コレクティブデザインB
（Learn Processing）

第8回　様々な応用編

コンテンツ

様々な応用編

1. センサ連動
2. 2Dオブジェクト描画
3. 3Dオブジェクト描画
4. 作品を感動的にするために

センサと連動

本来，音は物理的操作で発生するもの

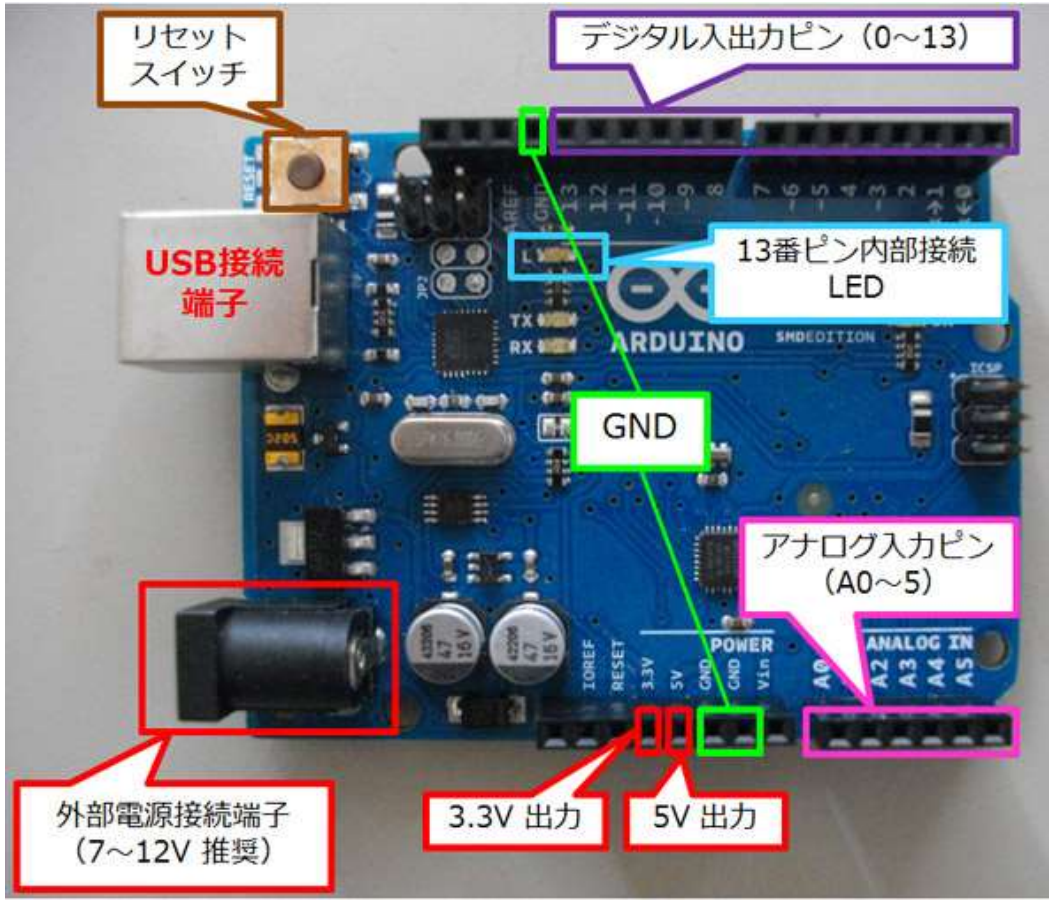
動作や挙動と連動させて鳴らしたり，動かしたりすることでリアルさが増す

この分野はPhysical Computingとして研究・開発が盛んに行われている

今回は超簡単に実現する方法についてのみ紹介する

センサと連動

物理現象を捉える機器としてArduinoを使う



ARDUINO 1.8.3

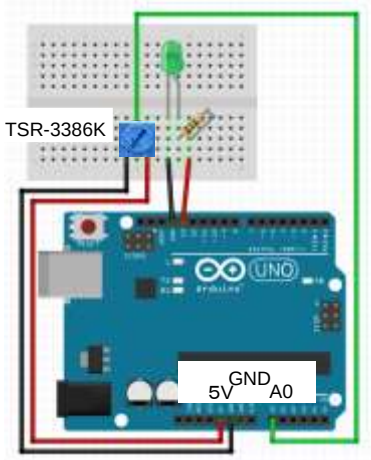
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board.
Refer to the Getting Started page for Installation instructions.

