

# Arduinoの基礎



Ver.2.0

2014/6/4(水)

堀間 利彦

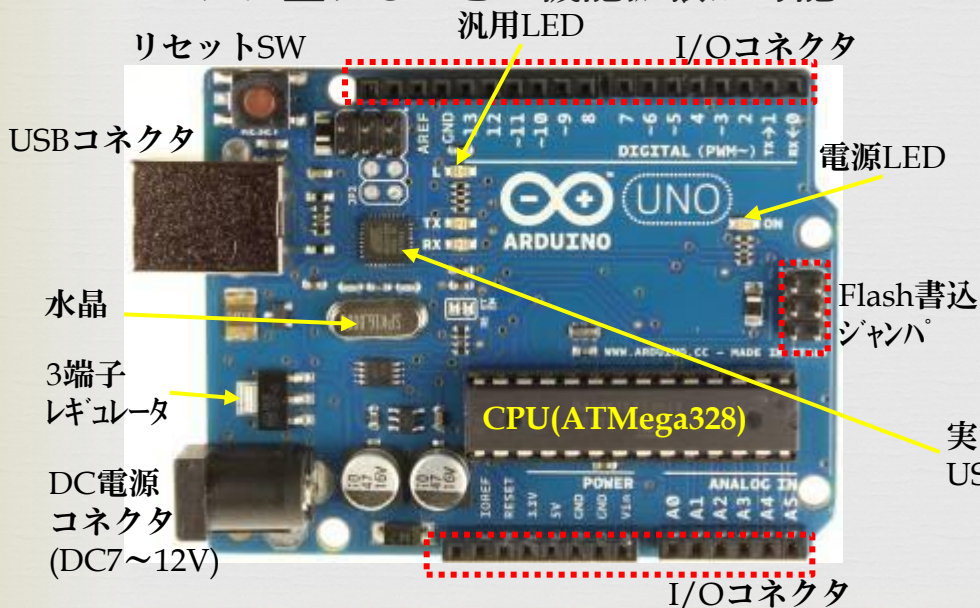
# Arduinoを使う



## Arduinoの概要

- 周辺機能一体型組み込み用マイコン(AVR)を容易に使えるよう電源、周辺回路、コネクタを備えた一体型マイコンボード
- PCでプログラム(スケッチ)を開発し、USB接続してCPUのフラッシュメモリに書込みし、ボード単体で動作させる
- オプションボード(シールド)をI/Oコネクタに重ねることで機能拡張が可能

主な仕様	
CPU	ATMega328
動作電圧	5V
入力電源電圧(推奨)	7~12V
入力電源電圧(最大)	6~20V
デジタルI/Oピン	14本(6本はPWM動作可能)
アナログ入力ピン	6本
I/Oピンの最大DC電流	40mA
3.3Vピンの最大DC電流	50mA
フラッシュメモリ	32kB(内0.5kBがブートローダ)
SRAM	2kB
EEPROM	1kB
クロック周波数	16MHz

