

デザイン言語ワークショップ (電子工作)

第2回 2017 / 10 / 2

講師: 山岡 潤一

スライドURL : <https://tinyurl.com/y9or6rsm>

本日の流れ

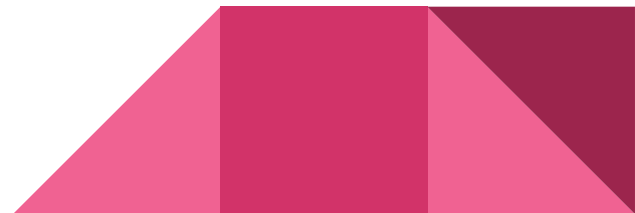
1. フィジカルコンピューティングについて

2. Arduino導入

3. デジタル入出力、アナログ入出力

LEDとスイッチを使用

4. 応用編、色々なセンサやアクチュエータの紹介



ArduinoIDEの準備

Arduino IDE: プログラミングをするツール

– IDEとは統合開発環境のこと

(Integrated Development Environment)

- Arduinoの公式サイトからダウンロード (<http://www.arduino.cc/>)

- Windowsの場合マイコンが認識されない場合がある

コントロールパネル → デバイスマネージャー → ほかのデバイスの下にある **不明なデバイス** を右クリック → ドライバーソフトウェアの更新 → コンピューターを参照して～を選択 → 「参照」をクリックして Arduinoフォルダの中の driversを選択 → 次へ

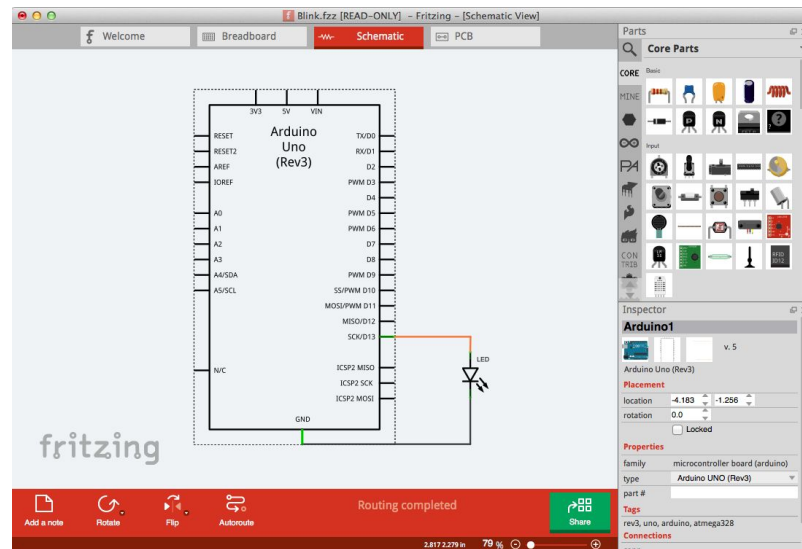
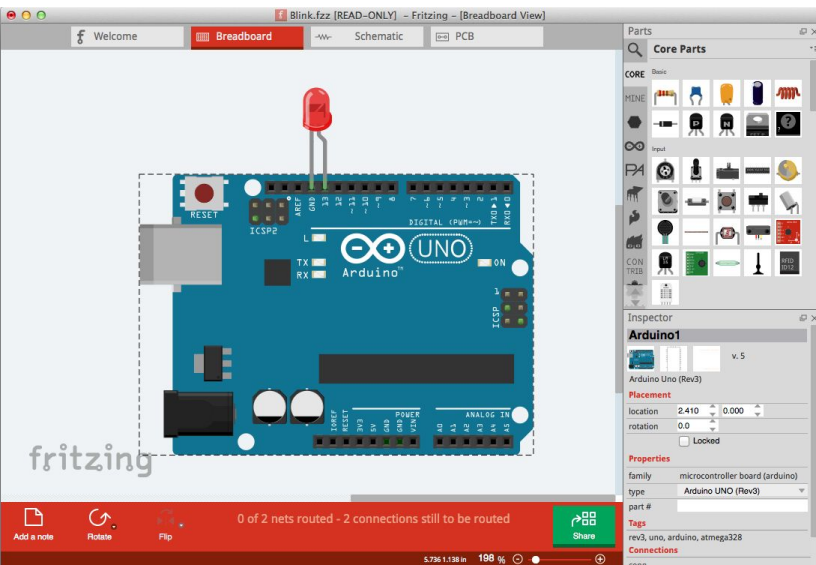
でドライバーをインストールします。

デバイスマネージャーのポートの下に Arduino UNOが表示されていれば完了です。

Fritzingの準備

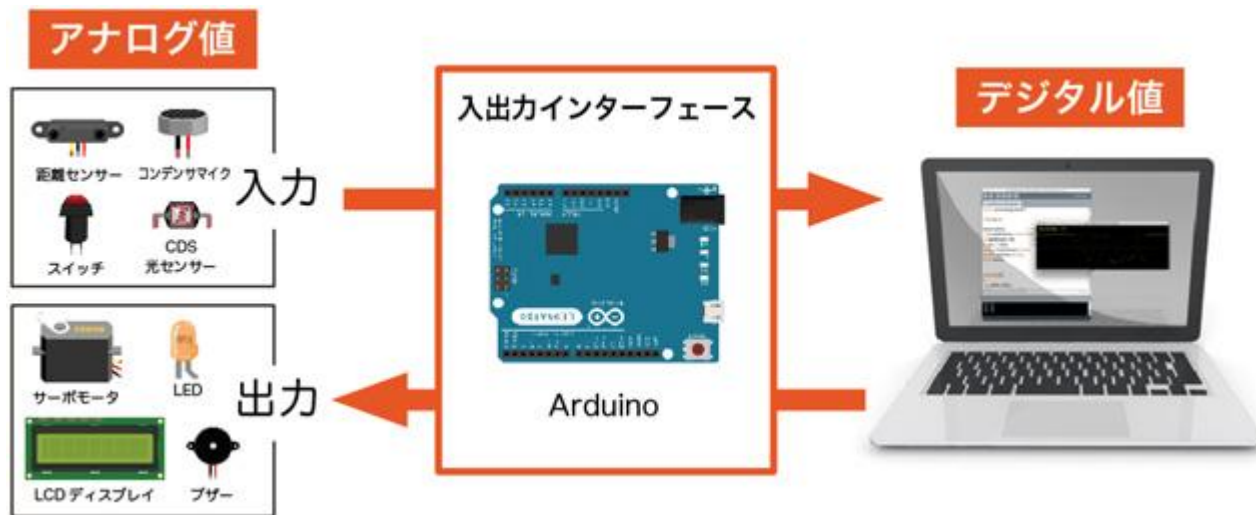
回路設計を行うソフトウェア。

<http://fritzing.org/download/>



フィジカルコンピューティング

コンピュータにさまざまな入出力デバイスをつないで、実世界との物理的なやりとりを実現する方法



フィジカルコンピューティング

Arduinoとは

初心者でも簡単に扱えるマイコンボード

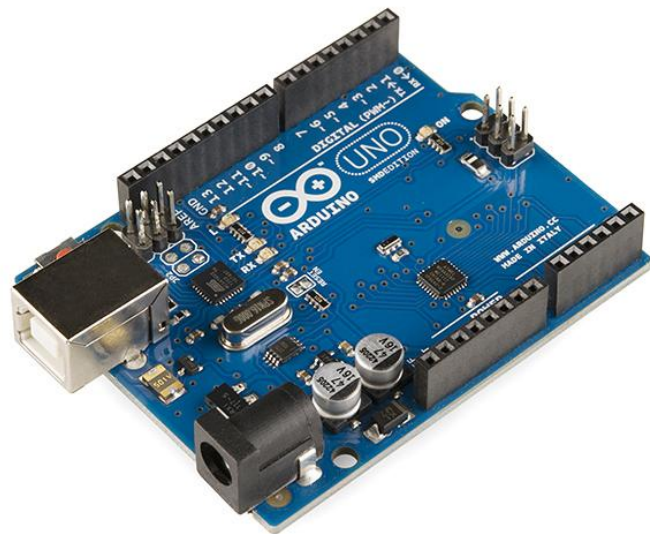
(micro-controller board)

電気を制御するプログラムを書き込める

PCとの通信もできる

電池をつなげば単体で動作する

プログラムはjava言語



フィジカルコンピューティング

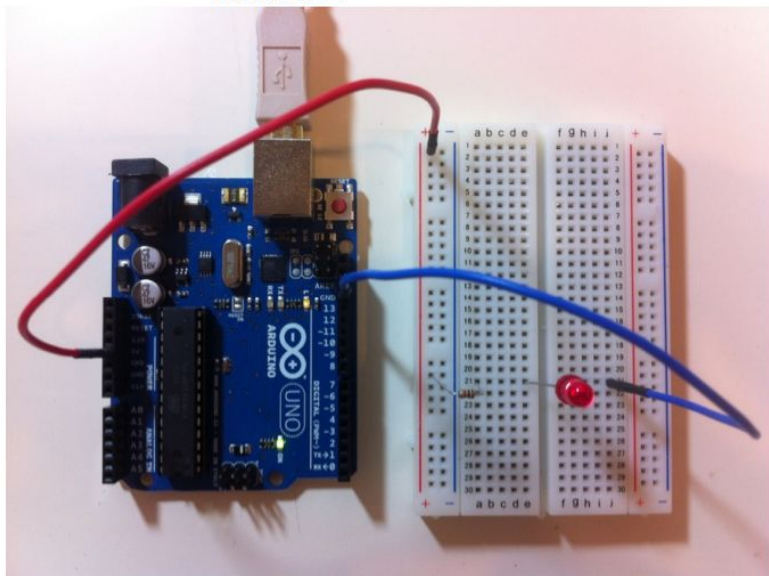
何ができるか

- ・タッチセンサの値に応じてLEDを光らせる
- ・リモコンでコントロールするラジコン
- ・水温をネット上にアップしていくデバイス



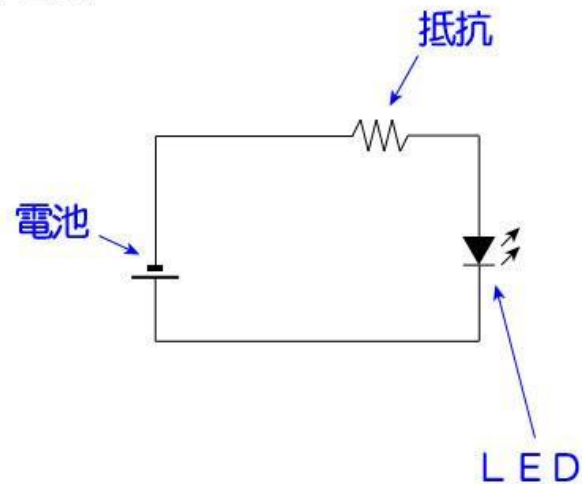
まずはLEDを光らせてみよう

PCのUSBへ



330Ω LED 赤
(橙橙茶金)

