



トキのカタチ Arduino 作品集

高辻 賢司

■はじめに

このまとめ冊子は、2011 年トキのカタチで Arduino を用いた作品の使用パーツ、プログラムをまとめたものです。学生それぞれは自分の作品のことはわかるけど、他の人がどんなふうにしたのかわからないと思い、このまとめ冊子を製作することにしました。この冊子を読めば、次に違うことをしたいと思った時に、この人のこのパーツを使えそうだなという具合に参考になるはずです。プログラムを読めば、この人と私のプログラムほとんど同じじゃん！みたいな発見もあると思います。この冊子が今後みなさんの役に立つことを祈ります。

■役立つサイト補充

課題をやっていく中で発見したとても役立つサイトをまとめておきます。

・ Arduino 日本語リファレンス <http://www.musashinodenpa.com/arduino/ref/index.php>

Arduino のコードを日本語で解説しています。特に右側のライブラリのところが参考になります。

・ デバイスの接続 http://www.geocities.co.jp/arduino_diecimila/use/

簡単なデバイスの回路例や、測距センサーのノイズをとるためのコンデンサの接続まで。

・ PIC AVR 工作室 http://nekosan0.bake-neko.net/connection_rtc.html

リアルタイムクロックやステッピングモーターの接続例など。

・ なんでも作っちゃう、かも <http://arms22.blog91.fc2.com/blog-entry-284.html>

謎なネーミングのブログ。コンデンサマイクの回路はここから拝借しました。

・ FLAB <http://www.vision.cs.chubu.ac.jp/P5/html/>

Processing の講義のまとめサイト。ここで僕も勉強しました。

■Arduino ノウハウ

・ [デジタル 0,1,13 ピンは極力使わない](#)

プログラム書き込み時にこのピンを使うようで、何かに接続しているとエラーが起きる可能性があります。もし使う場合はピンを外した状態で書き込みます。

・ [GND と Vin \(5V や 12V\) を逆にしない](#)

間違えると Arduino が燃えたり、PC がシャットダウンします。ブレッドボードの青をいつも GND という風に決めておけば大丈夫です。

・ [Arduino Uno を電池で動かしたいときは 9V 電池のマイナスを GND、プラスを Vin に接続する](#)

Vin というところを 9V にすると Arduino が起動します。

間違えて逆にすると燃えます。

・ [Arduino Pro mini を電池で動かしたいときは 9V 電池のマイナスを GND、プラスを RAW に接続する](#)

RAW が Arduino Uno でいうところの Vin ピンです。

逆にすると煙がのぼります。これはこれで面白いです。

・ [Arduino の裏側を絶縁する](#)

机などが帯電していると Arduino がショートして壊れる可能性があります。

ビニールテープなどで絶縁するとこれを防げます。

・ [うまくいかないときは基本に立ち戻る](#)

アナログピンのセンサの値をシリアル通信で出力するだけのプログラムや、正転反転をただ繰り返すだけのプログラムなど、基本的なプログラムを使って、ここのパーツが一応動くことを確認することが大事です。

いきなり最終型のプログラムを書くよりも確実に作品が完成します。急がば回れです。



時の再生機

安住 仁史 安見 卓志

■使用パーツ

- ・ Arduino UNO
- ・ ステッピングモーター (480 ステップで一周) (秋月で 250 円 <http://goo.gl/vdZAI>)
- ・ タミヤのラダーチェーン&スプロケット
- ・ ネオジウム磁石 (Φ10,h5)
- ・ 砂鉄
- ・ シナノキ
- ・ アクリル (厚 3)
- ・ LED8 個 (電球色、3.2V) (千石で 100 個 1000 円)
- ・ ブレッドボード
- ・ 自由継手 (スプロケットの軸とモータの出力軸の連結に使用)
- ・ そこら辺で拾った真鍮製 (?) のギア (操作ノブ)
- ・ 可変抵抗

■機構補足

チェーンに取り付けたネオジウム磁石が内部で循環し、ガイドによって磁石が砂鉄の入る容器の底部に沿って移動することで砂鉄の動きを実現しています。

Comments

ノブの回転角を読み取るために可変抵抗をセンサとして使っているところが 1 つポイントです。

シナノキを介してもネオジウム磁石が砂鉄を引っ張るようにシナノキはものすごく薄くしたんじゃないでしょうか。

展示中まったく壊れていなかったのがすごいと思いました。

```

#include <Stepper.h>
#define STEPS 480 // 使用するモータのステップ数
#define AMBIENT_A 6 // 左側
#define AMBIENT_B 9 // 右側
#define AMBIENT_C 5 // 中央
#define val1 0 // 左送り
#define val2 30 // 左
#define val3 150 // 停止
#define val4 230 // 右
#define val5 650 // 右送り
int i=0;
int j=0;
int k=0;
int l=0;
int m=0;
int n;
int o;
int mode=0;//0 で動作
int prev;

// ピン番号を指定して stepper
// クラスのインスタンスを生成
Stepper stepper(STEPS, 1, 2, 3, 4);

void setup() {
  stepper.setSpeed(20); // スピードを 20RPM に
  pinMode(AMBIENT_A,OUTPUT);
  pinMode(AMBIENT_B,OUTPUT);
  pinMode(AMBIENT_C,OUTPUT);
}

void loop()
{
  int val = analogRead(0); // センサの値を取得

  if(mode == 1 && val!=prev){
    mode = 0;
  }

  if(val < val2 && mode == 0){ // モード 1
    stepper.setSpeed(40);
    stepper.step(440);
    if(i%3 == 0){
      digitalWrite(AMBIENT_A,LOW);
      digitalWrite(AMBIENT_B,HIGH);
      digitalWrite(AMBIENT_C,LOW);
    }
    else if(i%3 == 1){
      digitalWrite(AMBIENT_A,LOW);
      digitalWrite(AMBIENT_B,LOW);
      digitalWrite(AMBIENT_C,HIGH);
    }
    else {
      digitalWrite(AMBIENT_A,HIGH);
      digitalWrite(AMBIENT_B,LOW);
      digitalWrite(AMBIENT_C,LOW);
    }
  }
  if(i>50){
    mode=1;
  }
  i++;
  j=0;
  k=0;
  l=0;
  m=0;
}

```

```

else if(val < val3 && mode == 0){ // モード 2
  stepper.setSpeed(20);
  stepper.step(40);
  if(j%2==0){
    digitalWrite(AMBIENT_C,LOW);
  }
  else{
    digitalWrite(AMBIENT_C,HIGH);
  }
  n=j/2;
  analogWrite(AMBIENT_B,255-n);
  analogWrite(AMBIENT_A,j/2);
  delay(750);
  if(j>512){
    mode = 1;
  }
  j++;
  i=0;
  k=0;
  l=0;
  m=0;
}

else if(val < val4 || mode == 1){ // モード 3
  stepper.setSpeed(0);
  stepper.step(0);
  if(k%2 == 0){
    digitalWrite(AMBIENT_A,LOW);
    digitalWrite(AMBIENT_B,LOW);
    digitalWrite(AMBIENT_C,LOW);
  }
  else{
    digitalWrite(AMBIENT_A,LOW);
    digitalWrite(AMBIENT_B,LOW);
    digitalWrite(AMBIENT_C,HIGH);
  }
  delay(1000);
  k++;
  i=0;
  j=0;
  l=0;
  m=0;
}

else if(val < val5 && mode == 0){ // モード 4
  stepper.setSpeed(20);
  stepper.step(-40);
  if(l%2==0){
    digitalWrite(AMBIENT_C,LOW);
  }
  else{
    digitalWrite(AMBIENT_C,HIGH);
  }
  analogWrite(AMBIENT_B,l/2);
  o=l/2;
  analogWrite(AMBIENT_A,255-o);
  delay(750);
  if(l>512){
    mode = 1;
  }
  l++;
  i=0;
  j=0;
  k=0;
  m=0;
}

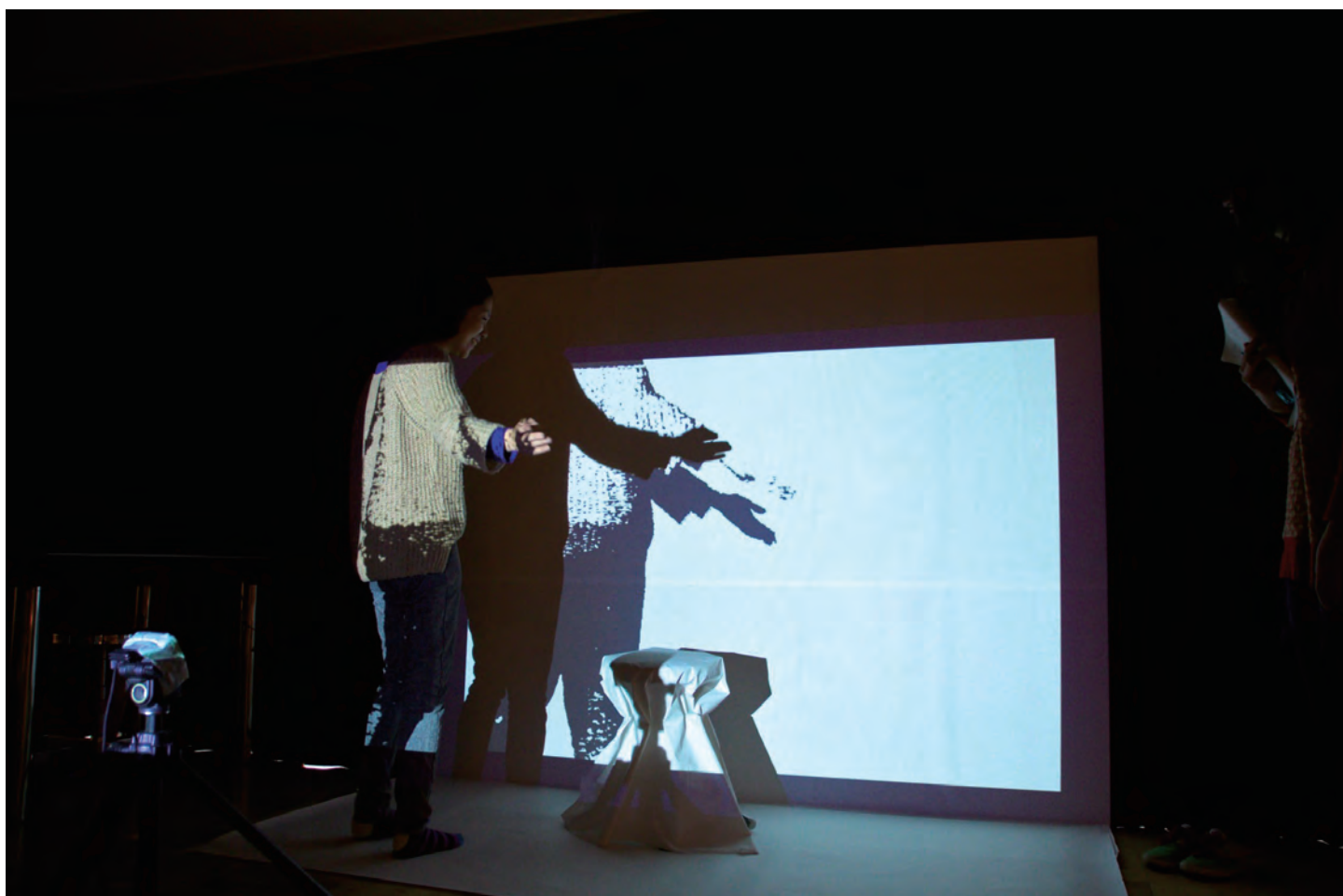
```

```

else if(mode == 0){ // モード 5
  stepper.setSpeed(40);
  stepper.step(-440);
  if(m%3 == 0){
    digitalWrite(AMBIENT_A,HIGH);
    digitalWrite(AMBIENT_B,LOW);
    digitalWrite(AMBIENT_C,LOW);
  }
  else if(m%3 == 1){
    digitalWrite(AMBIENT_A,LOW);
    digitalWrite(AMBIENT_B,LOW);
    digitalWrite(AMBIENT_C,HIGH);
  }
  else {
    digitalWrite(AMBIENT_A,LOW);
    digitalWrite(AMBIENT_B,HIGH);
    digitalWrite(AMBIENT_C,LOW);
  }
  if(m>50){
    mode=1;
  }
  i=0;
  j=0;
  k=0;
  l=0;
  m++;
}

int prev=val;
}