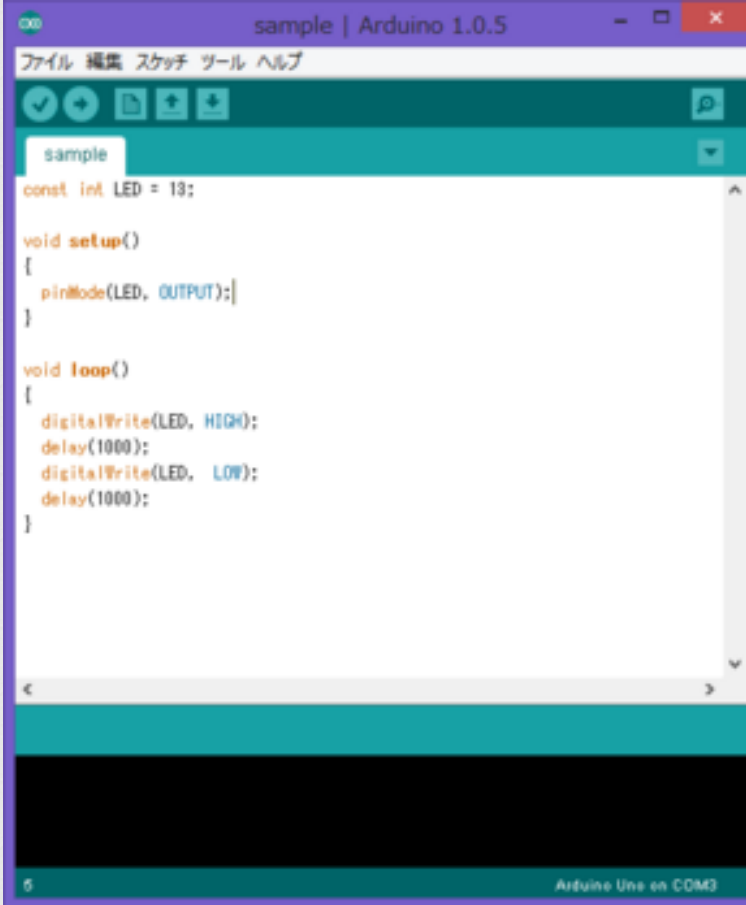


実はこれもCPU(ATMega16U)  
USB/シリアル変換に特化したプログラムが既にも書込まれている

# ArduinoIDEの使い方手順



- ☞ Arduinoプログラム開発環境をパソコンにセットアップ  
<http://arduino.cc/en/Main/Software> からPC(母艦)OSに合ったものをダウンロードし、PCにセットアップする
- ☞ IDEを起動する
- ☞ スケッチ(プログラム)を書く
- ☞ ArduinoをUSBケーブルで母艦に接続  
(IDE右下のCOMポートを確認する)
- ☞ IDメニューバーの右矢印(⇒)ボタンで  
ArduinoのCPUにプログラムを書込む  
(正常に書込めたことをメッセージで確認)
- ☞ 書込み終了と同時にArduinoはプログラムを実行開始



```
sample | Arduino 1.0.5
ファイル 編集 スケッチ ツール ヘルプ
sample
const int LED = 13;

void setup()
{
  pinMode(LED, OUTPUT);
}

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

Arduino Uno on COM3
```



# Arduinoのプログラミング



☞ C言語と共通点が多い

☞ C言語のサブセットと考えてよい

文の最後は「;」で終わる

「//」以降はコメント(無視される)

条件判断は「if(条件式) {処理内容}」等

☞ プログラム構造(C言語のmain()関数の代わりと理解)

```
void setup(){           // 初期設定内容を記述
    pinMode(xx,INPUT);   // デジタル入出力ピンxxを入力に(例)
    :
}
void loop(){            // 処理内容を記述
    :
}
```