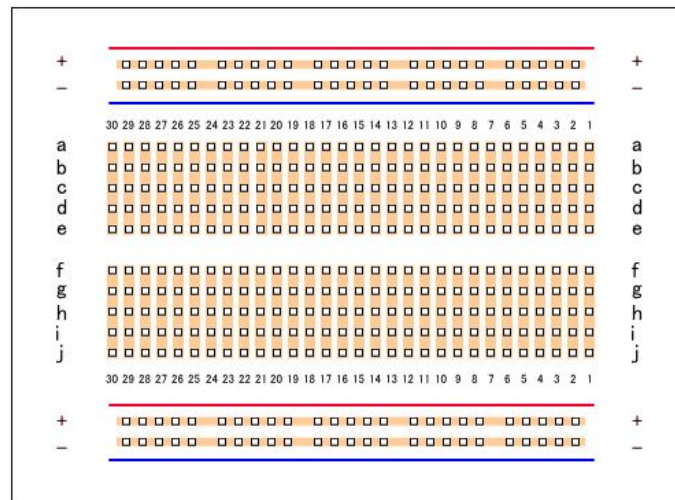
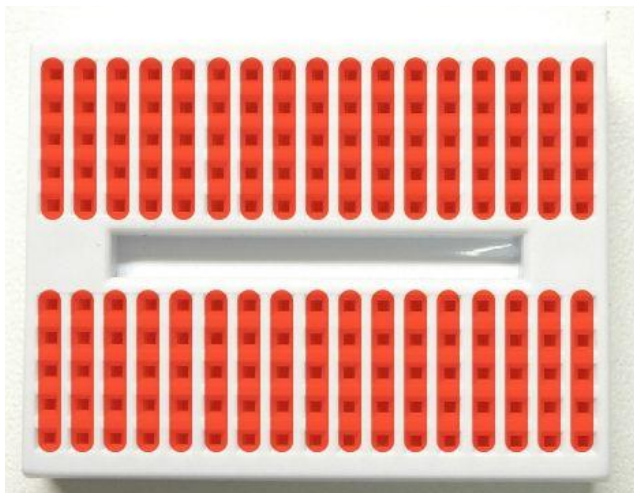
基本回路



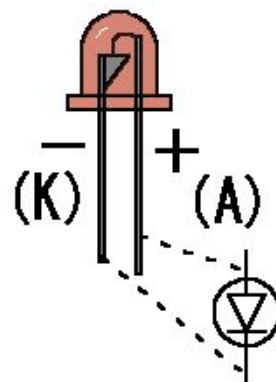
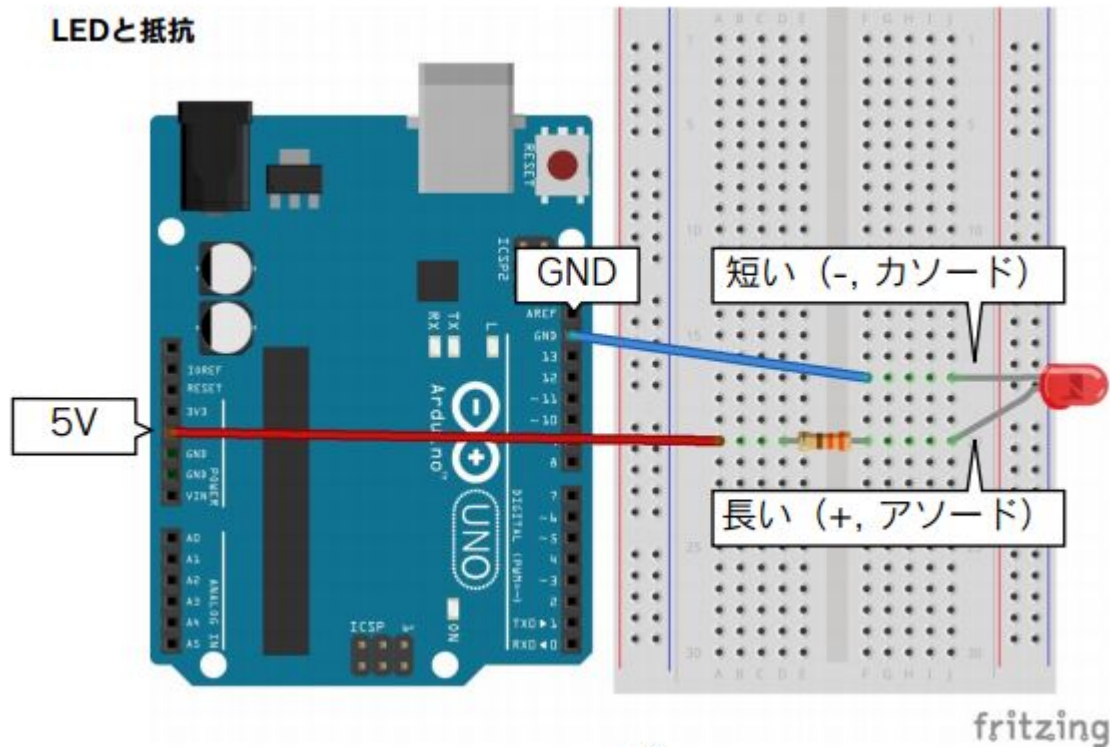
ブレッドボード

ブレッドボード

パーツやワイヤを挿して電子回路を作れる道具
内部に金具が入っていて **列単位**でつながっている



まずはLEDを光らせてみよう



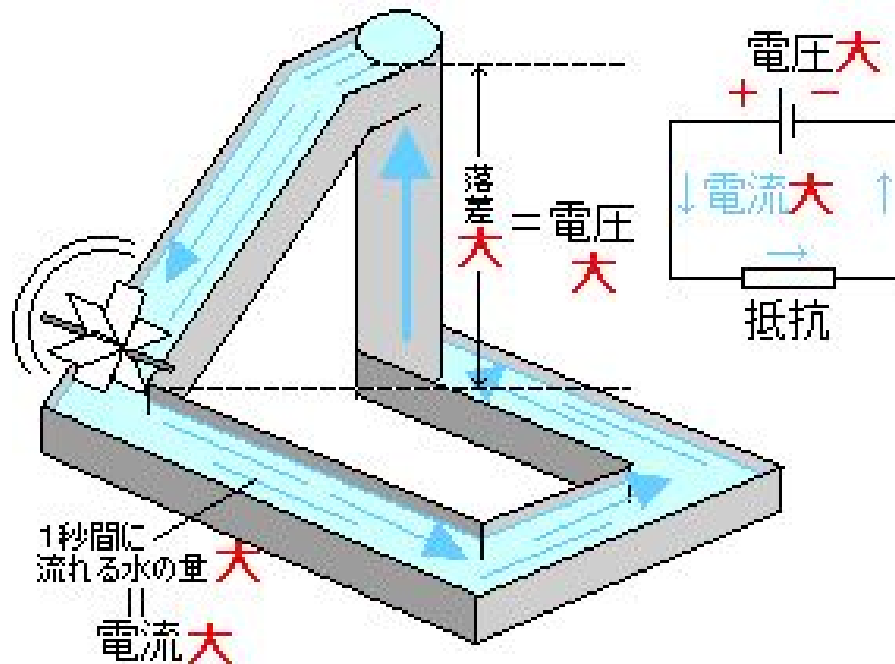
電流、電圧、抵抗の関係

電流の流れは水の流れに例えられる。

水路の落差→電圧

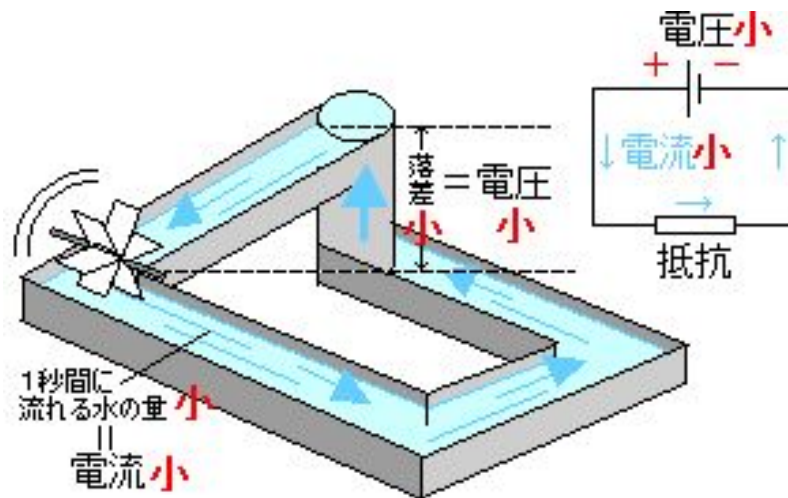
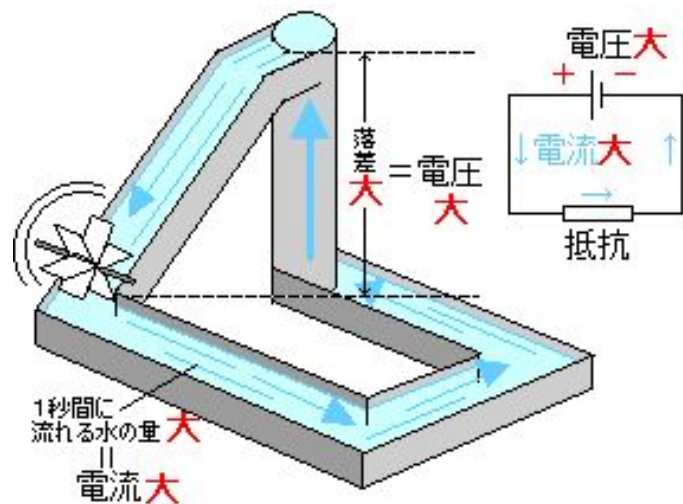
水車→電気抵抗

1秒間に流れる水の量→電流



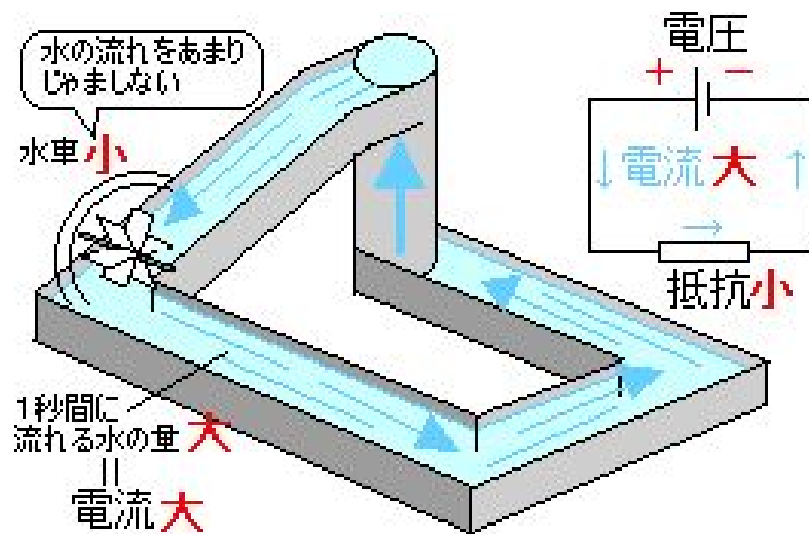
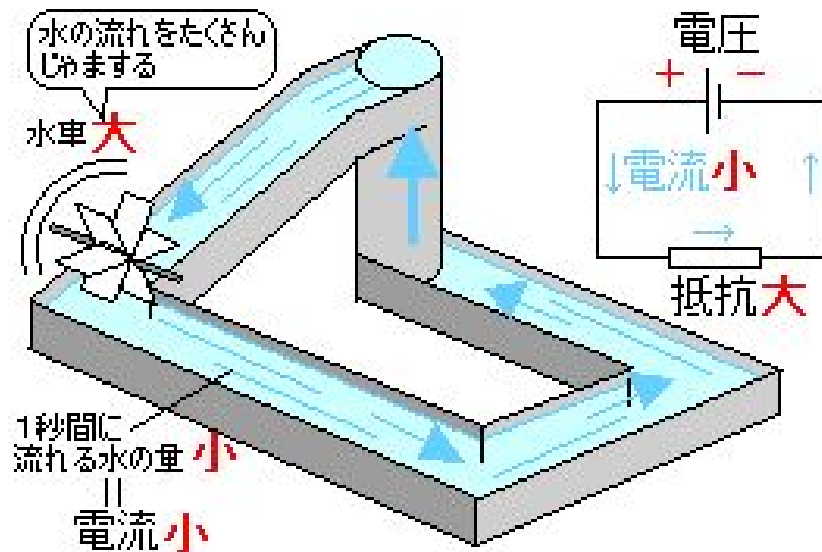
電圧と電流の関係

