）int型での受信データの判定

- Serial.read()は受信データをint型で返す
- 文字を受信したときは？
  - 文字コードが受信される
  - すなわち、「a」という文字がきた時は、97というint型の値を受け取っている
  - なので、このように判定することもできる（’a’ は 97 と置換可能）

```
int data = Serial.read();
if ( data == 97 ) {
  digitalWrite( 9, HIGH );
}
```

練習問題1

- コマンド入力によって以下の反応を返すプログラムにしてみよう
  - 「a」という文字を送信したら、LEDが点灯して「LED: turn on」という文字列を返す
  - 「b」という文字を送信したら、LEDが消灯して「LED: turn off」という文字列を返す
  - 「c」という文字を送信したら、光センサの値を「sensor input = ???」という形式で返す



練習問題2

- 光センサに手をかざしたときだけLEDが点灯するプログラムを作ってみよう
  - 【ヒント1】 if文によって、「センサの値がいくつ以上のときに点灯（or消灯）する」という分岐を作ってみよう。
  - 【ヒント2】 シリアル通信を使って、手をかざしているときに、センサの入力値がどのくらいになっているかチェックして閾値（しきいち）を決めよう。