☞ ライブラリ利用で簡単プログラミング

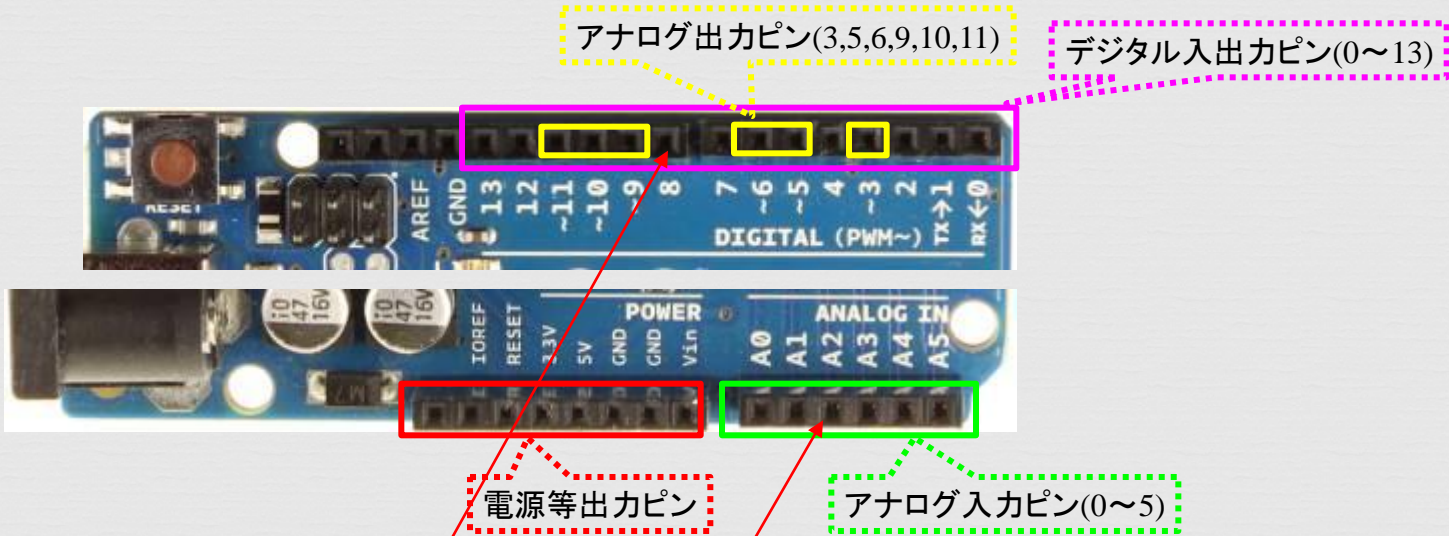
☞ 容易に利用できるよう予め設定や決まった動作をさせるプログラムがライブラリとして登録されている

☞ 第三者が作成したライブラリをネットから入手して活用できる

# ArduinoのI/Oポート



プログラムのアクセスするArduinoの入出力ピン



I/Oポートピンはプログラムからは番号でアクセスする

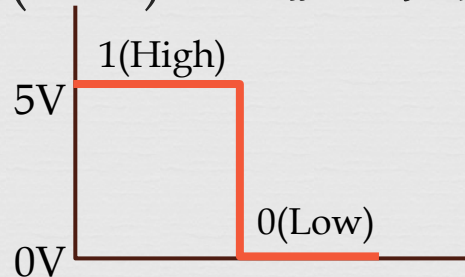
例) `pinMode(8,OUTPUT);` // デジタル入出力ピン8を出力に設定  
`voltage = analogRead(2);` // アナログピン0からの値を読む

# Arduinoから出力する



## デジタル出力

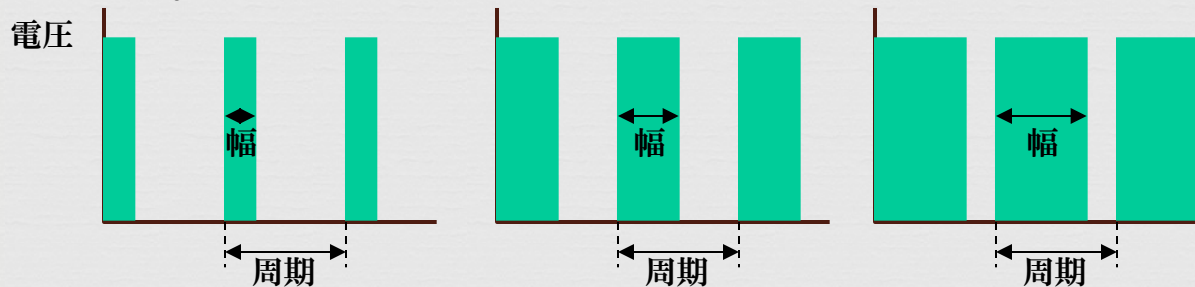
1(High) か 0(Low) の値を出力する



## アナログ出力

0~255(0xFF) の値を出力する

パルス幅変調：PWM(Pulse Width Modulation)  
Duty比を変えることで出力値を変化させる



$$\frac{\text{幅}}{\text{周期}} = \text{Duty比}$$



# 課題1：LEDを点灯する



## LEDの使い方

- データシート(部品の仕様を説明した文書)を確認(色、電流、電圧、輝度etc)
- お買い得パック、バルク品ではメーカーや型番が不明
- 中国製バッタものLEDチェッカ(大阪日本橋¥500)がちょっと便利
- アノードとカソード、明るさ、色、降下電圧等が確認可能(電圧がよほど高くなければ逆刺ししても壊れない)