落差(電圧)が大きいほど、1秒間に流れる水の量(電流)が多くなる。



抵抗と電流の関係

水車（抵抗）が小さいほど、1秒間に流れる水の量（電流）が多くなる



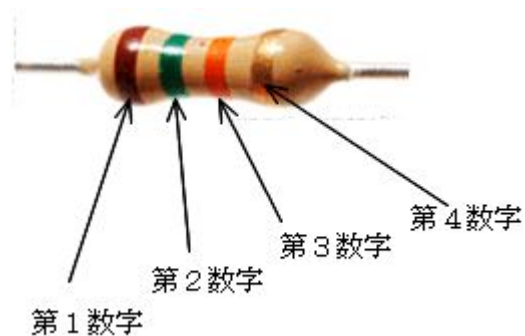
抵抗

抵抗の読み方

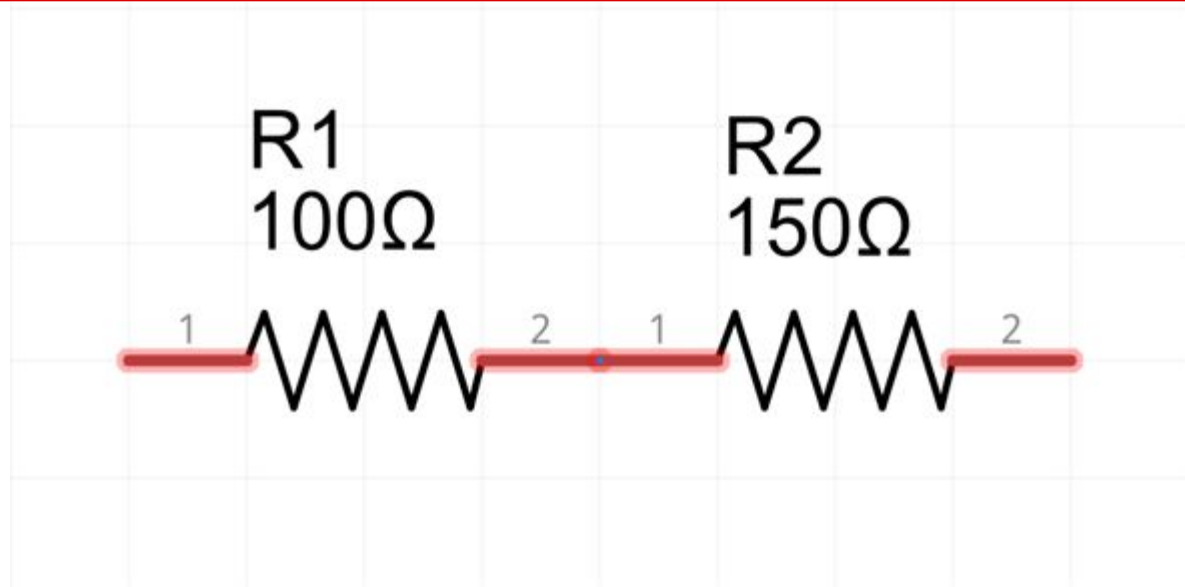
- ・100Ω抵抗(茶黒茶金)
- ・10kΩ抵抗(茶黒橙金)

カラー抵抗値の写真早読み表

http://part.freelab.jp/s_regi_list.html

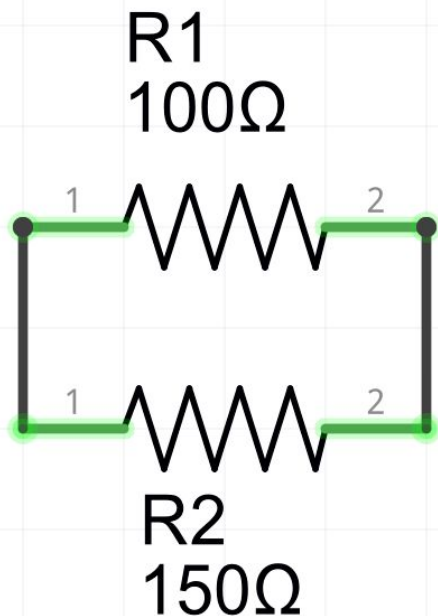


直列抵抗



$$\text{合成抵抗 } R = 100[\Omega] + 150[\Omega] = 250[\Omega]$$

並列抵抗



$$\begin{aligned}\text{合成抵抗 } R &= \frac{1}{\frac{1}{100[\Omega]} + \frac{1}{150[\Omega]}} \\ &= \frac{1}{0.01 + 0.0066} = \frac{1}{0.0166} \\ &= 60[\Omega]\end{aligned}$$

抵抗の選び方

■Electrical -Optical Characteristics

(Ta=25℃)

Item	Symbol	Condition	Min	Typ	Max	Unit
DC Forward Voltage	V _F	I _F =20mA	1.8	2.0	2.4	V
DC Reverse Current	I _R	V _R =5V	-	-	10	μA
Domi. Wavelength	λ _D	I _F =20mA	565	570	575	nm
Luminous Intensity	I _v	I _F =20mA	-	500	-	mcd
50% Power Angle	2θ _{1/2}	I _F =20mA	-	15	-	deg

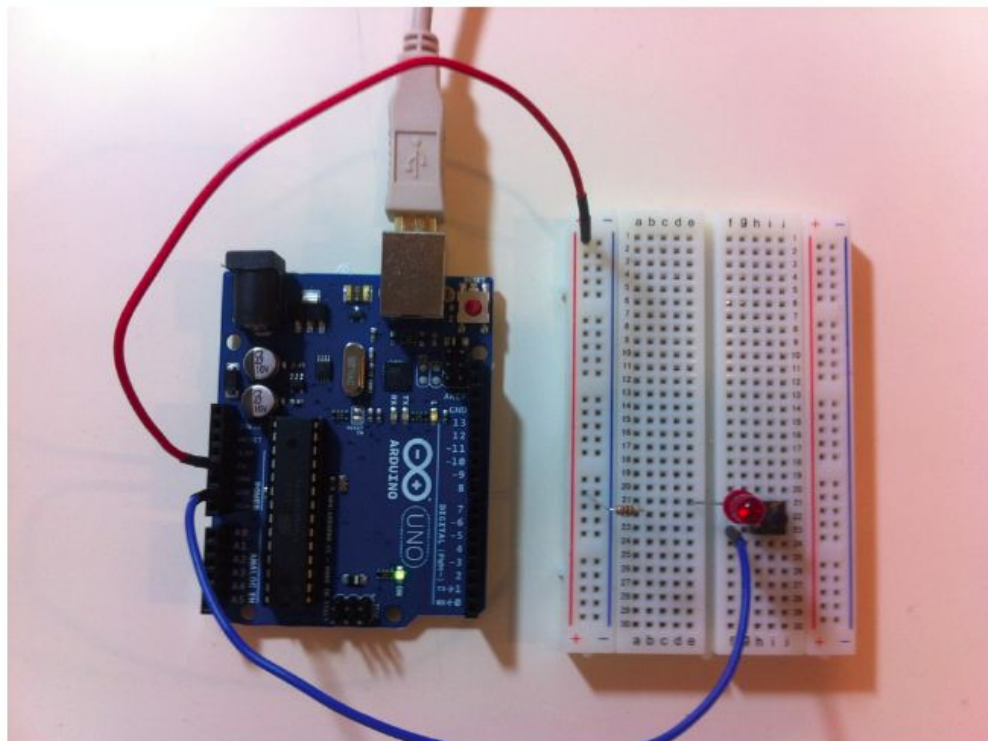
電位差 3[V]

= 合成抵抗 R × 電流 0.020[A]

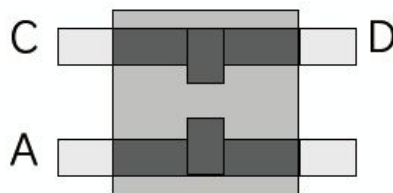
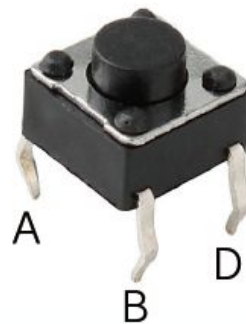
LED抵抗値計算

<http://diy.tommy-bright.com/>

スイッチとLED

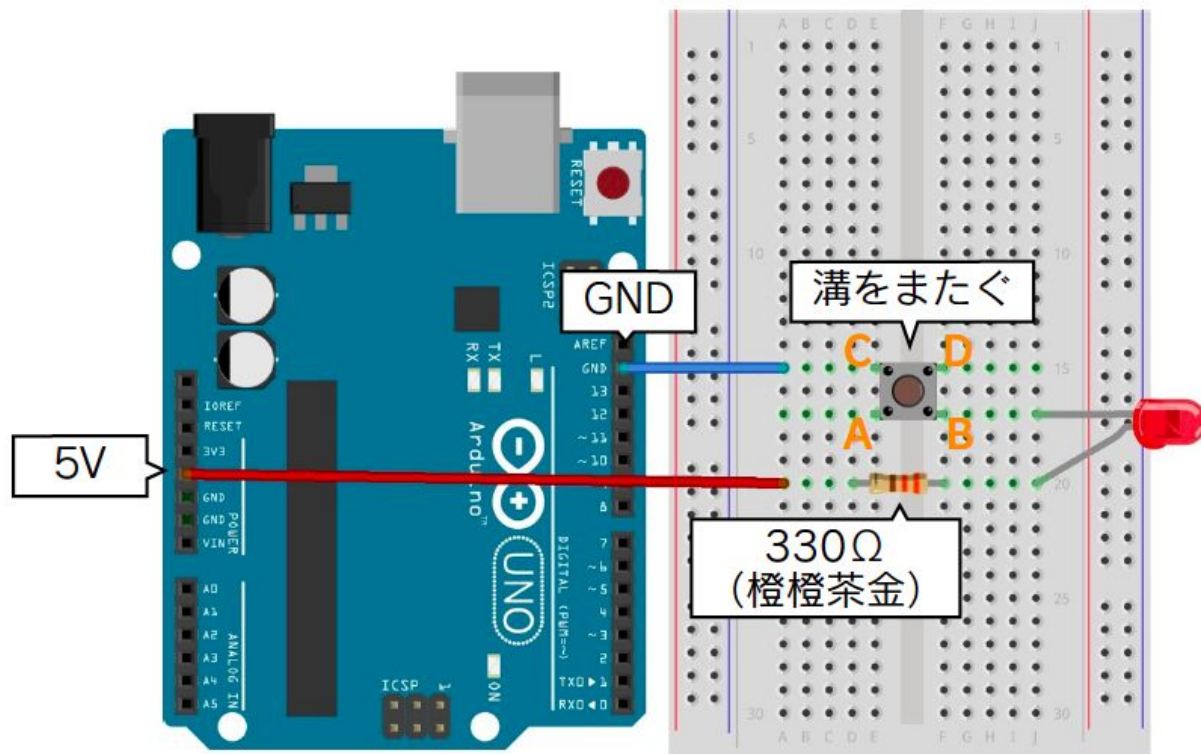


タクトスイッチ



押すと繋がる

スイッチとLED



プログラムと電子回路

プログラムで電子回路を操るとはということか？

－【電圧を読み取る】

- ・センサで計測される物理量は電圧で表現される

－【電圧を変える】

- ・LEDやモータを駆動するときの指令値も電圧で表現される

これら2つの実行手順をプログラムによって記述する

デジタル情報の表現

－ コンピュータの世界では「0」と「1」で表現される

－ 電子回路の世界では「LOW (0V)」と「HIGH (5V)」で表現される

ArduinoIDEの準備

Arduino IDE: プログラミングをするツール

– IDEとは統合開発環境のこと

(Integrated Development Environment)

