

第6講 応用編: 赤外線センサを用いて距離を測る

1 本実習の目標

- 赤外線センサを用いて、センサと物体との距離を測定する。

2 赤外線センサを用いて距離を測る

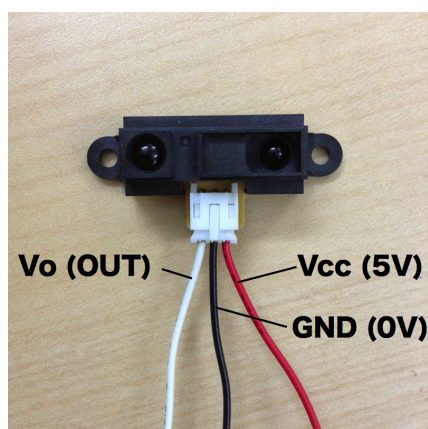
物体との距離を非接触で測定するための方式には「超音波方式」と「赤外線方式」があります。赤外線センサは、(その名の通り) 赤外線を用いて距離を測定しています。超音波センサは、物体から人体まで幅広いものの距離を検出することができます。

今回利用する赤外線センサは、大体 10cm ~ 80cm までの距離を測定することができます。赤外線センサは、外光による影響を受けやすいため、使用する環境によっては注意する必要があります。赤外線センサよりも、もう少し長い距離を測定したい場合は、超音波センサを使うと良いでしょう。

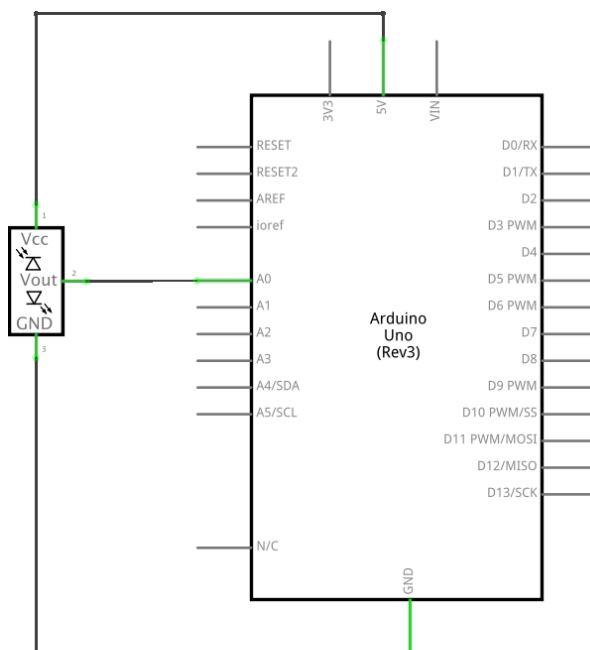
注意!

赤外線センサは、とてもデリケートで配線の向きを間違えただけでも破損してしまいます。PC に接続する前に配線が正しいかどうか、もう一度確認すること!

回路



赤外線センサ



fritzing

赤外線センサを用いた回路図

プログラム

光センサを曲げセンサに置き換えたときのように、プログラムは同じものが使えます。

3 センサの値を実際の距離にマッピングする

距離センサと後述する `map` 関数を組み合わせると、電子ものさしを作ることができます。

準備

距離センサの値を取得して、数値がどのように変化するのが観察してみましょう。また、センサの値が実際の距離とどのように対応しているのをメモしておいてください。

map 関数

Processing には `map` 関数という機能が用意されています。

```
float y = map(x, xBegin, xEnd, yBegin, yEnd);
```

`map` 関数は非常に便利なので覚えておくといよいでしょう。

`map` 関数は `x` を範囲 `xMin ~ xMax` から別の範囲 `yMin ~ yMax` へ変換する関数です。`map` 関数を利用して、センサの値を実際の距離の計測値に変換してみましょう。

```
void draw() {
  background(255);

  int val = arduino.analogRead(sensorPin);

  // 例、
  // 距離が
  // 10 cm のとき、センサの値が 768 で
  // 80 cm のとき、センサの値が 255 の場合
  float distance = map(val, 768, 255, 10, 80);

  textFont(font, 32);
  fill(0);
  text("└─val:─" + val, 32, 32);
  text("distance:─" + distance, 32, 64);
}
```