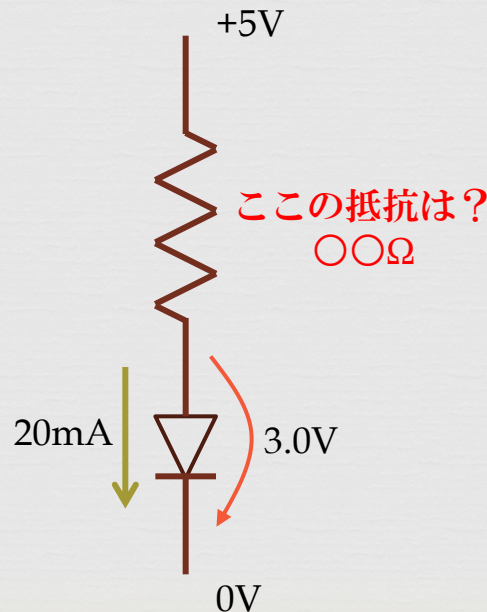
アノード(A)

足を切ったら分からなくなる  
そんなときは.....

LEDの内部構造



## 練習問題



絶対最大定格	(一瞬も超えてはダメ)
最大電圧	6.0V
最大順方向電流	40.0mA
動作温度	-20~125°C
保存温度	-40~150°C

電気的仕様	(標準的な利用時)
順方向電流	20mA
順方向電圧	3.0V
輝度	20lm

# LEDを点ける



☞ ボード上の汎用LEDを点滅させる

☞ 以下のプログラムを入力して実行させてみよう

// 【例題1】 オンボードLED点滅(デジタル出力)

```
void setup() {  
  pinMode(13, OUTPUT); // ポート13を出力に  
}
```

```
void loop() {  
  digitalWrite(13, HIGH); // ポート13をHに  
  delay(1000); // 1秒遅延  
  digitalWrite(13, LOW); // ポート13をLに  
  delay(1000); // 1秒遅延  
}
```

// 【例題2】 外付けLEDディマー(アナログ出力)

```
int brightness=0; //PWM幅  
int fadeAmmount=5; //増減幅設定
```

```
void setup() {  
  pinMode(3, OUTPUT); //ポート3を出力に  
}
```

注意  
アナログ出力は  
3,5,6,9,10,11ピンのみ

```
void loop() {  
  analogWrite(3, brightness); //PWM幅設定  
  brightness = brightness + fadeAmmount; //増減  
  
  if(brightness==0 || brightness==255){  
    fadeAmmount = -fadeAmmount; //増減反転  
  }  
  delay(30);  
}
```

例題2の回路例





# 配布パーツ



- ❧ 今後の実習をするのに必要なパーツを配布する
- ❧ 以下に配布パーツ一覧を示す

#	品名	型番	個数
1	液晶ディスプレイ	SC1602BS-B	1
2	DCモータ	FA-130RA-2270	1
3	9V乾電池	006P	1
4	電池スナップ付コード	自作	1
5	圧電スピーカ	PKM13EPYH4000-A0	1
6	タクトスイッチ	3色	3(各1)
7	半固定抵抗	10k $\Omega$ 、100k $\Omega$	2(各1)
8	温度センサ	MCP9700-E/TO	1
9	シャントレギュレータ	TL431	1
10	抵抗(1/4W)	51,100,150,270,470,750,1k,1.5k,2.7k,3.3k,4.7k,7.5k, 10k,15k,27k,47k,68k,100k,150k,200k	40(各2)
11	セラミックコンデンサ(50V)	0.1 $\mu$ F	1