- Arduinoの公式サイトからダウンロード (<http://www.arduino.cc/>)

- Windowsの場合マイコンが認識されない場合がある

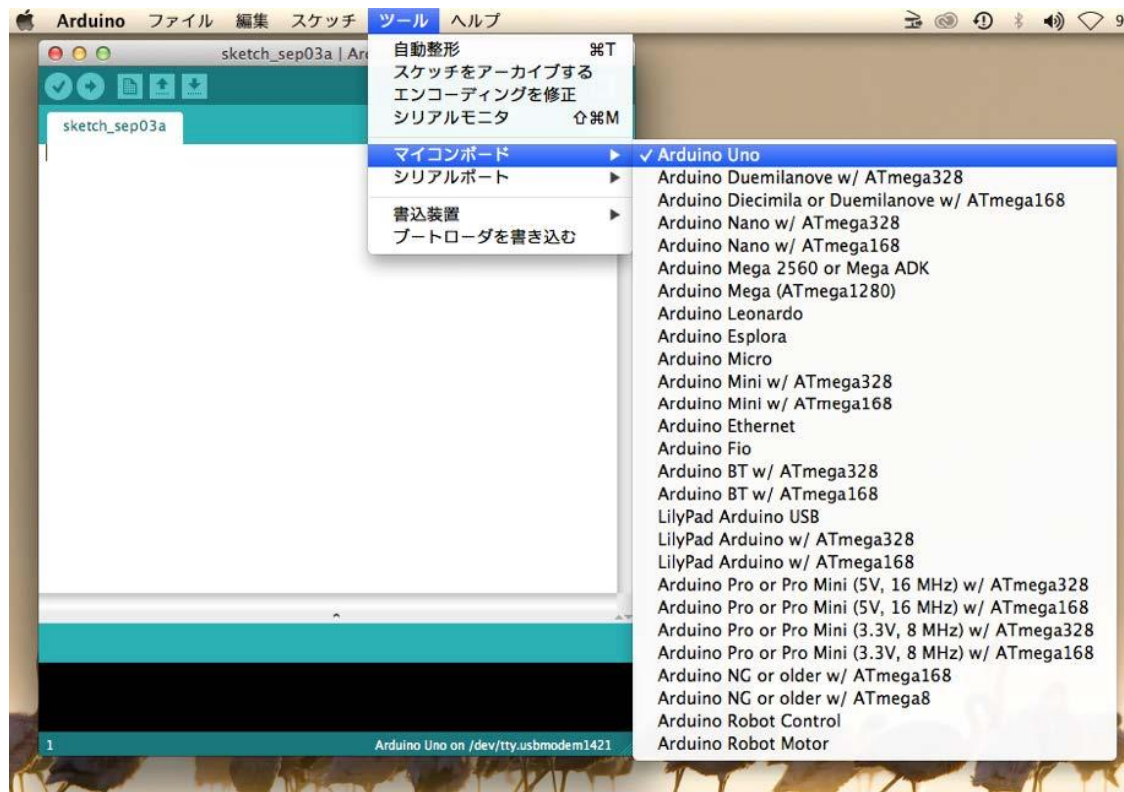
コントロールパネル → デバイスマネージャー → ほかのデバイスの下にある **不明なデバイス** を右クリック → ドライバーソフトウェアの更新 → コンピューターを参照して～を選択 → 「参照」をクリックして Arduinoフォルダの中の driversを選択 → 次へ

でドライバーをインストールします。

デバイスマネージャーのポートの下に Arduino UNOが表示されていれば完了です。

ArduinoIDEでスケッチを書き込んでみる

- ・USBケーブルで
ArduinoとPCを接続
- ・マイコンボード選択
Tools → Board →
Arduino Uno



ArduinoIDEでスケッチを書き込んでみる

・シリアルポート選択

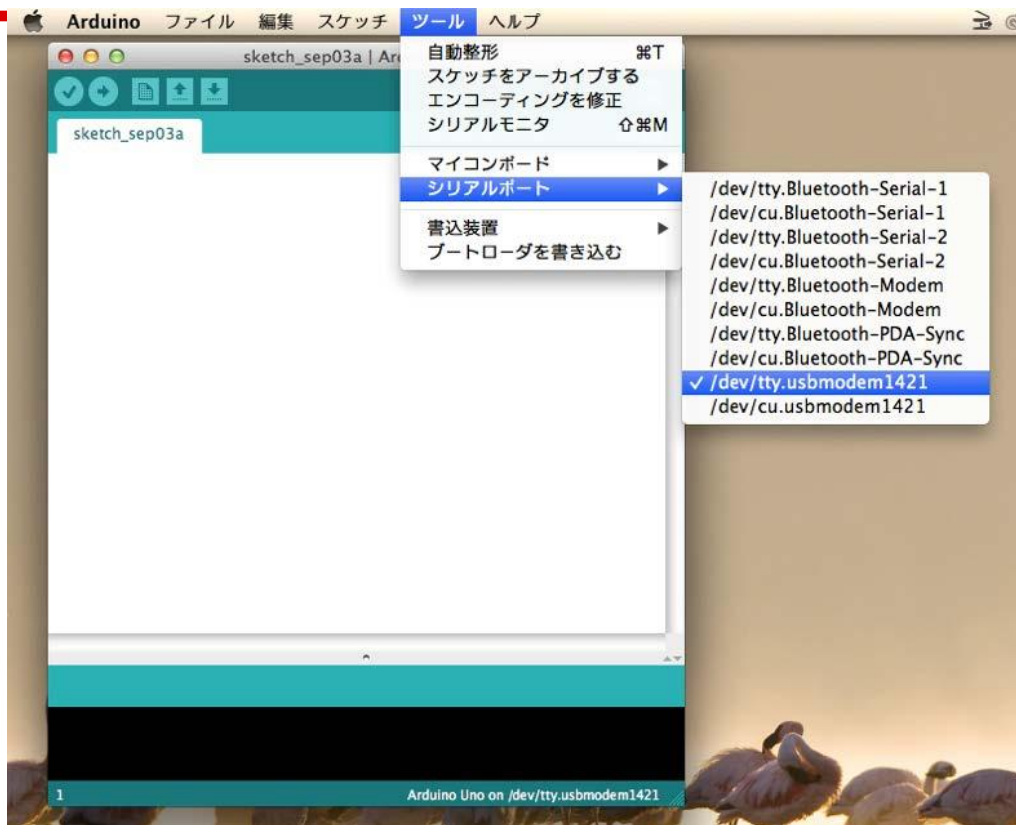
Tools → Port →

【Macの場合】

/dev/cu.usbmodem~

【Windowsの場合】

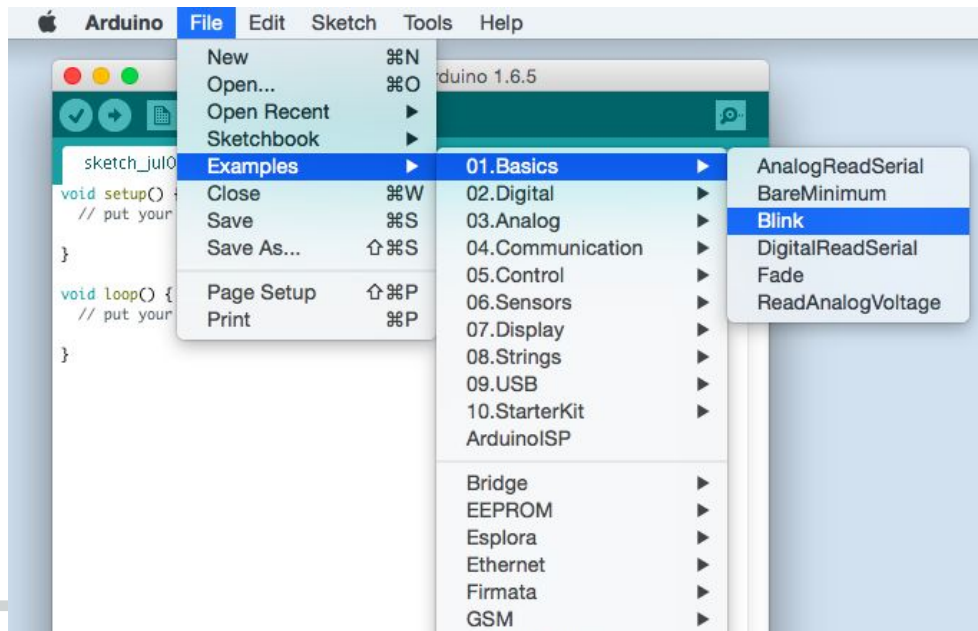
COM~



ArduinoIDEでスケッチを書き込んでみる

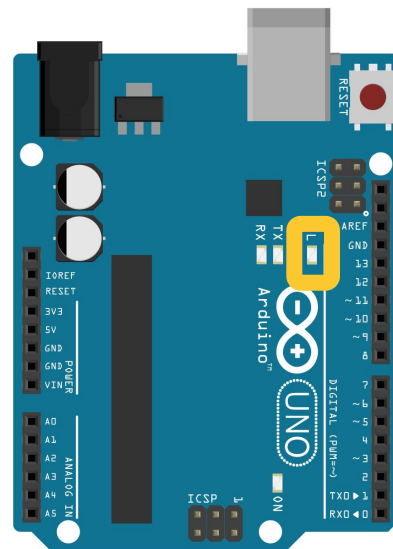
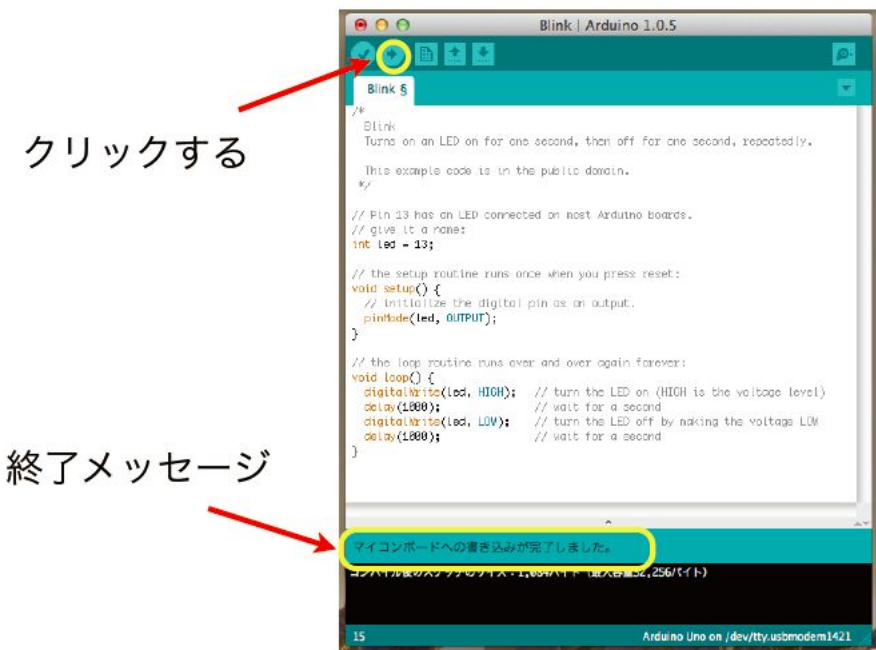
スケッチの選択