```



SECOND SPACE

生永 麻衣

■使用パーツ

- Mac
- Processing (パーツではない)
- web カメラ
- プロジェクタ

Comments

やりたいことを直感的にわかってもらうための変数の設定をもう少し擦り合わせたかったです。
カメラの画角やプロジェクタの画角をもう少し検討してから設置するともっとやりやすかったのかもしれない。
これで影絵のパフォーマンスとかしたら面白そう。

```

// 先頭行
import fullscreen.*;
FullScreen fs;
import processing.video.*;
Capture video;

int w=800; //mac screen size
int h=600;

//Start 画面ピクセル色を保存するための配列を用意
color[] StColor=new color[w*h];
// 許容値の変数：50 に設定しておく
int tolerance=50;
//glay
float glay=10;
//delaytime
int delaytime=2000;
int flag = 0;
int max_delay=2000;
int min_delay=200;
int delta=50;
int i=0;
//float x,y; // 座標値の変数
//float filterX,filterY; // フィルタをかけた座標値の変数
boolean movement=false; // 動体の有無のフラグ
long cur_time =0;
long old_time=0;
//setup
void setup(){
  size(w, h);
  frameRate(10);
  fs = new FullScreen(this);
  fs.enter();
  noCursor(); // オプション：マウスカーソルを消す場合
  background(255);
  video = new Capture(this, w, h);
  rectMode(CENTER);
  noStroke();
  loadPixels();
}

void draw(){
  cur_time = millis();
  if((cur_time-old_time)>delaytime){
    if(video.available()){
      video.read();
      video.loadPixels();
      loadPixels(); // 画面内ピクセルをロードしておく
      movement=false; // 動体のフラグを false に戻しておく
      for(int i=0;i<w*h;i++){

```

```

// 前回と今回の画面のピクセルの各色の差を求める
float difRed=abs(red(StColor[i])-red(video.pixels[i]));
float difGreen=abs(green(StColor[i])-green(video.pixels[i]));
float difBlue=abs(blue(StColor[i])-blue(video.pixels[i]));
// 色の差が許容値以上の場合（動体がある場合）
if(difRed>tolerance && difGreen>tolerance && difBlue>tolerance){
  movement=true; // 動体有りのフラグを true にしておく
  pixels[i]=color(0,0,0); // そのピクセルを black にする
}
}
if(movement==true){
  updatePixels(); // 画面内ピクセルをアップデート
}
}

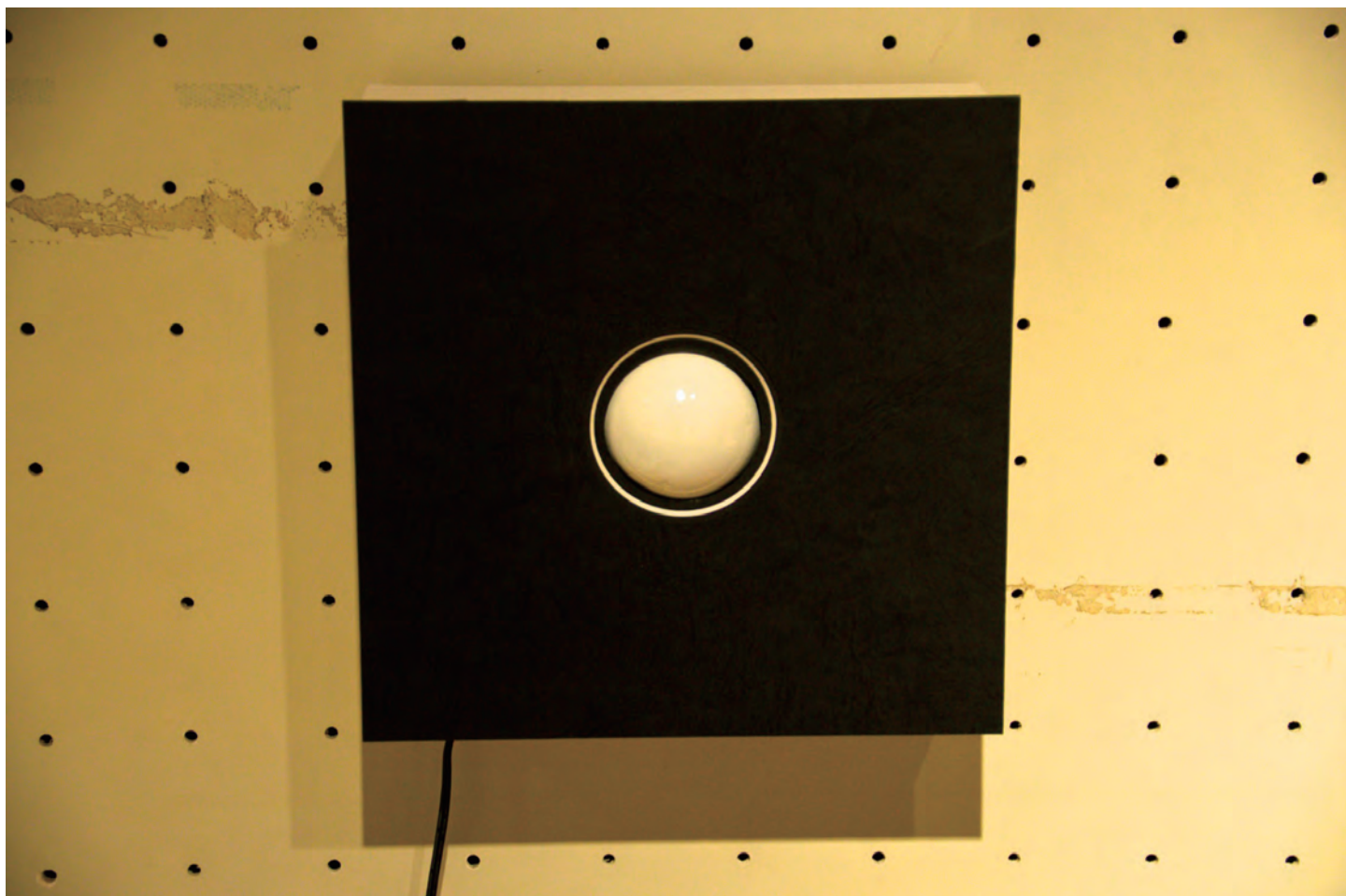
if(flag==0){
  if(delaytime > min_delay){delaytime = delaytime - delta;}
  if(delaytime == min_delay){flag = 1;}
}

if(flag ==1){
  if(delaytime < max_delay){
    delaytime = delaytime + delta;}
  if(delaytime ==max_delay){
    flag=0;}
}
old_time = cur_time;
}

//black to glay to white
for(int i=0;i<w*h;i++){
// 前回と今回の画面のピクセルの各色の差を求める
float glayRed=red(pixels[i])+glay;
float glayGreen=green(pixels[i])+glay;
float glayBlue=blue(pixels[i])+glay;
pixels[i]=color(glayRed,glayGreen,glayBlue);
}
updatePixels(); // 画面内ピクセルをアップデート
}

//mouse press reset
void mousePressed(){
  video.loadPixels();
  arraycopy(video.pixels,StColor);
  background(255);
}

```

EXPERI BALL

岩田 創太郎

■使用パーツ

- Arduino Uno
- ブレッドボード
- 銅板（タッチセンサ用）
- DC モーター
- ギアボックス
- モータードライバIC（たぶん 7291）
- 10Ω抵抗
- 5 MΩ抵抗

Comments

ノアポの岩田くん。いい写真がなかったですごめんなさい。→で許してください。
プログラムの静電容量の変化でタッチを感知しているのがポイント。
ハードが単純なので量産すれば面白かったかも。
どうしても目玉を触りたくなってしまうのは僕だけでしょうか。



```

#define LED 13
#define motor1 10
#define motor2 11

int f=0;//f がタッチセンサの値
int old_f=0;// 過去の f 値
int val=600;// しきい値
int count=0;// 触った回数
int counter=1;// モータが回るまでの回数

void setup(){
  // モニタリングのためシリアル通信開始
  Serial.begin(9600);
  //8 番ピンをデジタル出力
  pinMode(8,OUTPUT);
  //9 番ピンをデジタル出力
  pinMode(9,INPUT);
  //LED 点灯用に 13 番ピンをデジタル出力
  pinMode(LED,OUTPUT);
  pinMode(motor1,OUTPUT);
  pinMode(motor2,OUTPUT);
}

void loop(){
  // 静電容量変化量の変数を用意
  int a=0;
  //8 番ピンを HIGH で出力
  digitalWrite(8, HIGH);
  // 指が触れたとき 9 番ピンが HIGH になるまでをカウント
  while (digitalRead(9)!=HIGH){
    // カウントする
    a++;
  }
  delay(1);
  //8 番ピンを LOW にする
  digitalWrite(8, LOW);
  // 値を滑らかにするフィルタ式
  f+=(a-f)/2;
  // モニタリング：フィルタ値を出力

  if((f>val)&&(old_f<val)){
    count++;// 触った回数を 1 つ増やす
    delay(100); // バウンス除去
  }

  //Serial.println(counter);//counter という変数を表示
  Serial.println(f);//f という変数を表示

  old_f=f;