# Arduinoのモニタ機能



- ☞ Arduinoと接続したパソコンの画面にArduinoから送られたデータを表示する簡易モニタ機能の使い方をマスターする
- ☞ 以下のプログラムを入力し、Arduinoに書き込み、シリアルモニターを起動させてみよう

```
// 【例題1】シリアルモニター  
int i=0;  
void setup() {  
  Serial.begin(9600); // 利用宣言  
}  
  
void loop() {  
  i++;  
  Serial.println(i); // モニタ送信  
  delay(1000); // 1秒遅延  
}
```

```
// 【例題2】アナログ入力値をモニタ表示  
int ans;  
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  ans=analogRead(0); // アナログ入力0読込  
  Serial.println(ans); // PCモニタに送信  
  delay(1000); // 1秒遅延  
}
```

例題2の回路例



# Arduinoに入力する



## ☞ デジタル入力

☞ 1(High) か 0(Low) の値を入力する



TTLレベル：トランジスタをON/OFFするベース電圧同等という意味  
0~0.8V がLow 2.0~5V がHigh  
CMOSレベル：電源電圧/2 がHigh/Lowの境目になる

## ☞ アナログ入力

☞ AD変換器によりアナログ値(0~5V)をデジタル値(0~1023(0x3FF))に変換する

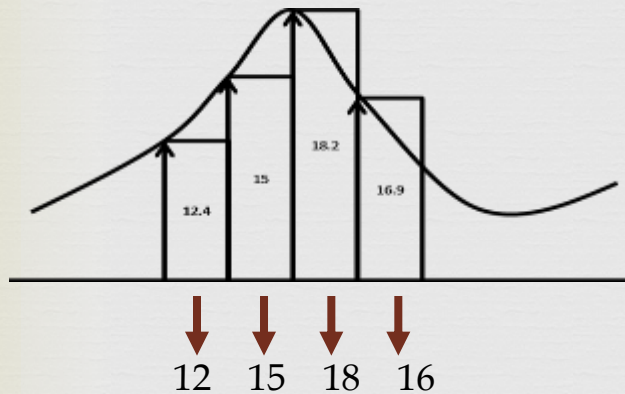
☞ AD変換については次ページに詳しく解説する



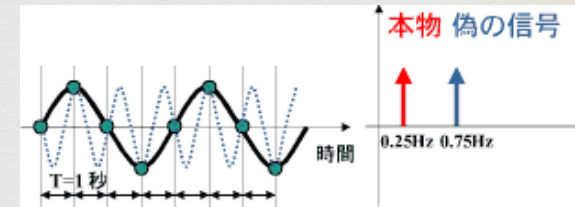
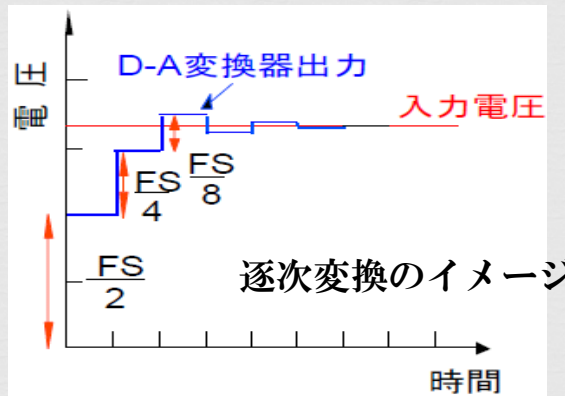
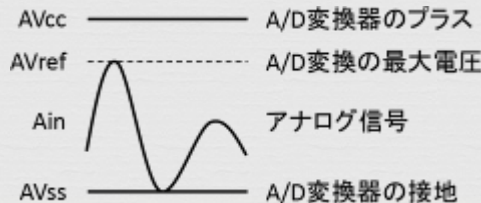
# AD変換の概要