https://www.arduino.cc/en/Main/Software

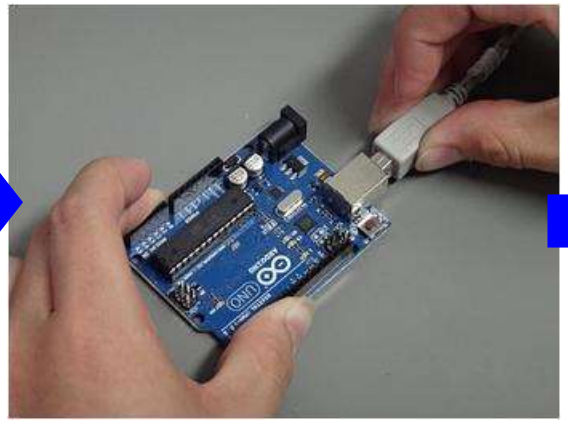
- 入手が容易
- 汎用性が高い
- Processingとの相性
- 安い（1000～3000円）

センサと連動

センサ設置



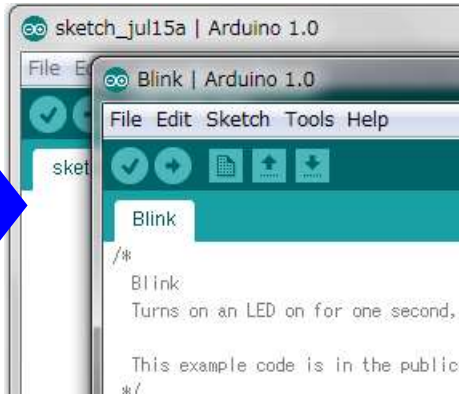
USBを接続



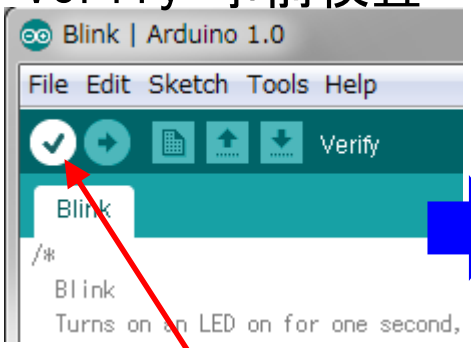
アプリ起動

arduino

プログラム記述

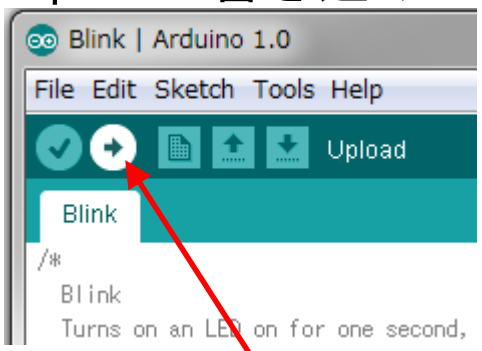


Verify=事前検査



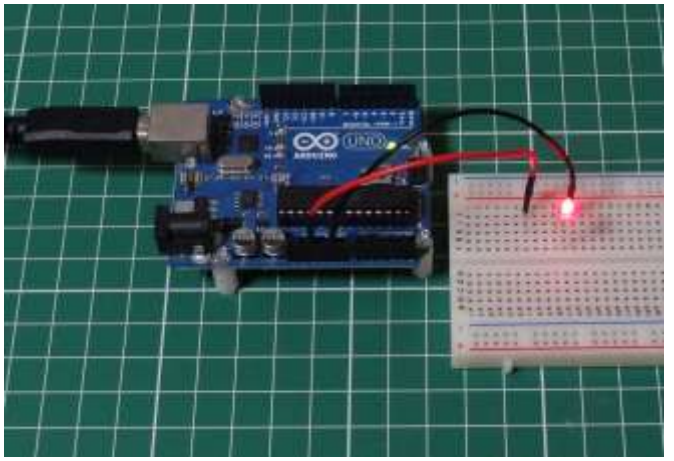
“クリック “

Upload=書き込み



“クリック “

動作完了



センサと連動

プログラム説明

Blink§

```
/*
  Blink
  Turns on an LED on for one second, then off for one

  This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boar
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);   // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // set the LED off
  delay(1000);              // wait for a second
}
```

