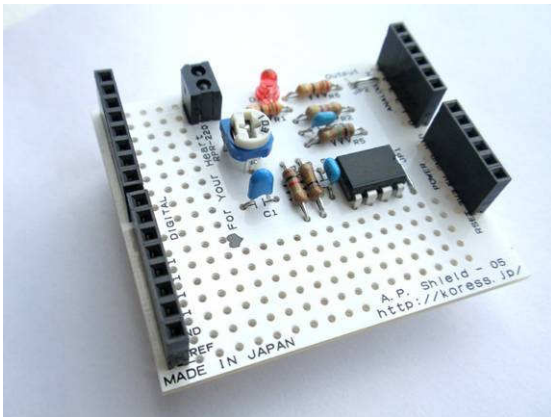


Arduino 互換機対応 心拍センサシールド

“A.P. Shield 05” by KORESS 取扱説明書



A.P. Shield (エーピーシールド) は、心拍や脈拍を使ったガジェットやメディアアートを簡単に作るための Arduino 互換機向け拡張ボードです。

※本製品は医療用途には使用できません。また、ノイズも含まれますので正確な脈拍とズレが生じる場合もあります。あらかじめご了承ください。

A.P. Shield を Arduino に接続する

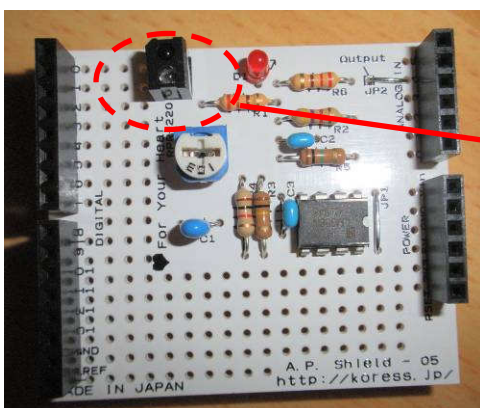
A.P. Shield のピンを、Arduino のピンソケットに静かに差し込んでください。この時、ピンが折れないように気をつけてください。1 方向にしか差し込めないようになっていますので、無理に刺すとピンを破損しますのでご注意ください。

A.P. Shield は、実際には Arduino の +5V および GND ピン、ANALOG IN の 4 番ピンを使用します。その他の DIGITAL IN 等のピンは使用しません。(蛇の目基盤に信号線が出ていますので自由に拡張可能です)

動作検証

A.P. Shield と Arduino を接続したら Arduino に AC アダプタか、USB ケーブルを接続して、電源を供給してください。電源を接続すると、LED が 2,3 秒、明るく光ります。

その後、センサー(A.P. Shield 上の四角形の黒い素子)を指で軽くトントンと叩いてみてください。LED が光ったり消えたりするはずです。この状態で心拍を読み取る準備ができています。