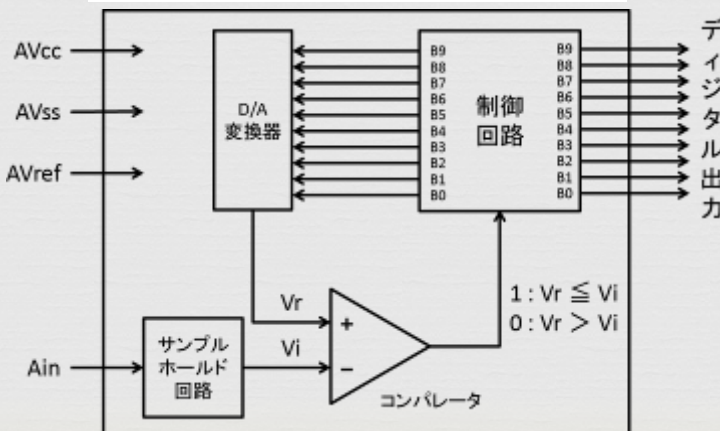
- センサなどから測定値を読み込むために利用する
- センサとは、測定対象の物理量を電圧に変換するもの
- 電圧をAD変換器によりデジタル値にして取り込む



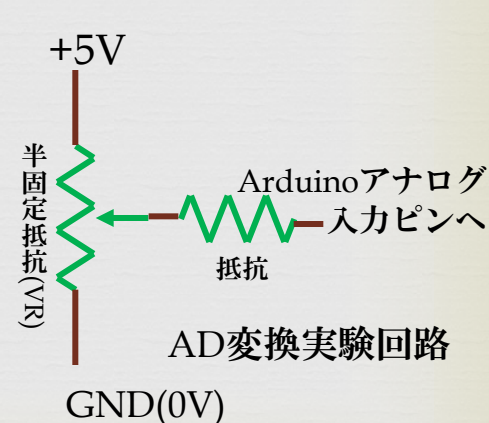
等間隔で標本化した値をデジタル値にすることを量子化という



**標本化定理**  
観測対象周波数の2倍以上の周波数でサンプリングが必要



逐次比較型AD変換器



AD変換実験回路

# センサの使い方

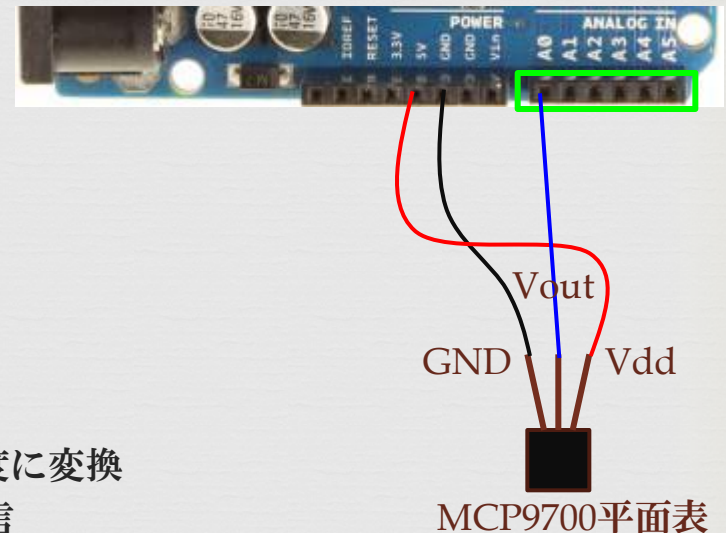


- Arduinoでセンサの測定値を取得する方法をマスターする
  - 温度センサの出力電圧から温度を取得する  
MCP9700の特性：電源5V、500mV(0°C)、10mV/°C
  - AD変換の数値と電圧の関係  
0 : 0V、1023 : 5V、102 : 500mV、1 : 0.5°C

```
// 【例題】 MCP9700で温度測定
float temperature;
void setup() {
  Serial.begin(9600);
}

void loop() {
  temperature =(analogRead(0)-102)*0.5; // 測定し温度に変換
  Serial.println(temperature,1); // PCモニタに送信
  delay(1000); // 1秒遅延
}
```