File → Examples → 01.Basics → **Blink**



ArduinoIDEでスケッチを書き込んでみる

マイコンボードに書き込む



アイコンの**LED**が点滅します

デジタル出力:Lチカしてみよう

LED点滅(通称:Lチカ、最初にやることの定番)

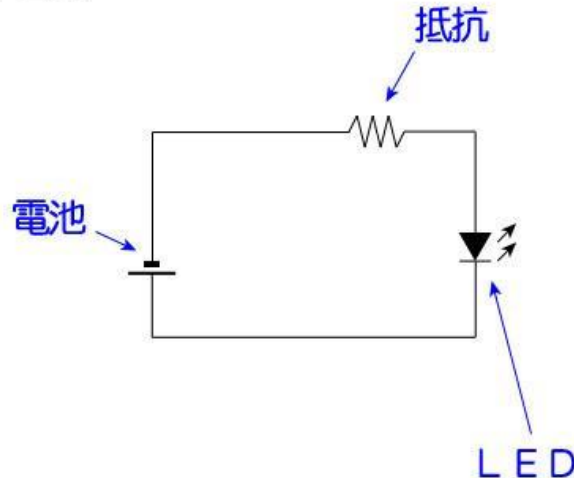
digitalWrite

digitalWrite(ピン,HIGH);

あるいは

digitalWrite(ピン,LOW)

基本回路



回路図

電池がArduinoに置き換わったと
考えれば良い

デジタル出力:Lチカしてみよう

つなぎ方(配線)

- ・LEDの向きに注意(LEDが壊れる)
- ・部品を指すときは
必ずUSBを抜いてから指す

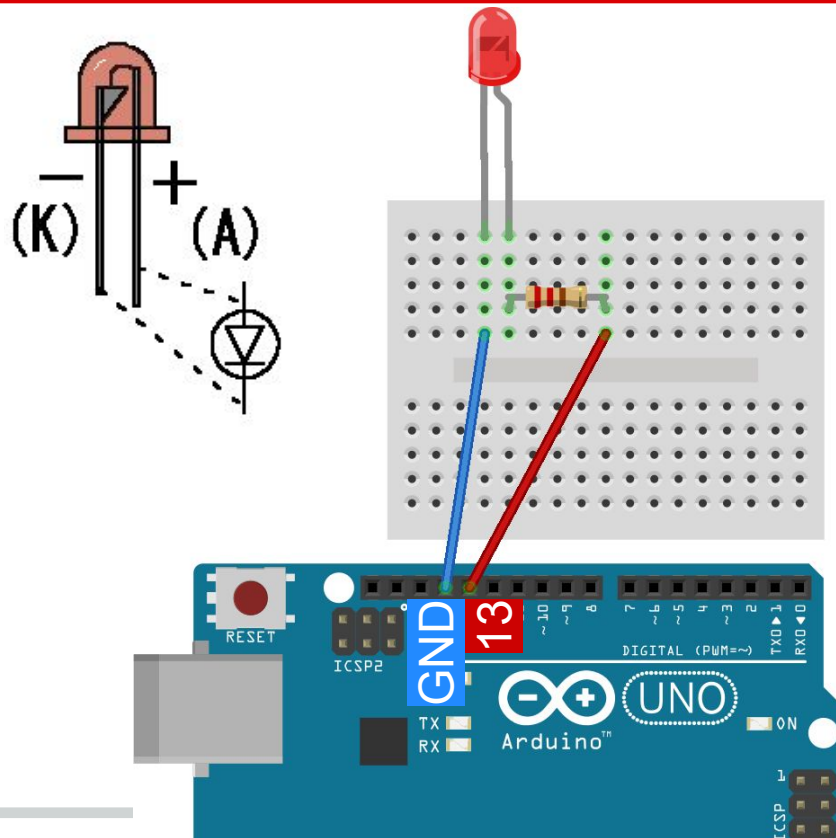
Arduino

ブレッドボード

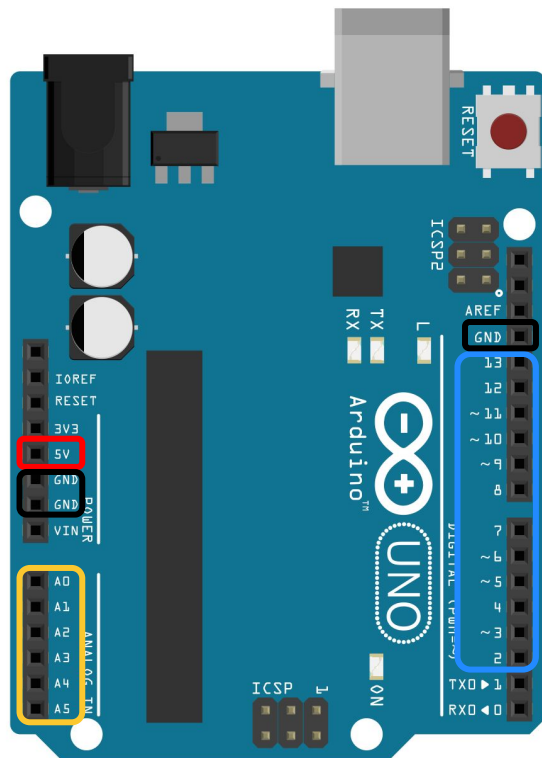
ジャンパワイヤ2本

LED1個

100Ω抵抗(茶黒茶金)



Arduinoのピンの説明



【2】～【13】 デジタル入出力端子

プログラムで入力電圧を読みとったり、
出力電圧を変えることができる

【A0】～【A5】 アナログ入力端子

入力電圧を1024段階読で読み取る

【5V】

5Vの電圧が常時出力されている。電源のプラス端子に相当

【GND】

グラウンドと読む（groundの略）。
電源のマイナス端子（0V）に相当。
3つあるが、どれにつないでも同じ。

デジタル出力:Lチカしてみよう

Blinkの解説

// や /* */ で囲まれている部分は無視されます(コメントアウト)

// セットアップ関数 起動の**1回だけ**呼ばれる

```
void setup() {  
  pinMode(13, OUTPUT); // ピンモードの設定 13番を出力するのでOUTPUT  
}
```

// ループ関数 **何度も繰り返して**呼ばれる

```
void loop() {  
  digitalWrite(13, HIGH); // 13番をHIGH(5V)にする  
  delay(1000);           // 1秒待つ  
  digitalWrite(13, LOW); // 13番をLOW(0V)にする  
  delay(1000);           // 1秒待つ  
}
```

デジタル出力:Lチカしてみよう

変数 int 値に**名前**を付けてまとめることができる。まとめて値を変えたいときに便利

```
int led = 13; //ピンの番号を宣言
```

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

```
void loop() {  
  digitalWrite(led, HIGH);  
  delay(1000);  
  digitalWrite(led, LOW);  
  delay(1000);  
}
```

デジタル出力:Lチカしてみよう

Blinkプログラム解説

```
int led = 13;
void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  digitalWrite(led, HIGH); // 13番をHIGH(5V)にする
  delay(1000);             // 1秒待つ
  digitalWrite(led, LOW);  // 13番をLOW(0V)にする
  delay(1000);             // 1秒待つ
  (~~~)
}
```

delayの値を変えてみて、明滅のタイミングを変えてみよう

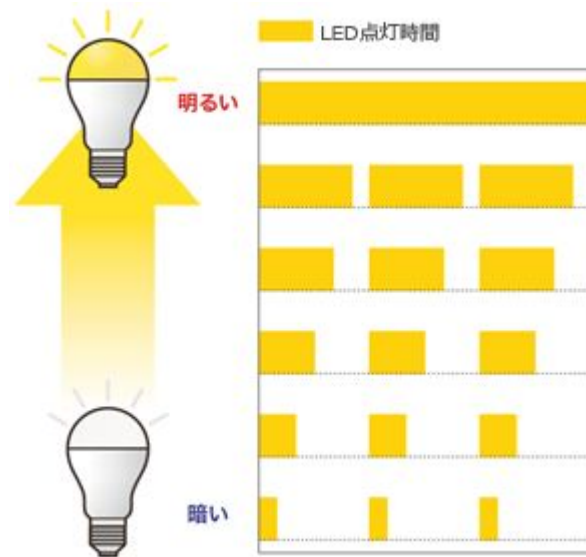
また(~~~)の部分にdigitalWriteとdelayを追加して複雑な明滅を作ってみる。

アナログ出力:PWM出力

LEDの明るさを自由に変える
PWM (Pulse Width Modulation)
[analogWrite](#)

原理

点灯する時間でLEDの明るさを調整する



アナログ出力:PWM出力

任意の明るさに変えてみよう

(アナログ出力の時はpinModeの指定はしなくていい)

```
int ledPin = 9;
void setup() {
}
void loop() {
  analogWrite(ledPin, ???); // 0~255の間の数値を入れる
}
```

プログラミング:for文

for(){ } の間は任意の回数繰り返すことができる。

例:

```
for(int i = 0; i < 10 ; i++){  
    digitalWrite(13,HIGH);  
}
```

整数iの初期数値は0; iが10になる前まで繰り返す; iに1を足していく
つまりdigitalWrite(13,HIGH);を10回繰り返す

アナログ出力: PWM出力

Fadingの解説 (delayを変えると速度が変わる)

fadeValueが0から255になるまで5ずつ足していく。

```
for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5) {  
    analogWrite(ledPin, fadeValue);    //fadeValueの値をアナログ出力  
    delay(30);  
}
```

fadeValueが255から0になるまで5ずつ引いていく。

```
for(int fadeValue = 255 ; fadeValue >= 0; fadeValue -=5) {  
    analogWrite(ledPin, fadeValue);  
    delay(30);  
}
```

デジタル入力:スイッチをつける

スイッチを押すと、LEDが光るようにする
[digitalRead](#)