この黒い四角い素子がセンサーです
フォトリフレクタと呼ばれる素子です。

心拍を計測するには、人差し指の腹側を軽くセンサーの上に置いて、5秒程度安静にしてください。数秒後、脈のリズムに合わせて LED がゆったりと点滅すれば成功です。

特に冬場や寒い場所では皮膚の毛細血管が収縮している関係で心拍をうまく検出できない場合があります。暖かい環境でお試しいただくか、他の人の指でお試してください。

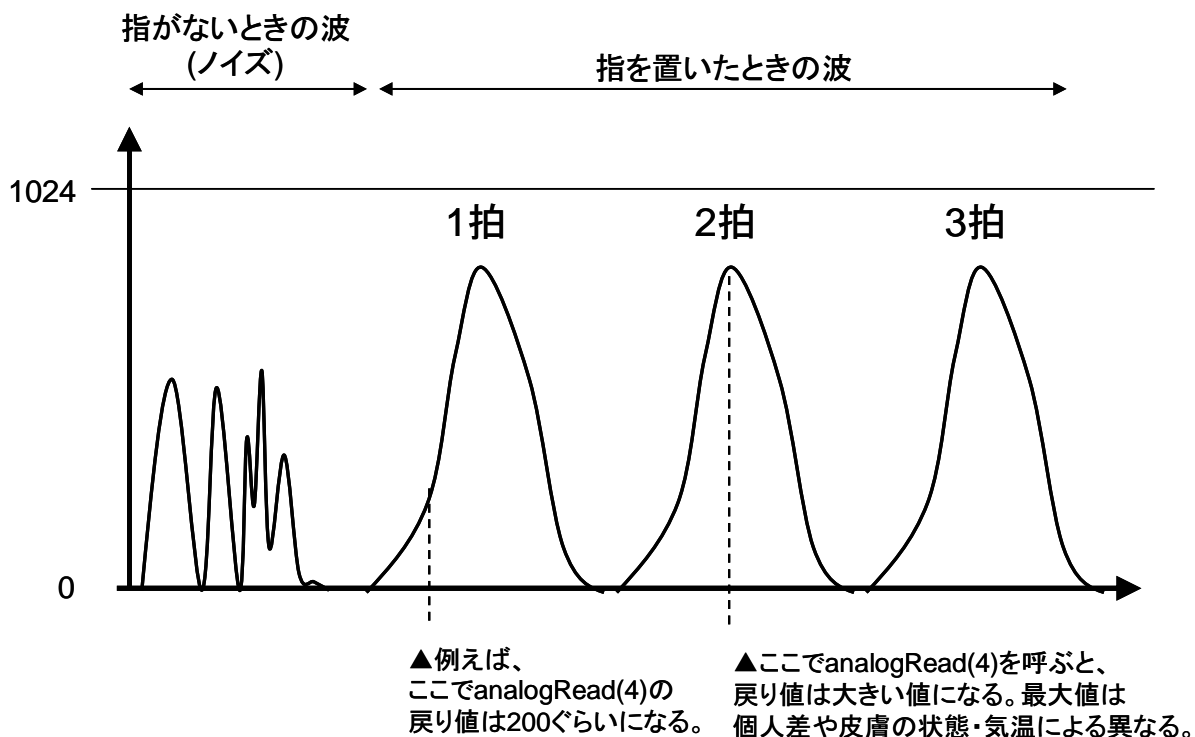
また、フォトフレクタにあまり強く指を押し付けると、反応しません。逆に弱すぎると指が不安定になり、LED がチカチカ素早く点滅してしまいます。うまい力加減を見つけてください。親指よりは人差し指、中指のほうが反応しやすい傾向があります。

また、まったく LED が光らない場合には Arduino への電源の供給等を確認してください。

Arduino でのプログラミング

A.P. Shield から心拍の情報は、Arduino で読み込むことができます。

A.P. Shield から接続されるアナログ 4 番ピンは、通常、下のグラフのような変化をします。そこで、`analogRead(4)` というメソッドを呼び出すことで、その時点のグラフの高さ(電圧)を、0 から 1024 の整数で取得することができます。ですから、例えば、この山の数と時間間隔を計算することで 1 分間あたりの心拍数をカウントすることができます。



なお、指を乗せていない時に `analogInput(4)` で数値を読み込むと、ノイズが含まれます。ノイズは大変高速かつ細かく値が変動します。指を置いた時の波は、ゆっくりとある程度規則正しく取得できます。このノイズをうまく除去して、脈拍だけを取得するようにプログラムを工夫して書く必要があります。