```

```

// モーター回転のプログラム
if(counter==count){// カウンターの数値まで触ったら

  digitalWrite(motor1,HIGH);
  digitalWrite(motor2,LOW);
  delay(250);//2 秒正転

  digitalWrite(motor2,HIGH);
  digitalWrite(motor1,HIGH);
  delay(3000);//3 秒止める

  digitalWrite(motor2,HIGH);
  digitalWrite(motor1,LOW);
  delay(250);//2 秒反転

  digitalWrite(motor2,HIGH);
  digitalWrite(motor1,HIGH);
  delay(1000);
  digitalWrite(motor2,LOW);
  digitalWrite(motor1,LOW);// 止める

  counter=counter+1;//counter を増やす
  count=0;// 触った回数を初期化

}
}

```




concord LAMP

宇崎 弘美

■使用パーツ

- Arduino Pro mini
- 100W 40A 白熱灯
- 電球ソケット & コンセント
- ECM C9767 コンデンサマイク
- IC LM386N-1 325mw オーディオアンプ
- オーディオコンデンサの $1\mu\text{F}$ と $10\mu\text{F}$
- cds+LED アナログフォロカブラ MI0202CL
- トライアック万能調光器

Comments

呼吸をセンシングするための高感度マイクを作るのにまず苦勞をしていました。

スイッチサイエンスからアンプ付きのコンデンサマイクが 800 円くらいででていたのでそれを使ったらもっと早くできたかも？

そして肝心の 100V の電球をふわふわ光らせるのも大変苦勞したようです。三石くんが。

大事なパーツがフォロカブラで、マイコンから抵抗を制御することができます。

プログラムはマイクのノイズを補正するために、一度に 25 個値をとってきて平均をとっています。(25 個もサンプリングする必要があるのかな？)

Arduino を 2 つも燃やしてしまうとは恐れ入りました。

```
#define TIME 25
#define DELAY 30
#define BORDER 630 // 息を吹きかけた時の最小値
#define MAX 1023
#define LED 3
```

```
int value = 0;
int duty = 0;
int i = 0;
int voice[TIME];
int sensorValue = 0;
```

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

```
void loop(){
  // センサの値を 25 個格納
  for(i=0; i < TIME; i++){
    voice[i] = analogRead(0);
    sensorValue += voice[i];
  }
```

```
  // センサの値を 25 回の平均値にする
  sensorValue = sensorValue/ TIME ;
  //Serial.println(sensorValue);
```

```
  // 息を吹きかけていたら duty を計算
  if(sensorValue > BORDER){
    duty = map(sensorValue, BORDER , MAX , 0 , 255);
    //Serial.println("ENTERED");
```

```
  //duty が 10 以上、250 以下だったら
  if(duty > 10 && duty < 250){
    delayMicroseconds(200);
    delayMicroseconds((255-duty)*38);
```

```
  // フォトリライアック ON
  digitalWrite(LED,HIGH);
  delayMicroseconds(100);
  // フォトリライアック OFF
  digitalWrite(LED,LOW);
}
```

```
  //duty が 10 以下だったら消す
  if(duty <= 10 ){
    digitalWrite(LED,LOW);
  }
```

```
  //duty が 250 以上だったら MAX で光らせる
  else{
    digitalWrite(LED,HIGH);
  }

}
```

```
  // 息を吹きかけていなかったら
  else{
    digitalWrite(LED,LOW);
  }
```

```
  // 分解能を設定 50 だと 1 秒間に 20 回
  delay(50);
}
```



Cure pillow

岡本 征子

■使用パーツ

- ・ Arduino Uno
- ・ フルカラー LED15 個くらい
- ・ 圧力センサ
- ・ 抵抗いくつか

Comments

プロトタイプは早々に出来ていましたが、ハンダ付けに入ってからが大変でした。

特に圧力によって出したい色をグラデーションで変えていくのが難しかったみたいです。

中村くんは3次関数を使っていました。胸を打たれますね。

電源を入れると赤のLEDが光ってしまう理由は0,1,13番ピンのどれかを使っていたからで、これらのピンは書き込み用に使うので極力使わないほうがいいということがわかりました。

```

#define SENSOR 0 // センサ
#define GREEN 11 //LED が接続されたピン (13 番ピンは使わないようにして)
#define RED 9
#define BLUE 10

#define STARTRED 30
#define STARTGREEN 70
#define STARTBLUE 75

#define FINALRED 255
#define FINALGREEN 0
#define FINALBLUE 30

#define x_GREEN 20
#define x_BLUE 10
#define x_RED 25

#define y_GREEN 5
#define y_BLUE 0

int delaytime = 30; // delay する時間の長さ
int val = 0; // センサの値
int intensity = 0; //LED の明るさ

//value 確認用
int val_red =0;
int val_green =0;
int val_blue=0;

void Write_OFF(){
  analogWrite(RED, 0);
  analogWrite(GREEN, 0);
  analogWrite(BLUE, 0);
}

void setup() {
  // declare pin 9 to be an output:
  pinMode(GREEN, OUTPUT);
  pinMode(RED, OUTPUT);
  pinMode(BLUE, OUTPUT);
  Serial.begin(9600);
}

void loop(){
  val = analogRead(SENSOR); // センサの値を読み取る
  intensity= map(val,0,1023,0,255);
  //value 確認用
  val_red=0.0004*(intensity-8)*(intensity-80)*(intensity-180);
  val_green=-0.004*(intensity-65)*(intensity-243);
  val_blue=-0.004*(intensity+80)*(intensity-150)-10;

  if(val_red>255){val_red=255;}
  if(val_red<0){val_red=0;}
  if(val_green>255){val_green=255;}
  if(val_green<0){val_green=0;}
  if(val_blue>255){val_blue=255;}
  if(val_blue<0){val_blue=0;}

  if(intensity>5){
    analogWrite(RED, val_red);
    analogWrite(GREEN, val_green);
    analogWrite(BLUE, val_blue);
  }
  else{
    Write_OFF();
  }
  //Serial.println(intensity);
}

```




?

尾形 朋美

■使用パーツ

- ・ Arduino Uno
- ・ LED 電球
- ・ ブレッドボード
- ・ 測距センサー (20~150cm ?)
- ・ トライアック万能調光器
- ・ フォトカプラ
- ・ ??

Comments

白熱灯と違って LED 電球はほわーっと光らせるのが難しいみたいです。
Arduino クラッシャー最後の刺客。


```

#define AC 10
#define MIN 80
#define MAX 180

#define BORDER 0.9

#define TIME 10
#define DELAY 20

int value;
int i;
float analogvalue;
float analogvolt;
float distance;

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

void measureDistance(){
  analogvalue = 0;
  //A0 番の電圧を測定
  for(i=0;i < TIME; i++){
    analogvalue += analogRead(0);
  }
  analogvalue = analogvalue/TIME;

  //10bit で出てくる analogvalue の値に 5/1024 を掛けて変換
  analogvolt=analogvalue*5/1024;
  if((analogvolt>0.4) && (analogvolt<=0.6)){
    distance = (analogvolt -0.93333)/(-0.0067);
  }
  else if((analogvolt>0.6) && (analogvolt<=0.7)){
    distance = (analogvolt -1.1)/(-0.001);
  }
  else if((analogvolt>0.7) && (analogvolt<=0.9)){
    distance = (analogvolt -1.5)/(-0.002);
  }
  else if((analogvolt>0.9) && (analogvolt<=1.1)){
    distance = (analogvolt -2.1)/(-0.004);
  }
  else if((analogvolt>1.1) && (analogvolt<=1.3)){
    distance = (analogvolt -2.1)/(-0.004);
  }
  else if((analogvolt>1.3) && (analogvolt<=1.6)){
    distance = (analogvolt -2.5)/(-0.006);
  }
  else if((analogvolt>1.6) && (analogvolt<=2.35)){
    distance = (analogvolt -3.85)/(-0.015);
  }

```

```

    else if((analogvolt>2.35) && (analogvolt<=2.75)){
      distance = (analogvolt -4.35)/(-0.02);
    }
    else if((analogvolt>2.75) && (analogvolt<=2.95)){
      distance = (analogvolt -4.35)/(-0.02);
    }
    else if((analogvolt>2.95) && (analogvolt<=3.2)){
      distance = (analogvolt -4.7)/(-0.025);
    }
  }

  void ledFade(){
    for(i=MIN;i<MAX;i++){
      analogWrite(AC, i);
      delay(DELAY);
    }
    for(i=MAX;i>MIN;i--){
      analogWrite(AC, i);
      delay(DELAY);
    }
  }

  void loop(){
    measureDistance();

    if(analogvolt > BORDER){
      Serial.println("LED");
      ledFade();
      value = 1;
    }
    else{
      if(value == 1){
        for(i=MIN;i>=0;i--){
          analogWrite(AC, i);
          delay(100);
          value = 0;
        }
      }
    }
  }

```



BLOCK WAVES

蒲田 勇樹

■使用パーツ

- ・ Arduino Uno
- ・ ブレッドボード
- ・ ステッピングモーター
- ・ MP4401 (ステッピングモーター用 IC)
- ・ シングルギヤボックス (4 速タイプ)
- ・ スチールピニオンギヤ 48 ピッチ (20T と 40T を、それぞれモーターとギヤに)
- ・ タッチセンサ用抵抗 (1MΩ)
- ・ タッチセンサ用鉄板 (0.3mm 厚)
- ・ 9V 電池
- ・ ステンレス丸棒 (φ6mm と φ5mm)
- ・ ゴムプーリーベルト
- ・ 樹脂製歯車 (プーリーとして改造)
- ・ プラ板 (0.5mm 厚)
- ・ ABS 樹脂丸棒 (φ10mm)
- ・ ケント紙
- ・ 木材多数