スイッチの配線

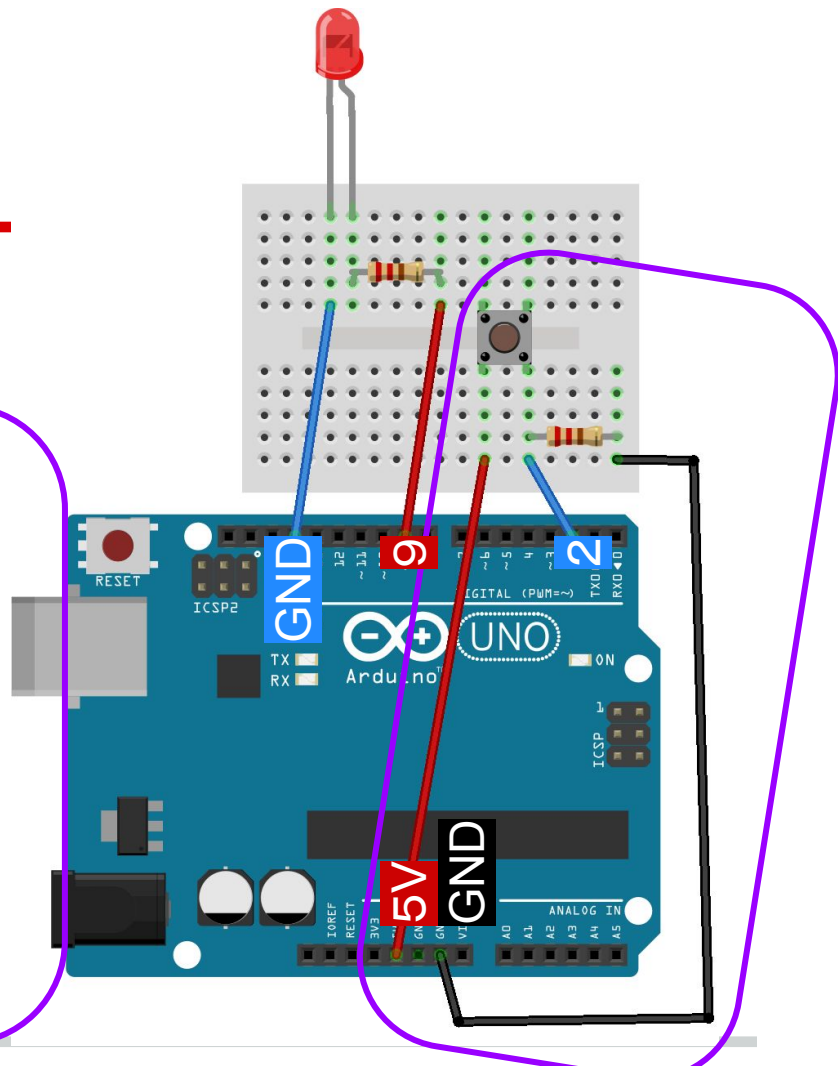
タクトスイッチ

10k Ω 抵抗(茶黒橙金)

330 Ω (LED)

ジャンパワイヤ3本

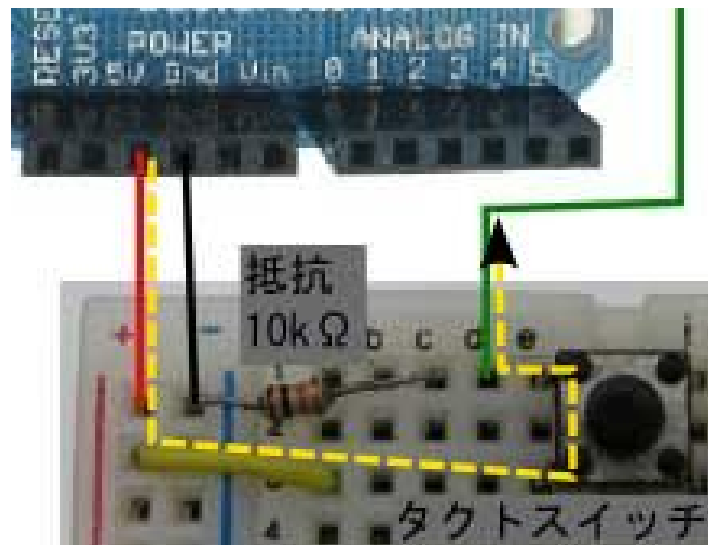
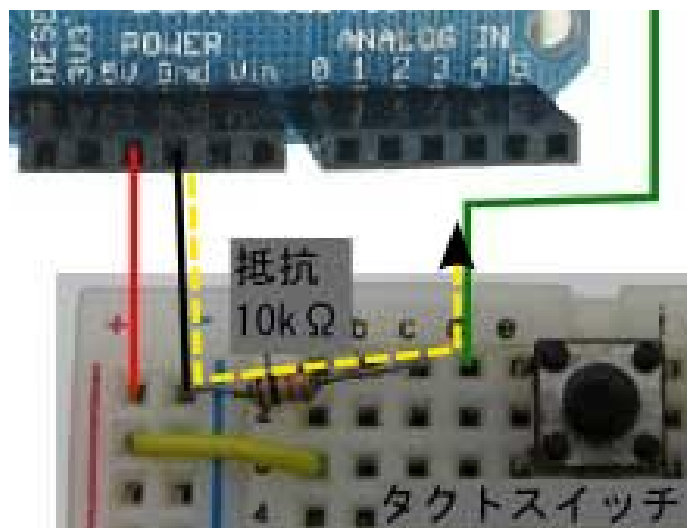
- ここで使われる抵抗は[プルダウン抵抗](#)
信号を確実に伝える為に取り付ける
スイッチを押していない間はLOW



プルダウン抵抗

スイッチ押されてない時

押されている時



プログラム:if文

ifは条件分岐

- ・もしAとBが一緒だったらCを実行

```
if( A == B){  
    C  
}
```

- ・その他

if(A > B) もしAがBより大きかったら

if(A != B) もしAとBが違ったら

プログラム:if文

- ・もしAとBが一緒だったらCを実行、それ以外だったらDを実行

```
if( A == B){  
    C  
}else{  
    D  
}
```

デジタル入力:スイッチをつける

スケッチを考えて自分で書いてみよう

2番ピンのデジタル入力値に応じて→9番のLEDを光らせる

```
void setup() {  
    pinMode( 2 , INPUT); // 2番をインプットモードにする  
    pinMode( 9 , OUTPUT); // 9番をアウトプットモードにする  
}  
void loop(){  
    int value = digitalRead(2); // 変数valueに2番ピンのデジタル入力値を入れる  
    if( ?? == ?? ){  
        ?????  
    }else{  
        ?????  
    }  
}
```


デジタル入力:スイッチをつける

ヒント

```
void setup() {  
    pinMode( 2 , INPUT); // 2番をインプットモードにする  
    pinMode( 9 , OUTPUT); // 9番をアウトプットモードにする  
}  
void loop(){  
    int value = digitalRead(2); // 変数valueに2番ピンのデジタル入力値を入れる  
    if( value == HIGH ){ // valueがHIGH(5Vだったら)  
        ?????  
    }else{ // それ以外だったら ( valueがLOWだったら)  
        ?????  
    }  
}
```

デジタル入力:スイッチをつける

答え

```
void setup() {  
    pinMode( 2 , INPUT);  
    pinMode( 9 , OUTPUT);  
}  
void loop(){  
    int value = digitalRead(2);  
    if( value == HIGH ){  
        digitalWrite(9,HIGH);  
    }else{  
        digitalWrite(9,LOW);  
    }  
}
```

デジタル入力: スイッチをつける

解説

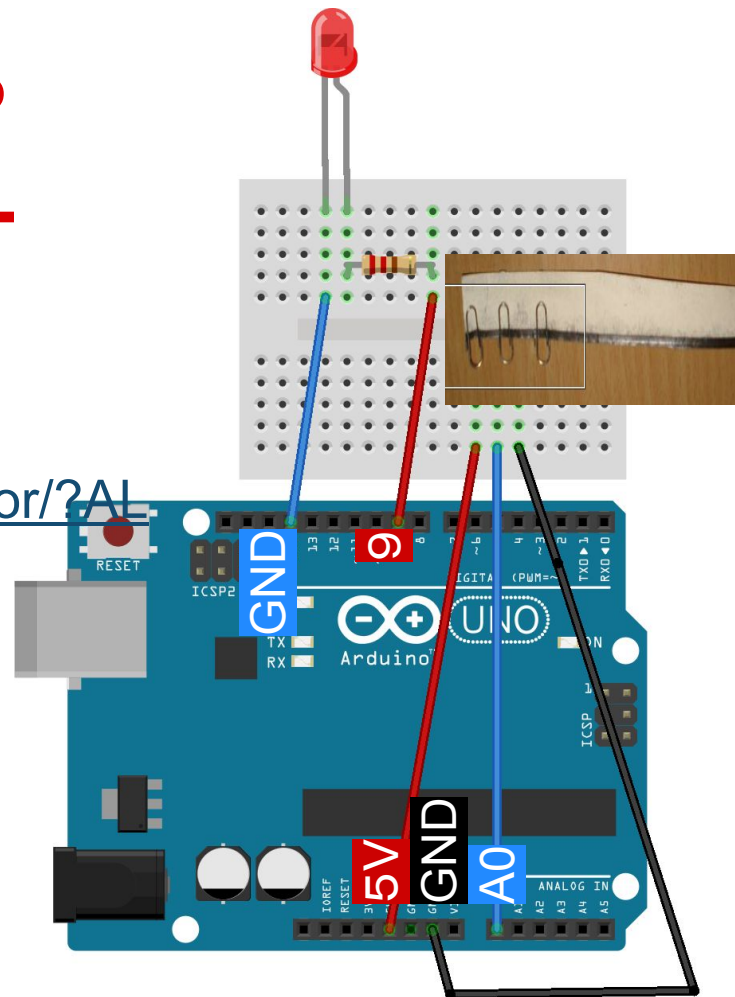
```
void setup() {  
    pinMode( 2 , INPUT); // 2番をインプットモードにする  
    pinMode( 9 , OUTPUT); // 9番をアウトプットモードにする  
}  
void loop(){  
    int value = digitalRead(2); 変数valueに2番ピンのデジタル入力値を入れる  
    if( value == HIGH ){ // valueがHIGH(5Vだったら)  
        digitalWrite(9,HIGH); // 9番ピンに5V流す  
    }else{ // それ以外だったら (valueがLOWだったら)  
        digitalWrite(9,LOW); //9番ピンに0V流す  
    }  
}
```

アナログ入力:アナログスイッチをつける

鉛筆で可変抵抗を作ってみよう

[analogRead](#)

<http://www.instructables.com/id/Paper-Resistor/?AL>
LSTEPS



アナログ入力:アナログスイッチをつける

スケッチを考えて書いてみよう

アナログピン0番の値に応じて、9番のLEDをアナログ出力で光らせる

使う命令 → **analogRead(ピンの番号);**

```
void setup() {  
    pinMode( 9 , OUTPUT);  
}  
void loop(){  
    int value = analogRead(?) / 4; // アナログ入力値を4で割り、変数valueにそれを入れる  
    ?????  
}
```

- ・ちなみに**analogRead**は0~5Vを0~1024で返すのに対して、**analogWrite**は0~5Vを0~255で表す
→ **analogWrite**を4で割ることで1024を255段階に変換する
-

アナログ入力:アナログスイッチをつける

答え

```
void setup() {  
    pinMode( 9 , OUTPUT); // 9番をアウトプットモードにする  
}  
void loop(){  
    int value = analogRead(0) / 4; //0番ピンのアナログ入力値を4で割り、変数valueにそれを入れる  
    analogWrite(9,value); // 9番ピンにvalue分の電気を流す  
}
```