ノイズ除去の方法として、次のサンプルスケッチ・プログラムを参考にしてください。ソースコードは <http://gist.github.com/413934> からダウンロードできます。

```
/*
A.P. Shield 05 Demo Program
Created by @shigaku / KORESS
http://koress.jp/
http://twitter.com/shigaku
*/

int sensorPin = 4;
int ledPin = 13;
int sensorValue = 0;
int integral_plus = 0;
int integral_minus = 0;
int count = 0;
int elapse_up = 0;
int elapse_down = 0;
int ave = 0;

void setup() {
  // Arduino 標準搭載のデモ用
  pinMode(ledPin, OUTPUT);
  // デバッグ用シリアル接続開始
  Serial.begin(9600);
}

void loop()
{
  int currentValue = analogRead(sensorPin);
  int diff = (currentValue - sensorValue);
  sensorValue = currentValue;

  // 変化量が0に近く、この時点までの積分値が
  // 一定量であれば心拍の候補として扱う
  if ( diff < 10 && diff > -10 && integral_plus > 160 && integral_minus < -200 ) {
    // 立ち上がり時間とたち下がり時間を調べて、
    // 人間っぽい数値になっていれば、心拍としてみなす。
    if( elapse_up > 50 && elapse_up < 400 && elapse_down > 96 && elapse_down < 800 ) {
```

```
// ★心拍が検出されるとここに処理が移る★
count++;

// デバッグ用にシリアルコンソールに出力
Serial.print(count,DEC);

Serial.println();
}

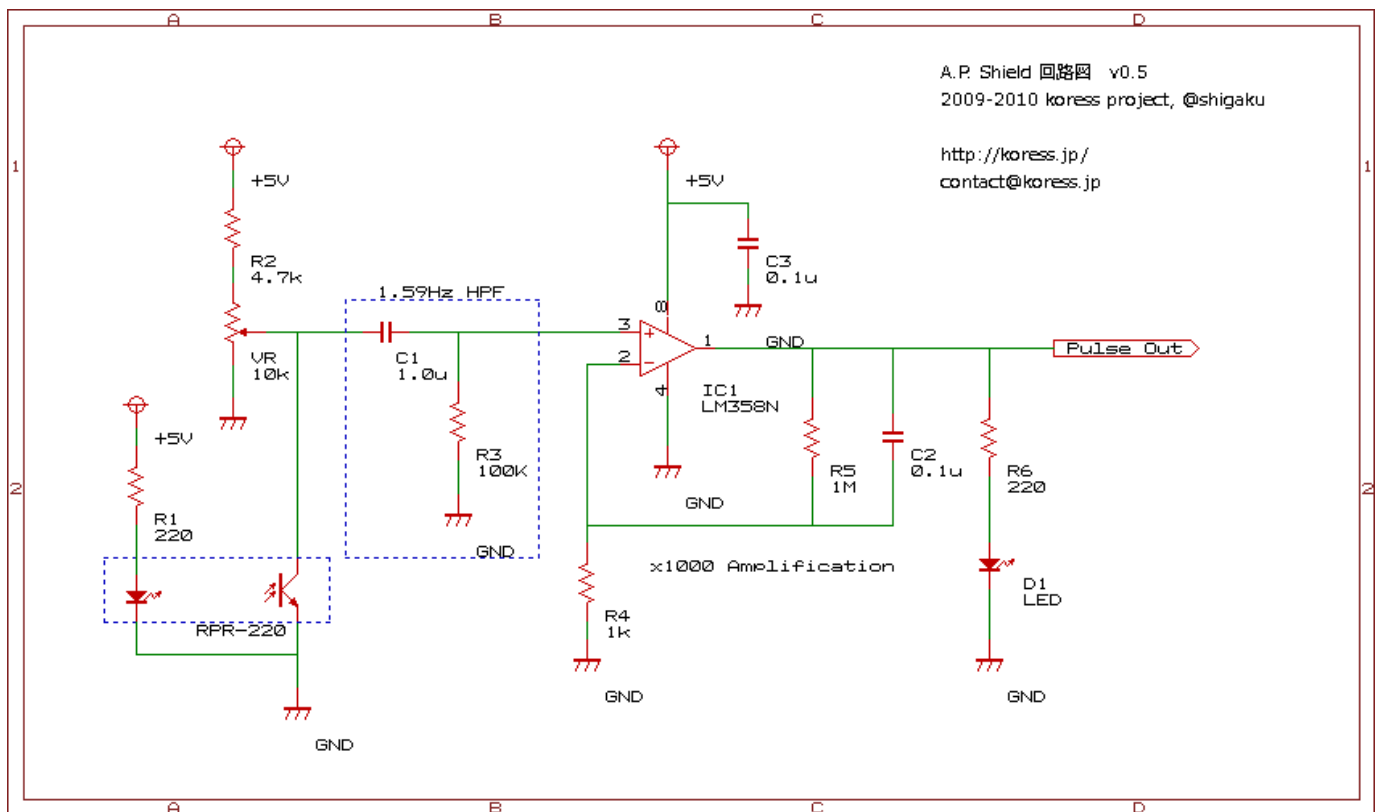
// クリア
elapsed_up = 0;
elapsed_down = 0;
integral_plus = 0;
integral_minus = 0;
} else if ( diff > 20 && diff < 280 ) {
// 波形の立ち上がりを検出
integral_plus += diff;    elapsed_up += 20;
} else if ( diff < -20 && diff > -200 ) {
// 波形の立下りを検出
integral_minus += diff;    elapsed_down += 20;
}

// 20ms ごとにサンプリングするために、待つ。
delay(20);
}
```

脈拍センシングの原理

A.P. Shield には、心拍センサとして ROHM 社の RPR-220 という素子が搭載されています。これはフォトリフレクタと呼ばれる素子で、眼に見えない赤外線を出し、その反射光によって電流を変化させる素子です。人間の指先には毛細血管が詰まっています、脈にあわせて血液が流れています。ここに赤外線を当てると、血液が流れている時・止まっている時で、反射光がかすかに変化します。この変化は、わずかな電流の変化としてフォトリフレクタで検出できます。このわずかな変化を、オペアンプという素子によって、LED を光らせるレベルまでに増幅させています。同時に、電圧はアナログ 4 番ピンから、0~1024 の値として Arduino から読み出すことができます。

回路図



技術的なお問い合わせはこちらのメールアドレスまでお願いします: contact@koress.jp

2010年10月14日版