Comments

結構重たいものをステッピングモーターで回せるかが不安でしたがちゃんと回りました。

展示中早々に回らなくなったのは電池で回していたからです。ずっと回る系は AC アダプターのほうがよかったかもしれません。

プログラムは岩田くんとほぼ同じタッチ検出プログラムと、ずっと回し続ける命令のみ。

```
#include <Stepper.h>

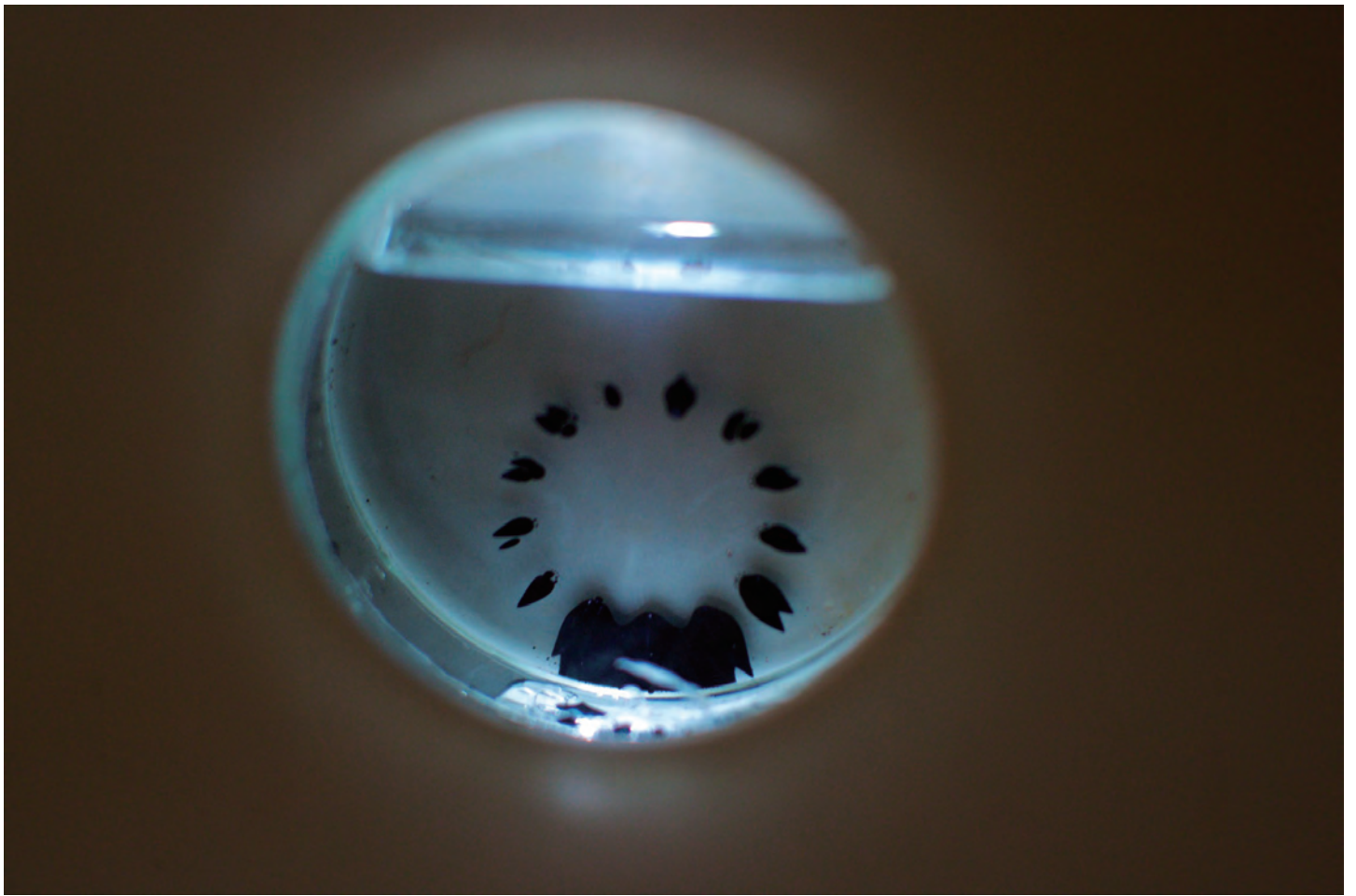
#define STEPS 360
#define BORDER 5

Stepper stepper(STEPS, 6, 9, 5, 10);
int previous = 0;
int old_move=0;
int f=0;
int move=0;
int old_f=0;
void setup() {

  stepper.setSpeed(30);
  Serial.begin(9600);
  pinMode(2, OUTPUT);
  pinMode(3, INPUT);
  pinMode(13, OUTPUT);
}

void loop() {
  int a=0;
  digitalWrite(2, HIGH);
  while (digitalRead(3)!=HIGH){
    a++;
  }
  delay(1);
  digitalWrite(2, LOW);
  f+=(a-f)/2;
  Serial.println(f);

  if((f>BORDER)&&(old_f<=BORDER)){
    move = 1 - move;
    //Serial.println("Entered!");
  }else{
  }
  if(move==1){
    stepper.step(10);
  }
  old_f=f;
}
```



?

河本 匠真

■使用パーツ

- ・ Arduino Uno
- ・ ブレッドボード
- ・ モーター制御 IC (たぶん 7291)
- ・ ジャンパワイヤ
- ・ DC モーター
- ・ ギアボックス
- ・ 測距センサー (たぶん <http://akizukidenshi.com/catalog/g/gl-03158/>)

Comments

磁性流体がまるで生き物みたいでおもしろかったです。

測距センサーがうまく働いていないようでしたが、たぶんマイナスの値とかが出ていたからだと思います。でもそのランダム性が逆に良かったかも。

机の下に測距センサーを仕込んでいて見た目には見えないようにしているところがすごく良かったです。

プログラムは正転反転を繰り返しながら途中でアナログピンを読み込んで goto→ストップさせるという単純なもの。

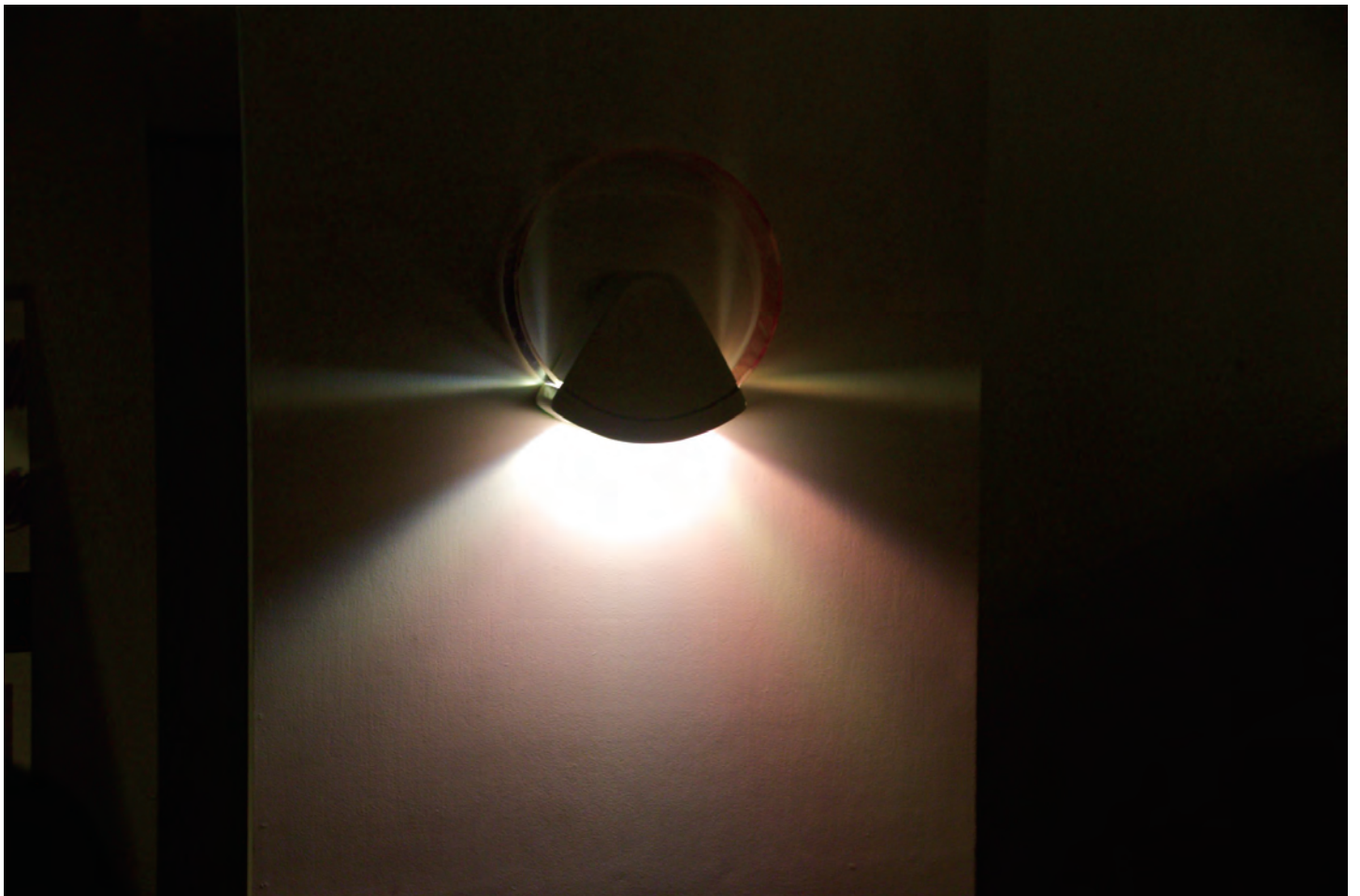
プロトタイプ時はいいですが、本番ではシリアル通信を切ったほうが速く動作すると思います。

```
// モーターへの接続
const int in1Pin = 12;
const int in2Pin = 13;
const int in3Pin = 10;
const int in4Pin = 11;
const int in5Pin = 8;
const int in6Pin = 9;
const int in7Pin = 6;
const int in8Pin = 7;
const int in9Pin = 4;
const int in10Pin = 5;
const int in11Pin = 2;
const int in12Pin = 3;
const int in13Pin = 14;
const int in14Pin = 15;
// 測距センサー
const int sensorPin = 0;
// 過去の距離を格納
int old_range=0;
// 前進
void forwardA(int value) {
  analogWrite(in1Pin, value);
  analogWrite(in2Pin, 0);
}
void forwardB(int value) {
  analogWrite(in3Pin, value);
  analogWrite(in4Pin, 0);
}
void forwardC(int value) {
  analogWrite(in5Pin, value);
  analogWrite(in6Pin, 0);
}
void forwardD(int value) {
  analogWrite(in7Pin, value);
  analogWrite(in8Pin, 0);
}
void forwardE(int value) {
  analogWrite(in9Pin, value);
  analogWrite(in10Pin, 0);
}
void forwardF(int value) {
  analogWrite(in11Pin, value);
  analogWrite(in12Pin, 0);
}
void forwardG(int value) {
  analogWrite(in13Pin, value);
  analogWrite(in14Pin, 0);
}
// 後退
void reverseA(int value){
  analogWrite(in1Pin, 0);
  analogWrite(in2Pin, value);
}
void reverseB(int value){
  analogWrite(in3Pin, 0);
  analogWrite(in4Pin, value);
}
void reverseC(int value){
  analogWrite(in5Pin, 0);
  analogWrite(in6Pin, value);
}
void reverseD(int value){
  analogWrite(in7Pin, 0);
  analogWrite(in8Pin, value);
}
```

```
void reverseE(int value){
  analogWrite(in9Pin, 0);
  analogWrite(in10Pin, value);
}
void reverseF(int value){
  analogWrite(in11Pin, 0);
  analogWrite(in12Pin, value);
}
void reverseG(int value){
  analogWrite(in13Pin, 0);
  analogWrite(in14Pin, value);
}
// 停止
void despinA(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in1Pin, HIGH);
    digitalWrite(in2Pin, HIGH);
  }
  else{
    digitalWrite(in1Pin, LOW);
    digitalWrite(in2Pin, LOW);
  }
}
void despinB(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in3Pin, HIGH);
    digitalWrite(in4Pin, HIGH);
  }
  else{
    digitalWrite(in3Pin, LOW);
    digitalWrite(in4Pin, LOW);
  }
}
void despinC(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in5Pin, HIGH);
    digitalWrite(in6Pin, HIGH);
  }
  else{
    digitalWrite(in5Pin, LOW);
    digitalWrite(in6Pin, LOW);
  }
}
void despinD(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in7Pin, HIGH);
    digitalWrite(in8Pin, HIGH);
  }
  else{
    digitalWrite(in7Pin, LOW);
    digitalWrite(in8Pin, LOW);
  }
}
void despinE(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in9Pin, HIGH);
    digitalWrite(in10Pin, HIGH);
  }
  else{
    digitalWrite(in9Pin, LOW);
    digitalWrite(in10Pin, LOW);
  }
}
```

```
void despinF(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in11Pin, HIGH);
    digitalWrite(in12Pin, HIGH);
  }
  else{
    digitalWrite(in11Pin, LOW);
    digitalWrite(in12Pin, LOW);
  }
}
void despinG(boolean useBreake = true){
  if(useBreake){
    digitalWrite(in13Pin, HIGH);
    digitalWrite(in14Pin, HIGH);
  }
  else{
    digitalWrite(in13Pin, LOW);
    digitalWrite(in14Pin, LOW);
  }
}
//Setup
void setup(){
  // モーター
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(in3Pin, OUTPUT);
  pinMode(in4Pin, OUTPUT);
  pinMode(in5Pin, OUTPUT);
  pinMode(in6Pin, OUTPUT);
  pinMode(in7Pin, OUTPUT);
  pinMode(in8Pin, OUTPUT);
  pinMode(in9Pin, OUTPUT);
  pinMode(in10Pin, OUTPUT);
  pinMode(in11Pin, OUTPUT);
  pinMode(in12Pin, OUTPUT);
  pinMode(in13Pin, OUTPUT);
  pinMode(in14Pin, OUTPUT);
  Serial.begin(9600);
}
void loop(){
  int value = analogRead(sensorPin);
  int range = (6780 / (value -3)) -4;
  //80cm 以上での動き
  if(range>=50){
    delay(100);
  }
  //50cm 以内 30cm 以上での動き
  if((range>30)&&(range<50)){
    forwardA(255);
    Serial.println("Forward");
    if(analogRead(sensorPin)<=20){
      goto bailout;}
    delay(300);
    despinA(true);
    /* 省略しますここにたくさんの正転反転がある */
    delay(300);
  }
  //30cm 以内での動き
  if ((range <= 30)&&(old_range>30)){
    bailout: //goto によって飛んでくる先
    reverseA(255);
    Serial.println("Reverse");
  }
```

```
delay(100);
despinA(false);
Serial.println("Despin (without brake)");
delay(10);
reverseB(255);
Serial.println("Reverse");
delay(100);
despinB(false);
Serial.println("Despin (without brake)");
delay(10);
reverseC(255);
Serial.println("Reverse");
delay(100);
despinC(false);
Serial.println("Despin (without brake)");
delay(10);
reverseD(255);
Serial.println("Reverse");
delay(100);
despinD(false);
Serial.println("Despin (without brake)");
delay(10);
reverseE(255);
Serial.println("Reverse");
delay(100);
despinE(false);
Serial.println("Despin (without brake)");
delay(10);
reverseF(255);
Serial.println("Reverse");
delay(100);
despinF(false);
Serial.println("Despin (without brake)");
delay(10);
reverseG(255);
Serial.println("Reverse");
delay(100);
despinG(false);
Serial.println("Despin (without brake)");
delay(10);
old_range = range;
delay(100);
}
```