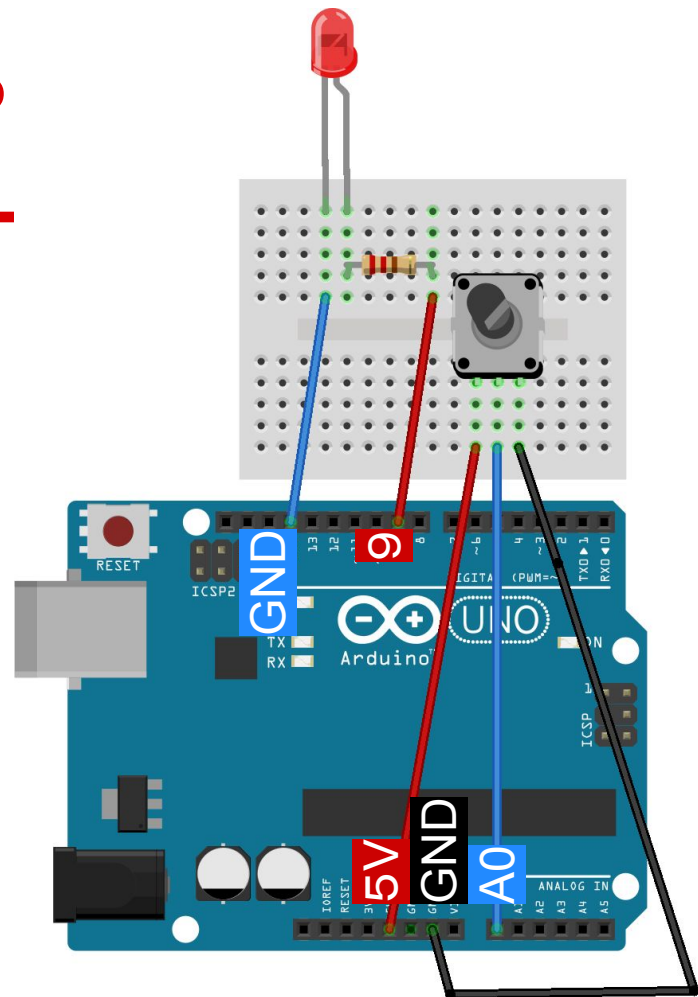
アナログ入力:アナログスイッチをつける

ボリュームを回すと、LEDも合わせて明るくなる
[analogRead](#)

ボリュームの配線

ボリューム

ジャンパワイヤ3本



練習問題1

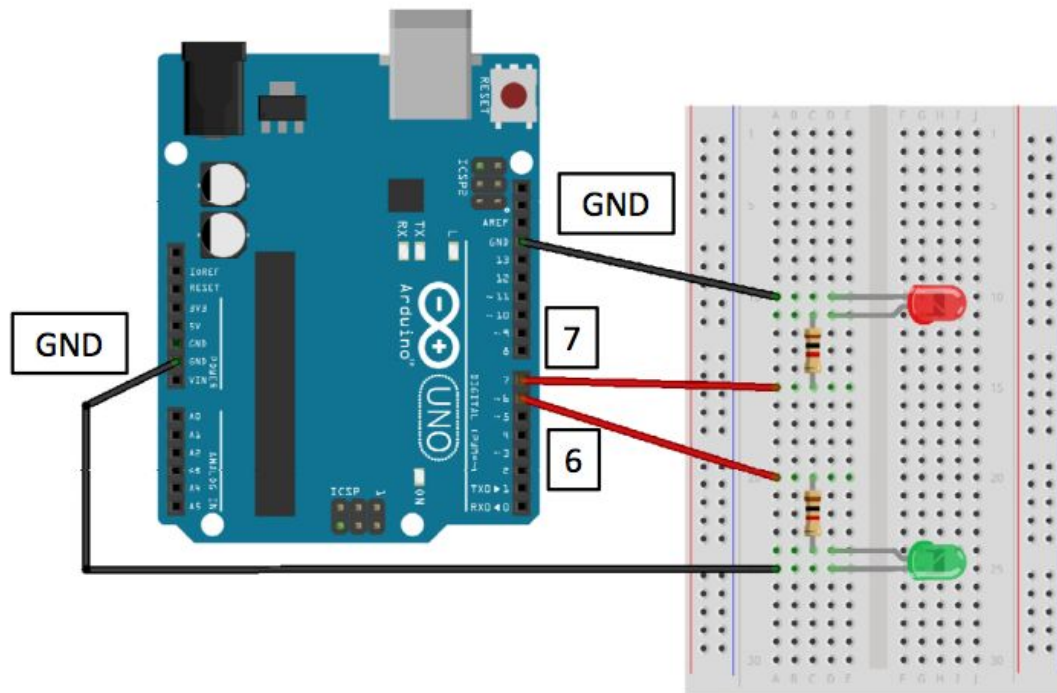
2個のLEDを使った回路を作って、プログラムで交互に点灯させなさい
(以下は回路例)

配線

LED 2個

100Ω抵抗(茶黒茶金) 2個

ジャンプワイヤ4本



練習問題2

スイッチを3回押したらLEDが光りっぱなしになるプログラムを作りなさい

【ヒント】状態を保持する変数を作ろう

例: `int count = 0;`

もしスイッチが押されたら

`count`に1を足す

`delay(?)`; // スイッチを押している間カウントしないように少し待つ

もし`count`が3になったら

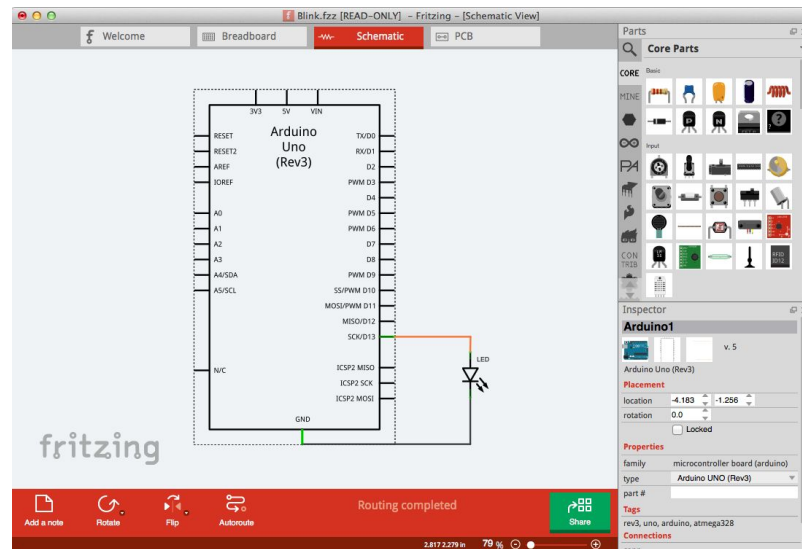
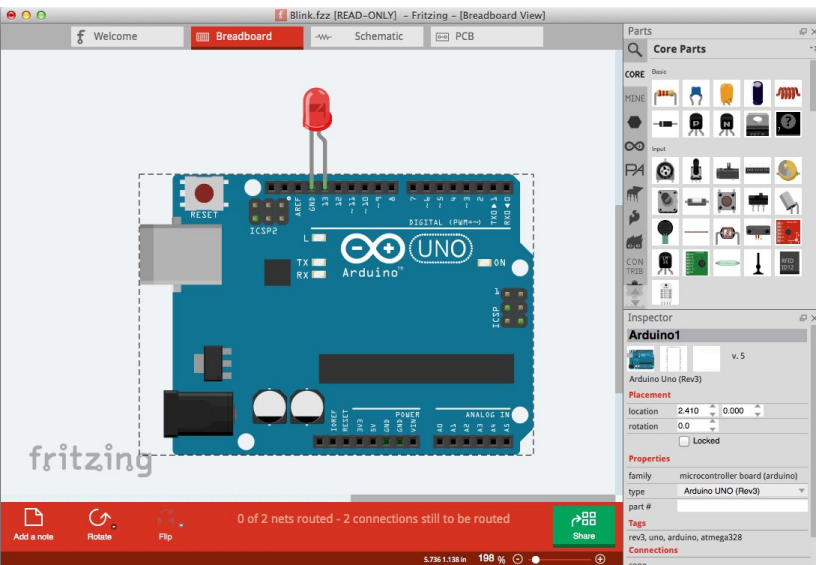
 LEDを点ける