例題の回路例

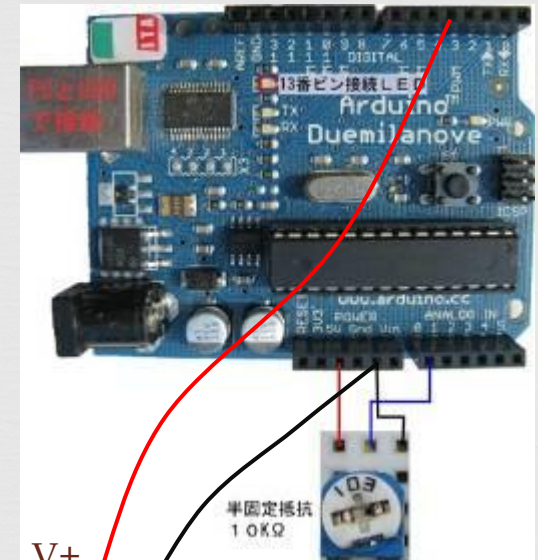


# モータを制御する(1)



- ☞ Arduinoでものを動かすためにはモータを制御する必要がある
- ☞ 最も安価なDCモータの動かし方を学ぶ
- ☞ PWMでアナログ出力を行えば速度制御ができるはず・・・正解  
以下のプログラム、以下の回路でモータを回してみよう

例題の回路例



DCモータ

```
// 【例題】半固定抵抗でDCモータ速度を制御
int ans;
void setup() {
  Serial.begin(9600);
  pinMode(3,OUTPUT);
}

void loop() {
  ans=analogRead(0)/4; // アナログ入力0読込
  Serial.println(ans); // PCモニタに送信
  analogWrite(3,ans); // PWMポート出力
  delay(1000); // 1秒遅延
}
```



# モータを制御する(2)



☞モータを回すには何かを考慮する必要があるぞ

☞前ページの回路で回らないのはなぜか

☞モータの仕様を確認してみよう(右表)

☞中国語だが、漢字だから何となく分かる  
パーツの世界、中国語の仕様には慣れよう

☞なにやら電流値が色々書いてあるぞ

☞モータの基本を理解しよう

☞モータは電流で動かす

☞必要な電流を流さないと回らない

☞仕様から始動には600mA以上の電流が必要

☞回ってしまえば負荷がなければ200mAで回り続ける

☞Arduinoのポートの仕様を確認しよう

☞Arduinoの出力ポートは20mA出力が限界  
LSIの構造上、ポートから出力できる電流、  
入力できる電流の限界値が決まっている

☞仕様に満たない場合はどうやって電流を増やせばよいのか?

☞トランジスタの増幅機能を利用する

电气性能Electrical characteristics	
空载电流 No load current:	0.20A (0.26A Max)
空载转速 No load speed:	9100 ± 10% rpm
负载电流 Load current:	0.66A (0.85A Max)
负载转速 Load speed:	6990 ± 10% rpm
負 載 Load :	6.0g. cm
负载功率 Load output :	0.43 W
效 率 Efficiency:	44%
堵转电流 Stall current:	2.2 A
堵转力矩 Stall torque:	26 g. cm
负载风叶尺寸: (客供风叶) 负载电流: 负载转速:	

# モータを制御する(3)



- Arduinoでモータを動かすためにトランジスタの電流増幅機能を利用する
- NPNトランジスタの典型的な利用方法  
注意事項：正負が反転してしまう  
(PWM幅が狭広逆の補正が必要)

// 【例題】半固定抵抗でDCモータ速度を制御

```
int ans;  
void setup() {  
  Serial.begin(9600);  
  pinMode(3,OUTPUT);  
}  
  
void loop() {  
  ans=analogRead(0)/4; // PWM値の設定  
  Serial.println(ans); // PCモニタに送信  
  analogWrite(3,255-ans); // PWMポート出力  
  delay(1000); // 1秒遅延  
}
```

例題の回路例