Ray

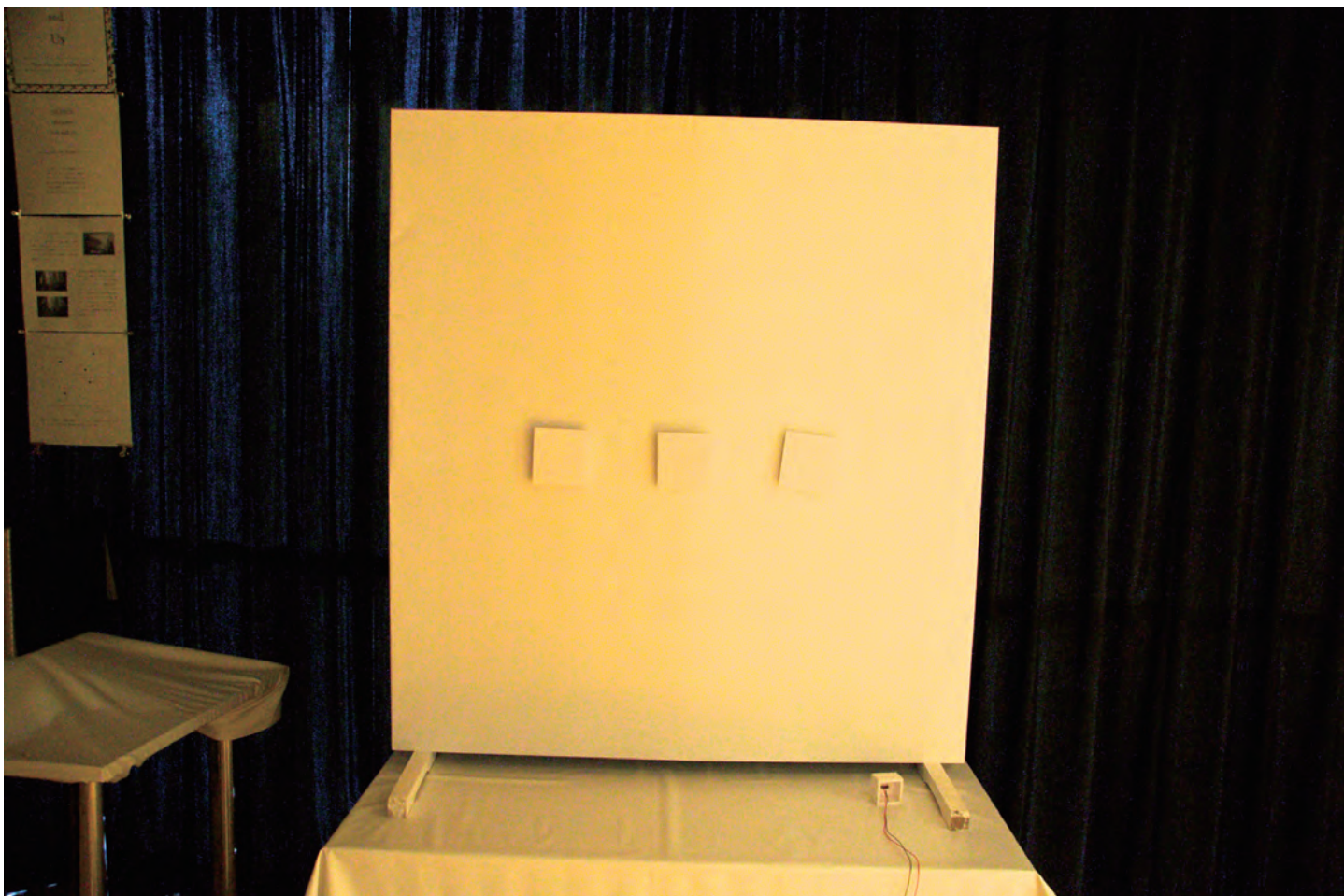
黒尾 智也

■使用パーツ

- Arduino Uno
- 100V 電球
- SSR
- ステッピングモーター
- モーター用 IC MP4401

Comments

デジタルとアナログのバランスがすごくいいと思います。Arduino を使うときのポイントかもしれません。
パワーっと光らせたい場合はトライアック万能調光器とフォトカブラを使うといいでしょう。
アクリルの色を変えて海版と森版が見てみたい！



image

近藤 正之

■使用パーツ

- Arduino Uno 3つ
- DC モーター
- ギアボックス
- モータードライバIC 7291
- スイッチ

Comments

非現実を作ってから現実をすり合わせるのは大変そうでした。

Processing で映像を作っていればもう少し簡単にやりたいことが実現したかもしれません。

映像のクオリティが高かったのもそれだけでもよかったかも？笑

```

#define MOTORin 2
#define MOTORout 3
#define SWITCH 0
#define DLY1 2//delay の調整用
#define DLY2 40
#define DLY3 2
#define DLY4 30
#define END 261000//261000
#define START1 91000//start 調整用 //91000
#define END1 95000//95000
#define START2 97000//97000
#define END2 1090000//99000
#define START3 205000//205000
#define END3 255000//255000

```

```

unsigned long start_time=0;
int start=0;//start してたら 1 を格納
int val=0;
int old_val=0;
int sw_border=800;// スイッチの閾値

```

```

void maxrot(){
digitalWrite(MOTORin,HIGH);
digitalWrite(MOTORout,LOW);
}

```

```

void rot(int i, int j){
digitalWrite(MOTORin,HIGH);
digitalWrite(MOTORout,LOW);
delay(i);
digitalWrite(MOTORin,HIGH);
digitalWrite(MOTORout,HIGH);
delay(j);
}

```

```

void rmaxrot(){
digitalWrite(MOTORin,LOW);
digitalWrite(MOTORout, HIGH);
}

```

```

void rrot(int i, int j){
digitalWrite(MOTORin,LOW);
digitalWrite(MOTORout,HIGH);
delay(i);
digitalWrite(MOTORin,HIGH);
digitalWrite(MOTORout,HIGH);
delay(j);
}

```

```

void stop(){
digitalWrite(MOTORin,HIGH);
digitalWrite(MOTORout,HIGH);
delay(500);
digitalWrite(MOTORin,LOW);
digitalWrite(MOTORout,LOW);
}

```

```

void setup() {
pinMode(MOTORin,OUTPUT);
pinMode(MOTORout,OUTPUT);
pinMode(SWITCH,INPUT);
//Serial.begin(9600);
}