`count`を0にする

Fritzingの準備

回路設計を行うソフトウェア。

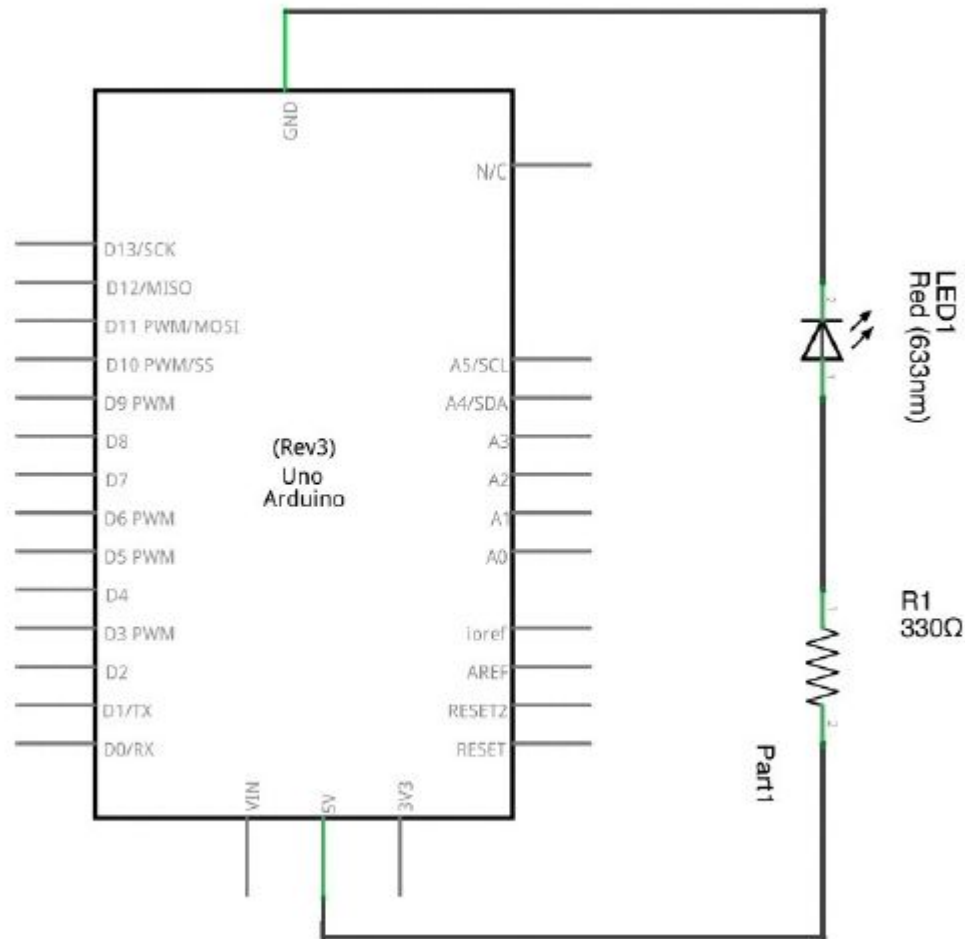
<http://fritzing.org/download/>



Fritzing

部品を配置してみよう

回路図



回路図

電源



グランド



回路図

抵抗



10k
R1



可変抵抗



503
R3



CdS



R
CDS



スイッチ



SW
PUSHSW

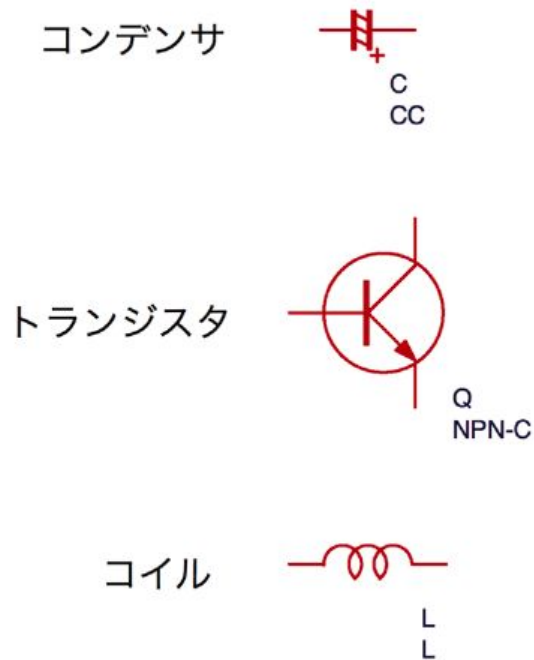


SW
SW1-2

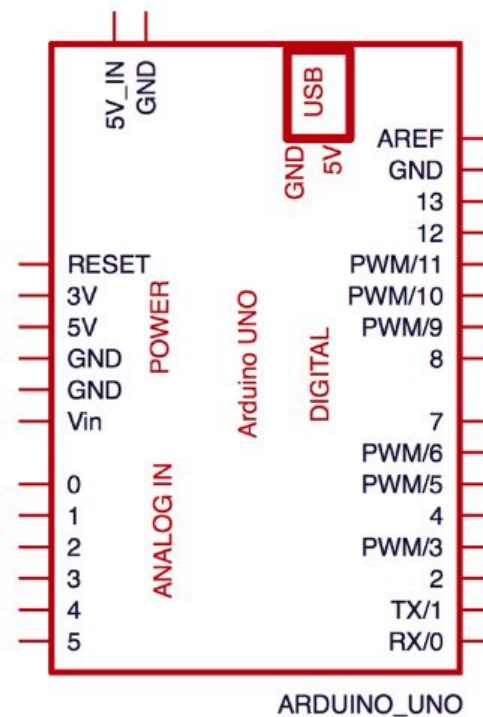


SW
SW1-1

回路図



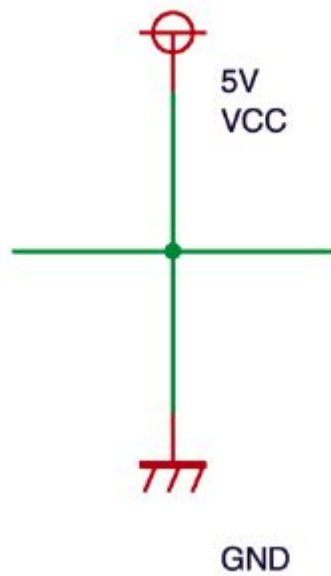
マイコンボード



単位

電気定数	記号	読み方	よく使う単位
電圧	V	ボルト	mV, V
電流	A	アンペア	mA, A
電力	W	ワット	mW, W
抵抗	Ω	オーム	Ω , k Ω , M Ω
静電容量	F	ファラッド	pF, μ F
周波数	Hz	ヘルツ	Hz, kHz, MHz

回路图



データシート

部品のスペックを記したドキュメント

<http://akizukidenshi.com/catalog/g/gI-00881/>

データシート

(1) 絶対最大定格

(Ta=25°C)

項 目	記 号	最大定格	単位
順電流	I _F	30	mA
パルス順電流	I _{FP}	100	mA
逆電圧	V _R	5	V
許容損失	P _D	120	mW
動作温度	T _{opr}	-30 ~ + 85	°C
保存温度	T _{stg}	-40 ~ + 100	°C
半田付け温度	T _{sld}	265°C 10sec 以内	

I_{FP} 条件 : パルス幅 ≤ 10ms, デューティー比 ≤ 1/10

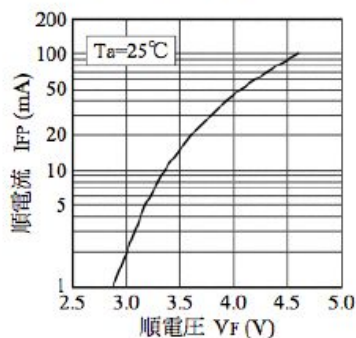
データシート

項 目		記 号	条 件	標 準	最大	単位
順電圧		V_F	$I_F=20\text{ [mA]}$	(3.6)	4.0	V
逆電流		I_R	$V_R=5\text{ [V]}$	—	50	$\mu\text{ A}$
光度		I_V	$I_F=20\text{ [mA]}$	(18000)	—	m c d
色度座標 ※	x	—	$I_F=20\text{ [mA]}$	0.31	—	—
	y	—	$I_F=20\text{ [mA]}$	0.32	—	—

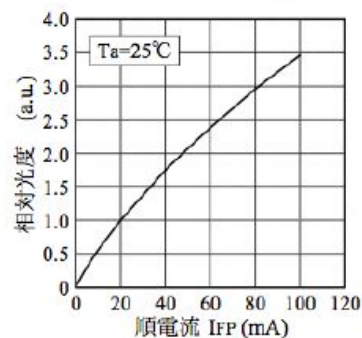
※ 色度座標は、CIE 1931 色度図に基づくものとします。

データシート

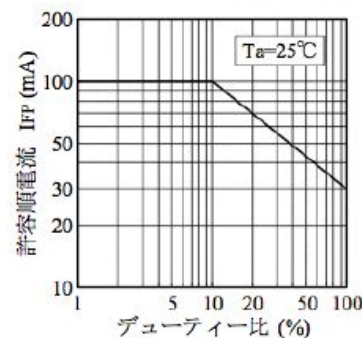
■ 順電圧－順電流特性



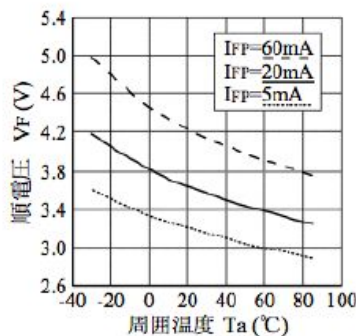
■ 順電流－相対光度特性



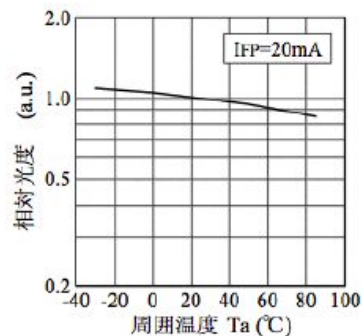
■ デューティー比－許容順電流特性



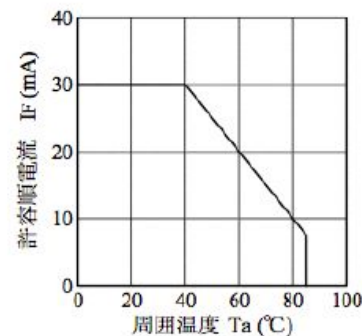
■ 周囲温度－順電圧特性



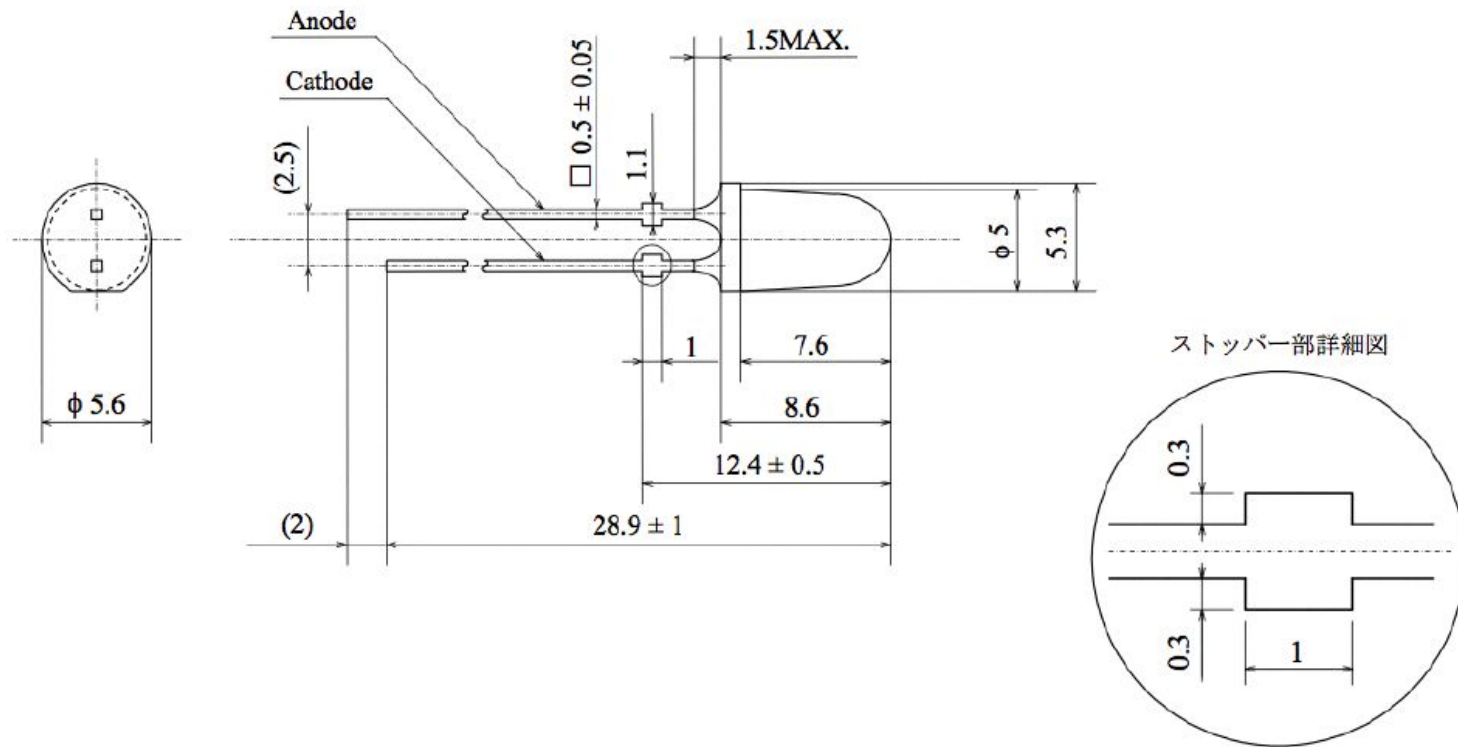
■ 周囲温度－相対光度特性



■ 周囲温度－許容順電流特性



データシート



今の状態だと、スイッチを押している間だけ光っている。

- ・クリックしたらONとOFFを切り替えるには、プログラミング的にどうするか。

```
sample> digital > statechangedetection
```





```
const int buttonPin = 2;
const int ledPin = 13;
int buttonPushCounter = 0;
int buttonState = 0;
int lastButtonState = 0;
void setup() {
  pinMode(buttonPin, INPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  ButtonState = digitalRead(buttonPin);
  if (buttonState != lastButtonState) {
```

```
    if (buttonState == HIGH) {
      buttonPushCounter++;
      Serial.println("on");
      Serial.print("number of button
pushes: ");
```

```
Serial.println(buttonPushCounter);
    }
    else {    Serial.println("off");
    }
    delay(50);
  }
  lastButtonState = buttonState;
  if (buttonPushCounter % 4 == 0) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }
}
```



次回(10/16)

セメダインの導電インクWS

用意するもの:

- ・導電インクを塗ってみたい素材やモノ
(プラスチック、フィルム、布など)、
- ・絵筆



課題

・今日の感想