```

```

void loop()
{
val=analogRead(SWITCH);
//Serial.println(val);
if((start==0)&&(val>sw_border)&&(old_val<sw_border)){
start_time=millis();
//Serial.println("start");
start=1;
}
while((start==1)&&(millis())>(start_time+START1))&& (millis())<(start_time+END1)){
//Serial.println("start1");
rrot(DLY1,DLY2);
delay(10);
}
while((start==1)&&(millis())>(start_time+START2))&& (millis())<(start_time+END2)){
//Serial.println("start2");
rot(DLY3,DLY4);
delay(10);
}
while((start==1)&&(millis())>(start_time+START3))&& (millis())<(start_time+END3)){
//Serial.println("start3");
maxrot();
delay(10);
}
if(millis()>start_time+END){
//Serial.println("stop");
stop();
start=0;
}
old_val=val;
}

```




視核

関根 ひかり

■使用パーツ

- Arduino Uno
- ブレッドボード
- ジャンパワイヤ
- DC モーター
- ギアボックス
- 測距センサー (20~150cm) (GP2Y0AO2YK <http://akizukidenshi.com/catalog/g/gi-03158/>)
- AC アダプター 9V 1.3A
- モータードライバ IC TA7291P

Comments

測距センサの範囲に入ったらモーターを正転反転を繰り返すプログラムで、delay の直前に analogRead をいれて人がいなかったら goto でモーターをストップさせています。

展示中うまく行かなかったのは 7291 のピンの回路が間違えていたのと、測距センサの range の値がマイナスになっていたため。

また、モーターが回りだすと板が振動し、それが測距センサーに伝わって変な値も返ってきていました。

会場でシリアル通信してどんな値が返っているか一度みるのが大事でしたね。

個人的にはリセットを押しまくる関根さんがおもしろかったです。

```

#define SENSOR 0
#define motor1_pin1 2
#define motor1_pin2 3
#define motor2_pin1 4
#define motor2_pin2 5
#define motor3_pin1 6
#define motor3_pin2 7
#define motor4_pin1 8
#define motor4_pin2 9

int val =0;
int old_val=1000;
int sensor_v=0;
int range=0;
int border=200;// 反応する距離
int val_border=100;// 止まる値

// 全て停止
void motor_stop(){
  digitalWrite(motor1_pin1,HIGH);
  digitalWrite(motor1_pin2,HIGH);
  digitalWrite(motor2_pin1,HIGH);
  digitalWrite(motor2_pin2,HIGH);
  digitalWrite(motor3_pin1,HIGH);
  digitalWrite(motor3_pin2,HIGH);
  digitalWrite(motor4_pin1,HIGH);
  digitalWrite(motor4_pin2,HIGH);

  delay(500);
  digitalWrite(motor1_pin1,LOW);
  digitalWrite(motor1_pin2,LOW);
  digitalWrite(motor2_pin1,LOW);
  digitalWrite(motor2_pin2,LOW);
  digitalWrite(motor3_pin1,LOW);
  digitalWrite(motor3_pin2,LOW);
  digitalWrite(motor4_pin1,LOW);
  digitalWrite(motor4_pin2,LOW);
}

//motor1 の正回転
void motor1_forward(){
  digitalWrite(motor1_pin1,HIGH);
  digitalWrite(motor1_pin2,LOW);
}

//motor1 の逆回転
void motor1_reverse(){
  digitalWrite(motor1_pin1,LOW);
  digitalWrite(motor1_pin2,HIGH);
}

//motor2 の正回転
void motor2_forward(){
  digitalWrite(motor2_pin1,HIGH);
  digitalWrite(motor2_pin2,LOW);
}

```

```

//motor2 の逆回転
void motor2_reverse(){
  digitalWrite(motor2_pin1,LOW);
  digitalWrite(motor2_pin2,HIGH);
}

//motor3 の正回転
void motor3_forward(){
  digitalWrite(motor3_pin1,HIGH);
  digitalWrite(motor3_pin2,LOW);
}

//motor3 の逆回転
void motor3_reverse(){
  digitalWrite(motor3_pin1,LOW);
  digitalWrite(motor3_pin2,HIGH);
}

//motor4 の正回転
void motor4_forward(){
  digitalWrite(motor4_pin1,HIGH);
  digitalWrite(motor4_pin2,LOW);
}

//motor4 の逆回転
void motor4_reverse(){
  digitalWrite(motor4_pin1,LOW);
  digitalWrite(motor4_pin2,HIGH);
}

void setup(){
  //Serial.begin(9600);
  pinMode(motor1_pin1,OUTPUT);
  pinMode(motor1_pin2,OUTPUT);
  pinMode(motor2_pin1,OUTPUT);
  pinMode(motor2_pin2,OUTPUT);
  pinMode(motor3_pin1,OUTPUT);
  pinMode(motor3_pin2,OUTPUT);
  pinMode(motor4_pin1,OUTPUT);
  pinMode(motor4_pin2,OUTPUT);
  //Serial.begin(9600);
}

void loop(){
  val = analogRead(SENSOR); // センサの値取得
  sensor_v=(val+old_val)/2;
  range = (6000/ sensor_v);
  //Serial.println(val);
  delay(100);

  if(range<=border){
    motor1_forward();
    //Serial.println(analogRead(SENSOR));
    if(analogRead(SENSOR)<val_border){
      goto bailout;}

```

```

    delay(800);
    motor2_forward();
    Serial.println(analogRead(SENSOR));

    if(analogRead(SENSOR)<val_border){
      goto bailout;}

    delay(1500);
    motor3_forward();
    //Serial.println(analogRead(SENSOR));
    // Serial.println("forward");
    if(analogRead(SENSOR)<val_border){
      goto bailout;}

    delay(500);
    motor4_forward();
    //Serial.println(analogRead(SENSOR));
    if(analogRead(SENSOR)<val_border){
      goto bailout;}

    delay(1200);
    motor1_reverse();
    // Serial.println("reverse");
    //Serial.println(analogRead(SENSOR));
    if(analogRead(SENSOR)<val_border){
      goto bailout;}

    delay(800);
    motor2_reverse();
    //Serial.println(analogRead(SENSOR));

    if(analogRead(SENSOR)<val_border){
      goto bailout;}
    delay(1500);

    motor3_reverse();
    // Serial.println("reverse");
    //Serial.println(analogRead(SENSOR));

    if(analogRead(SENSOR)<val_border){
      goto bailout;}
    delay(500);
    motor4_reverse();
    //Serial.println(analogRead(SENSOR));
    if(analogRead(SENSOR)<val_border){
      goto bailout;}
    delay(1200);
  }
  if(range>border){
    bailout:
    motor_stop();
    delay(100);
  }
  old_val=val;
}

```




Time is Money

角川 菜穂子

■使用パーツ

- Arduino Uno
- LED2 色
- ブレッドボード
- スイッチ 10 個
- ギアボックス 3 個
- 圧電スピーカー（振動検知用）
- 7 セグディスプレイ（動かない）
- DC モーター
- モータードライバ IC 7291 を 3 個
- 9V 1.3A AC アダプター

Comments

作品の見た目がメカメカしくてすごく好きです。

ギアボックスは動かしはじめは遅く、ある程度動かすと早く回るという性質があり、delay で duty 比制御をして一定速度で回るように工夫しています。ボックスの中に入れるときにピンが外れたりすることも多かったのでハンダ付けしちゃったほうがよかったかもしれませんね。

USB からのみ電源をとる場合と 9V アダプターを指す場合とで、モーターの回転角が違うので、完成時と同じ条件でプロトタイプ製作をする必要がありました。最終日になって LED が壊れるという事件があったので、予備用に LED を買っておけばよかったですね。

ディスプレイと針を動かすなら、もう一個 Arduino を入れて中で連携させると簡単にできる気がします！

```
#define G_LED 4 // LED が接続されているピン
#define R_LED 3
#define BUTTON 7 // プッシュボタンが接続
#define SOUND 30
#define SPIN 110
#define M1in 12
#define M1out 13
#define M2in 11
#define M2out 10
#define M3in 9
#define M3out 8
```

```
int val = 0; // 入力ピンの状態 val に記憶される
int sstate=0;
int i=0;
int gled=0;
int rled=1;
unsigned long time=0;
unsigned long sttime=0;
int ten=0;
int five=0;
int one=0;
int f=0;
int money=0;
int noize = 0;
```

```
void setup() {
  pinMode(G_LED, OUTPUT); //LED が出力
  pinMode(R_LED, OUTPUT); //LED が出力
  pinMode(BUTTON, INPUT); // BUTTON は入力
  pinMode(M1in,OUTPUT); //1 ピンを信号用ピン
  pinMode(M1out,OUTPUT); //2 ピンを信号用ピン
  pinMode(M2in,OUTPUT); //1 ピンを信号用ピン
  pinMode(M2out,OUTPUT); //2 ピンを信号用ピン
  pinMode(M3in,OUTPUT); //1 ピンを信号用ピン
  pinMode(M3out,OUTPUT); //2 ピンを信号用ピン
  Serial.begin(9600);
}
```

```
void loop(){
  val = digitalRead(BUTTON); // val に格納
  // 入力 HIGH( ボタンが押されている状態 ) が ?
```

```
  if(val == HIGH) {
    digitalWrite(G_LED,HIGH);
    if(sttime==0){
      sttime=millis();// 時間計測開始
    }
  }
  else{
    digitalWrite(G_LED,LOW);
    if(sttime!=0){
      time=millis()-sttime;
      sttime=0;
      time=time/1000;
    }
  }
  noize = analogRead(0);
  Serial.println(noize);
  if( noize > SOUND){
    rled=1-rled;
    digitalWrite(R_LED,rled);
    if(sttime!=0){
      time=millis()-sttime;
```

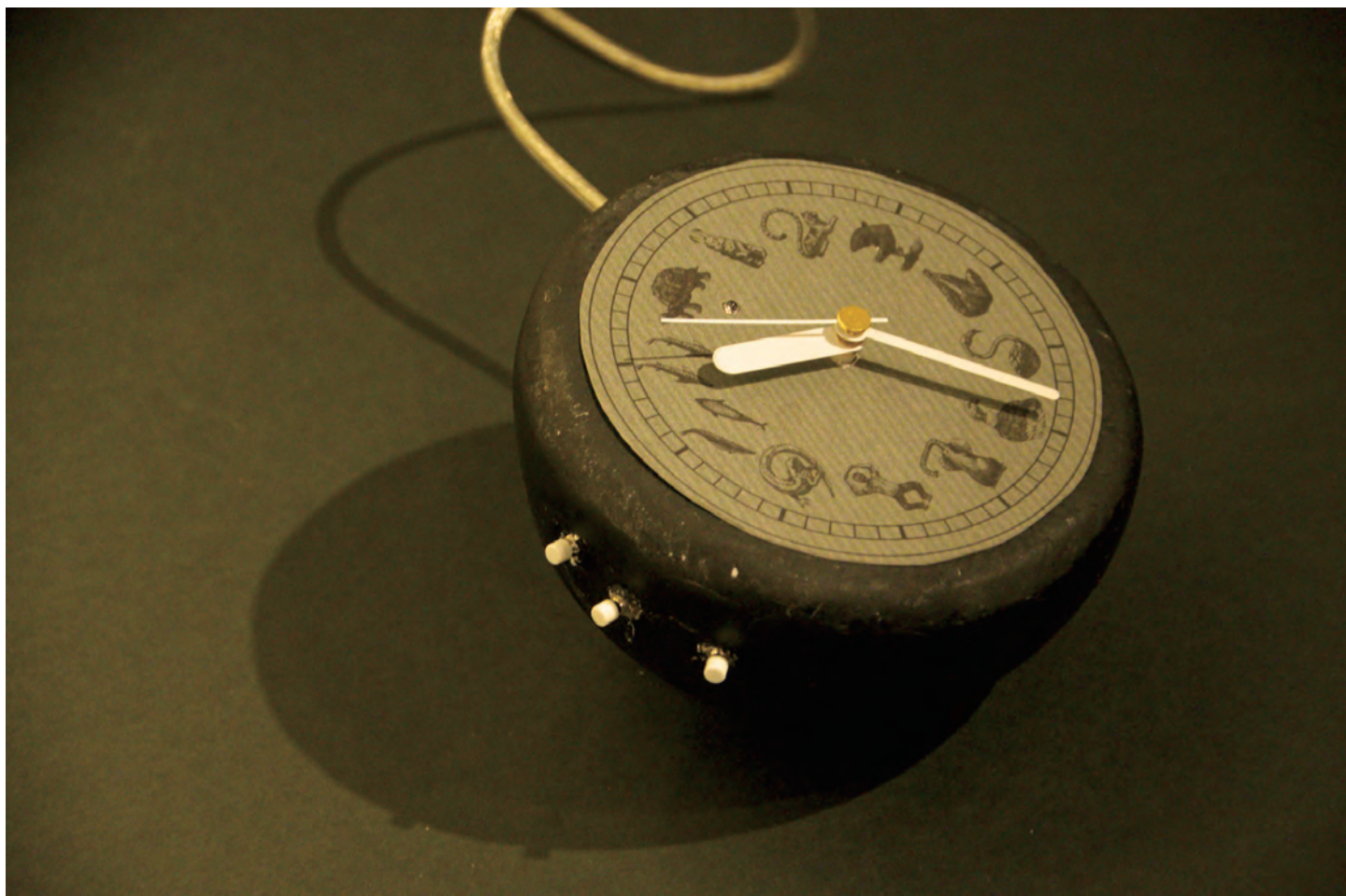
```
      sttime=0;
      time=time/1000;
    }
    delay(10);
  }
  else{
    digitalWrite(R_LED,LOW);
  }

  Serial.println(val);
  if(time!=0){
    Serial.println(time);
    //money=time;
    money=time*0.1051587;// 以下お金計算
    Serial.println(money);
    ten=(money/10);
    five=(money-ten*10)/5;
    one=money-five*5-ten*10;
    // モーターを正転させる
    for(f=0;f<ten;f++){ //10 円のぶん
      for(i=0;i<SPIN+35;i++){
        digitalWrite(M1in,HIGH);
        digitalWrite(M1out,LOW);
        //2 秒待つ ( まわる )
        gled=1-gled;
        rled=1-rled;
        digitalWrite(G_LED,gled);
        digitalWrite(R_LED,rled);
        delay(1);
        digitalWrite(M1in,HIGH);
        digitalWrite(M1out,HIGH);
        delay(2);
      }
    }
```

```
    for(i=0;i<3;i++){
      // モーターを stop
      digitalWrite(M1in,HIGH);
      digitalWrite(M1out,HIGH);
      gled=1-gled;
      rled=1-rled;
      digitalWrite(G_LED,gled);
      digitalWrite(R_LED,rled);
      delay(100);
    }
  }
```

```
    for(i=0;i<5;i++){
      // モーターを stop
      digitalWrite(M1in,LOW);
      digitalWrite(M1out,LOW);
      gled=1-gled;
      rled=1-rled;
      digitalWrite(G_LED,gled);
      digitalWrite(R_LED,rled);
      delay(100);
    }
  }
  for(f=0;f<five;f++){ //5 円のぶん
    for(i=0;i<SPIN+25;i++){
      digitalWrite(M2in,HIGH);
      digitalWrite(M2out,LOW);
      //2 秒待つ ( まわる )
      gled=1-gled;
      rled=1-rled;
      digitalWrite(G_LED,gled);
      digitalWrite(R_LED,rled);
```

```
      digitalWrite(M2in,HIGH);
      digitalWrite(M2out,HIGH);
      delay(2);
    }
    for(i=0;i<3;i++){
      // モーターを stop
      digitalWrite(M2in,HIGH);
      digitalWrite(M2out,HIGH);
      gled=1-gled;
      rled=1-rled;
      digitalWrite(G_LED,gled);
      digitalWrite(R_LED,rled);
      delay(100);
    }
    for(i=0;i<5;i++){
      // モーターを stop
      digitalWrite(M2in,LOW);
      digitalWrite(M2out,LOW);
      gled=1-gled;
      rled=1-rled;
      digitalWrite(G_LED,gled);
      digitalWrite(R_LED,rled);
      delay(100);
    }
    for(f=0;f<one;f++){ //1 円のぶん
      for(i=0;i<SPIN+15;i++){
        digitalWrite(M3in,HIGH);
        digitalWrite(M3out,LOW);
        //2 秒待つ ( まわる )
        gled=1-gled;
        rled=1-rled;
        digitalWrite(G_LED,gled);
        digitalWrite(R_LED,rled);
        delay(1);
        digitalWrite(M3in,HIGH);
        digitalWrite(M3out,HIGH);
        delay(2);
      }
      for(i=0;i<3;i++){
        // モーターを stop
        digitalWrite(M3in,HIGH);
        digitalWrite(M3out,HIGH);
        gled=1-gled;
        rled=1-rled;
        digitalWrite(G_LED,gled);
        digitalWrite(R_LED,rled);
        delay(100);
      }
      for(i=0;i<5;i++){
        // モーターを stop
        digitalWrite(M3in,LOW);
        digitalWrite(M3out,LOW);
        gled=1-gled;
        rled=1-rled;
        digitalWrite(G_LED,gled);
        digitalWrite(R_LED,rled);
        delay(100);
      }
    }
    time=0;
    money=0;
    ten=0;
    five=0;
    one=0;
  }
}
```



Anima - Clock

中 祥子

■使用パーツ

- Arduino Uno
- LED
- ステッピングモーター
- モーター用 IC MP4401
- スイッチ 3 個
- 時計

Comments

どうやって時計を早くしたり遅くしたりするかが重要なポイントでした。

結局時計を分解し、中のギアにモーターを直付していました。

LED は眩しくて目がくらむほどだったので制限抵抗で 100〜500Ω くらいを挟むといいと思います。

スイッチを押し直したときにどのくらい針が戻るかを計算するのが大変そうでした、高橋くんが。



Your Own Time

平島 眞 池田 大夢 福田 昴正

■使用パーツ

- ・アクリル管
- ・ビニル管
- ・L字ガラス管
- ・試験管用ゴム栓
- ・ウィダーインゼリー袋
- ・水彩絵具
- ・油彩筆洗い液
- ・アルミ板
- ・長ねじ、ナット
- ・ソレノイド（新電元 S75 プッシュ型 <http://goo.gl/I1Syb>）
- ・Arduino Uno
- ・ブレッドボード
- ・心拍センサ A.P.Shield (<http://goo.gl/akEHV>)
- ・木材いろいろ
- ・綿布
- ・スポットライト



Comments

心拍はスイッチサイエンスで売っているシールドでセンシングしている。

なんといってもボコッと液体がでてくる機構がポイント右上参照。ウィダー万能だなぁ。

ランダムにボコボコなっているような演出は MODE を何個も作っておいてループさせていたからなんですね。メンバーの協力体制が良かった作品。

```

#define time 250
#define MODE 18 //-1wasurezu MODE+1 で一周
#define jisa 65
int sensorPin=4;
int ledPin=13;
int sensorValue=0;
int integral_plus=0;
int integral_minus=0;
int count=0;
int elapse_up;
int elapse_down=0;
int ave=0;
int k=0;
int a[5]={2,4,7,8,12};
int mode=0;
int h=0;

```

```

// 押す関数 y どこに入力か
void push(int x,int y){
    if(x==1){
        digitalWrite(a[y],HIGH);
    }
    else{
        digitalWrite(a[y],LOW);
    }
}

// 状態を与える
void seigyo2(int u,int v,int w,int x,int y){
    push(u,0);
    push(v,1);
    push(w,2);
    push(x,3);
    push(y,4);
}

// 状態を与えた後元に戻す
void seigyo(int u,int v,int w,int x,int y){
    push(u,0);
    push(v,1);
    push(w,2);
    push(x,3);
    push(y,4);
    delay(time);
    for(int i=0;i<5;i++){
        digitalWrite(a[i],LOW);
    }
}

```

```

void setup(){
    pinMode(ledPin,OUTPUT);
    for(int i=0;i<5;i++){
        pinMode(a[i],OUTPUT);
    }
    Serial.begin(9600);
}

```

```

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

    if(diff<10&&diff>-10&&integral_plus>
160&&integral_minus<-180){
    if(elapse_up>39&&elapse_up<460
&&elapse_down>76&&elapse_down<860){
    if(mode>=0&&mode<=8){

```

```

if(mode==0){
    seigyo(1,0,0,0,0);
}
else if(mode==1){
    seigyo(0,1,0,0,0);
}
else if(mode==2){
    seigyo(0,0,1,0,0);
}
else if(mode==3){
    seigyo(0,0,0,1,0);
}
else if(mode==4){
    seigyo(0,0,0,0,1);
}
else if(mode==5){
    seigyo(0,0,0,1,0);
}
else if(mode==6){
    seigyo(0,0,1,0,0);
}
else if(mode==7){
    seigyo(0,1,0,0,0);
}
else{
    seigyo(1,0,0,0,0);
}

}


```

```

else if(mode>=9&&mode<=17){
    if(mode==9){
        seigyo(0,0,0,0,1);
    }
    else if(mode==10){
        seigyo(0,0,0,1,0);
    }
    else if(mode==11){
        seigyo(0,0,1,0,0);
    }
    else if(mode==12){
        seigyo(0,1,0,0,0);
    }
    else if(mode==13){
        seigyo(1,0,0,0,0);
    }
    else if(mode==14){
        seigyo(0,1,0,0,0);
    }
    else if(mode==15){
        seigyo(0,0,1,0,0);
    }
    else if(mode==16){
        seigyo(0,0,0,1,0);
    }
    else{
        seigyo(0,0,0,0,1);
    }
}

else if(mode==18){
    if(k==0){
        seigyo(1,1,1,1,1);
        k=1;
    }
}

```

```

else if(k==1){
    seigyo2(0,0,1,0,0);
    delay(jisa);
    seigyo2(0,1,1,1,0);
    delay(jisa);
    seigyo2(1,1,1,1,1);
    delay(time-2*jisa);
    seigyo2(0,0,0,0,0);
    k=2;
}
else if(k==2){
    seigyo(0,0,1,0,0);
    k=3;
    mode=mode-1;
}
else if(k==3){
    seigyo(0,1,0,1,0);
    k=4;
    mode=mode-1;
}
else{
    seigyo(1,0,0,0,1);
    k=0;
}

}


```

```

if(mode<MODE){
    mode=mode++;
}
else{
    mode=0;
}

}


```

```

    elapse_up=0;
    elapse_down=0;
    integral_plus=0;
    integral_minus=0;

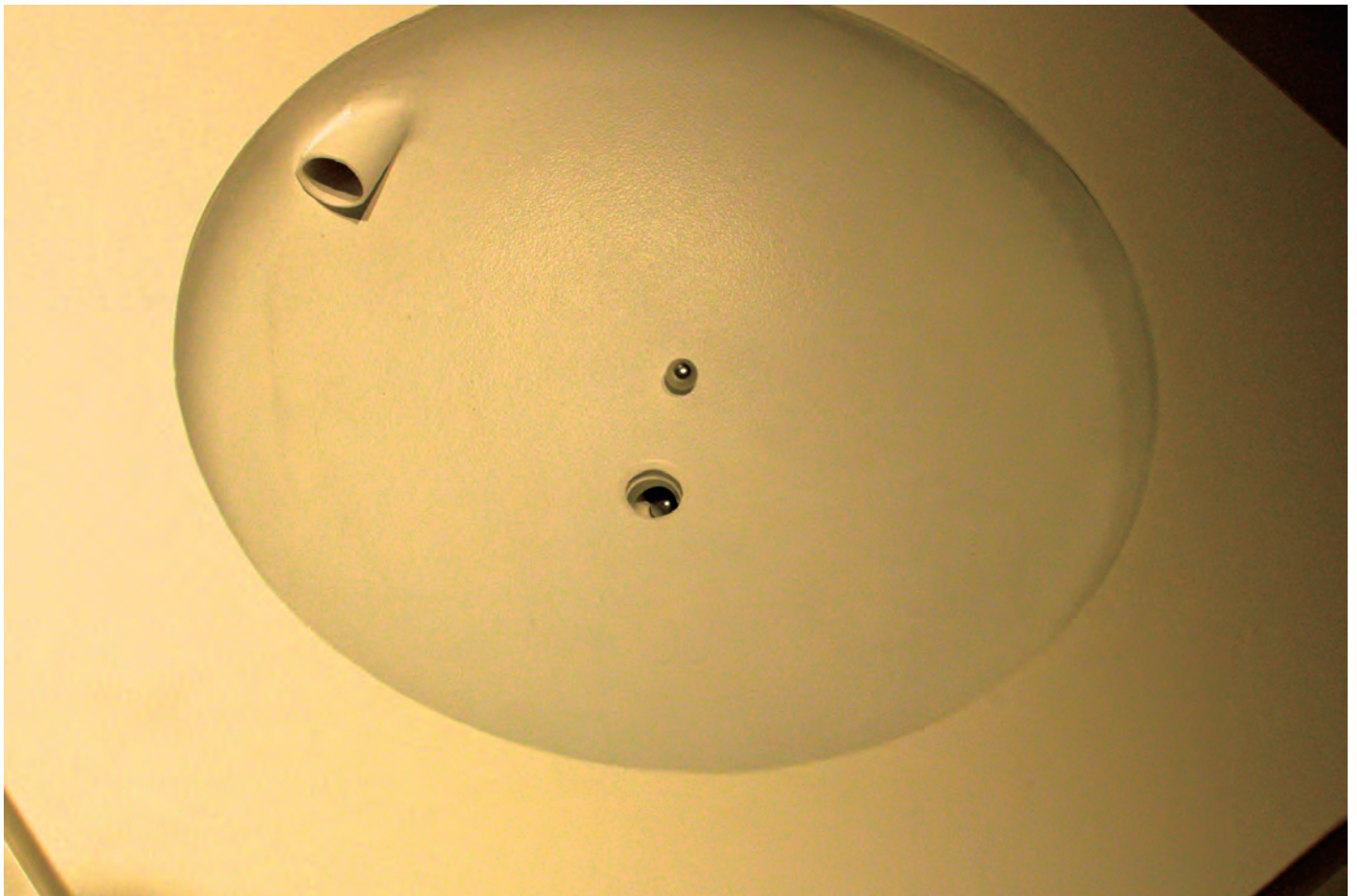
}

    else if(diff>10&&diff<200){
        integral_plus+=diff;
        elapse_up+=20;
    }

    else if(diff<-10&&diff>-200){
        integral_minus+=diff;
        elapse_down+=20;
    }

    delay(20);
}


```

whirr

丸形 彩実

■使用パーツ

- ・ Arduino Uno
- ・ プッシュ型ソレノイド
- ・ 鉄球とアルミ球
- ・ ネオジウム磁石

Comments

玉を一定間隔で出す機構をソレノイドで実現していました。
一度アドバイス会に来ただけで作り上げてしまったのがすごい。
鉄球とアルミ球の2種類を用意したのがうまいと思いました。



SILENCE

三宅 瑠人

■使用パーツ

- ・ Arduino Uno
- ・ DC モーター
- ・ ギア
- ・ モータードライバ IC 7291
- ・ 圧電スピーカー（振動検出用）
- ・ LED ? 電球 ?

Comments

Facebook をフル活用して質問回答クラウドを達成。

プログラムは関根さんや河本さんとほぼ同じ。

石井くんが問題を分割したのを「怖い」と言っていたのがとても面白かったです。



ときのかん

森 志帆

■使用パーツ

- Arduino Uno
- ブレッドボード
- ジャンパワイヤ
- DC ソレノイド（プル型）(MD25202)
- トランジスタアレイ（東芝 TD62083APG）
- AC アダプタ 12V

Comments

プル型の静かなソレノイドをうまく使って静かな空間を作り出していました。
人が入ってきたら鳴り出すとかにしてもよかったかも？
精神と時の部屋。

```

#define SOLENOIDPIN1 9
#define SOLENOIDPIN2 10
#define SOLENOIDPIN3 11

long randNumber1 = 0;
long randNumber2 = 0;
long randNumber3 = 0;
long randNumber4 = 0;
int count1 = 0;
int count2 = 0;
int count3 = 0;
int pin1 = 0;
int pin2 = 0;
int pin3 = 0;

void setup(){
  Serial.begin(9600);
  pinMode(SOLENOIDPIN1,OUTPUT);
  pinMode(SOLENOIDPIN2,OUTPUT);
  pinMode(SOLENOIDPIN3,OUTPUT);
  randomSeed(analogRead(0));
}

void loop(){
  if(count1==randNumber1){
    count1=0;
    randNumber1 = random(2,10);
    pin1 = 1- pin1;
    Serial.println("Enter! 1");
    Serial.println(randNumber1);
  }
  if(count2==randNumber2){
    count2=0;
    randNumber2 = random(2,10);
    pin2 = 1- pin2;
    Serial.println("Enter! 2");
    Serial.println(randNumber2);
  }
  if(count3==randNumber3){
    count3=0;
    randNumber3 = random(2,10);
    pin3 = 1- pin3;
    Serial.println("Enter! 3");
    Serial.println(randNumber3);
  }
}

```

```

digitalWrite(SOLENOIDPIN1,pin1);
digitalWrite(SOLENOIDPIN2,pin2);
digitalWrite(SOLENOIDPIN3,pin3);
if(pin1==0&&pin2==0&&pin3==0){
  randNumber4 = random(9,11);
  Serial.println("Enter! RND");
  digitalWrite(randNumber4,HIGH);
}

delay(450);

digitalWrite(SOLENOIDPIN1,LOW);
digitalWrite(SOLENOIDPIN2,LOW);
digitalWrite(SOLENOIDPIN3,LOW);
delay(450);

count1++;
count2++;
count3++;

}

```




Beating

山本 直

■使用パーツ

- Arduino Uno
- ブレッドボード
- 傾きセンサ（水銀スイッチ？）
- DC モーター

Comments

水銀スイッチである角度にするとブルッと震えるというすごく単純な機構で、すぐにプロトタイプが完成しました。
どの方向にもしたい場合は、加速度センサなどを入れるといいでしょう。
あと重心の位置に DC モーターを持ってくれば、岩自体が震えている感じがよりでたんじゃないかと思います。

```
// 2 秒間のうち何秒間ぶるぶるさせるかを定める
// KODO が 100 だと 0.1 秒間ふるえる
#define KODO 100
// ふるえる大きさを定める.
// 振動が大きすぎたり小さすぎる場合はここを変える
#define VALUE 100

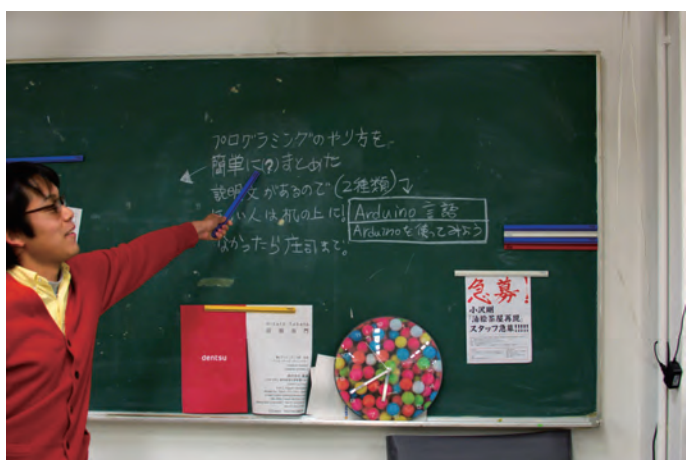
#define TWOSECONDS 2000

int val = 0; // 傾き検出用変数

void setup(){
  pinMode(1,OUTPUT); // 信号用ピン
  pinMode(2,OUTPUT); // 信号用ピン
  pinMode(4,INPUT); // スイッチ用ピン
}

void loop(){

  val = digitalRead(4);
  if(val == HIGH){
    digitalWrite(1,HIGH);
    digitalWrite(2,LOW);
    analogWrite(3,VALUE);
    delay(KODO);
    digitalWrite(1,LOW);
    digitalWrite(2,LOW);
    delay(TWOSECONDS - KODO);
  }
  else{
    digitalWrite(1,LOW);
    digitalWrite(2,LOW);
  }
}
```

石井ちゃんの写真はありませんでした ... すまん ...