ほぼProcessingと同じ
（作者一緒なので当然）

最初の1回通過する関数

13pinを出力に設定する

動作中ループする関数

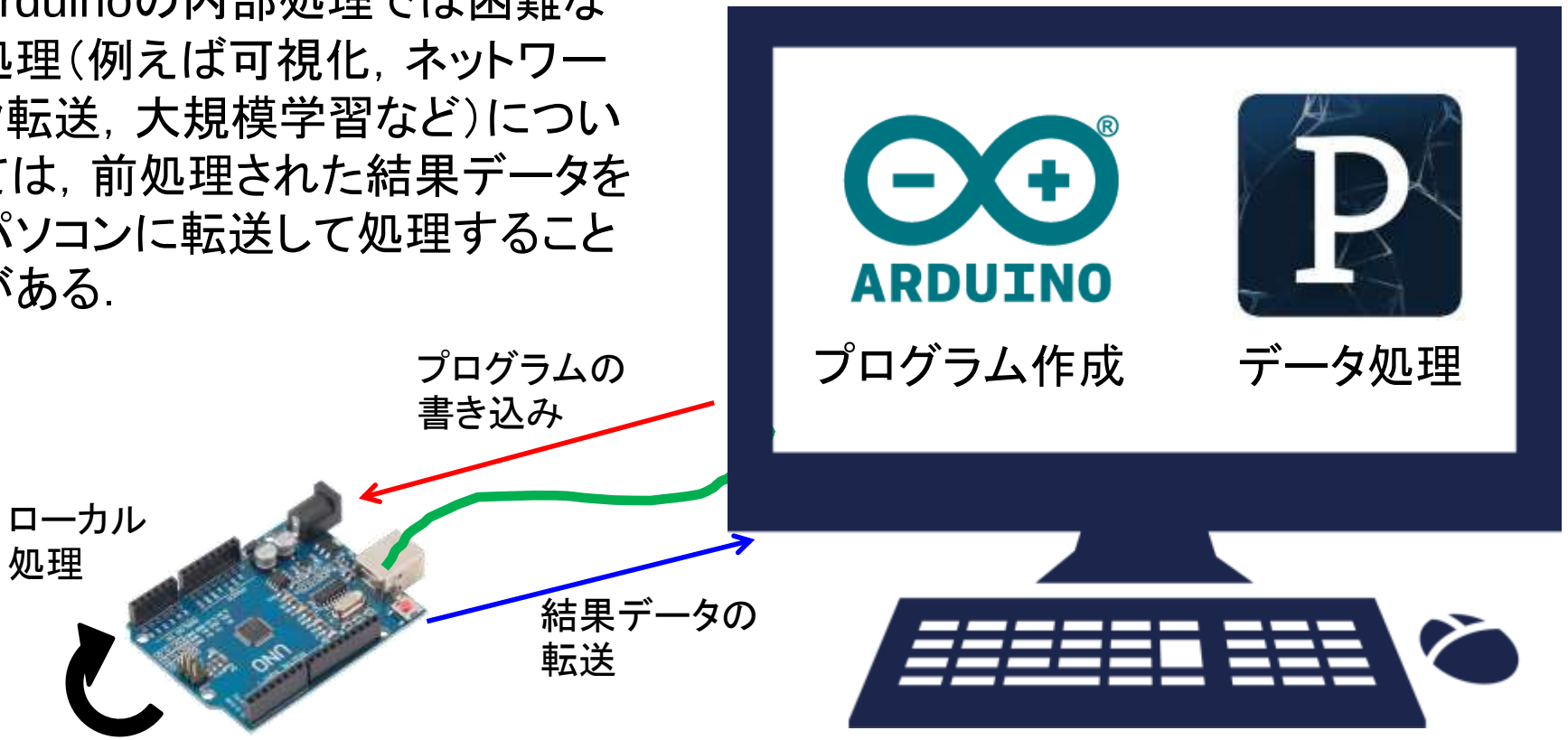
13pinをHigh（5V）にして待つ

13pinをLow（0V）にして待つ

センサと連動

Processingとのシリアル通信

Arduinoの内部処理では困難な処理（例えば可視化，ネットワーク転送，大規模学習など）については，前処理された結果データをパソコンに転送して処理することがある．



接続はシリアル通信なので，CやJavaでも可能であるが，Processingが楽．

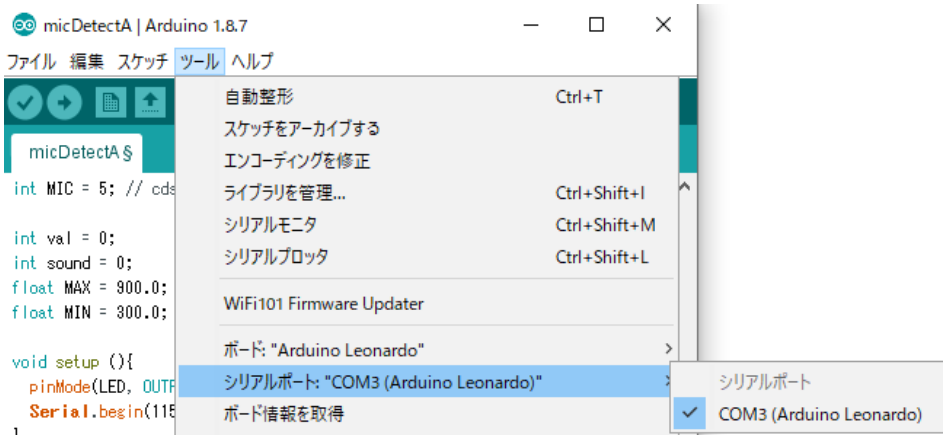
センサと連動

s8_001a
＜デモのみ＞

Arduino側　　公開しているプログラムをArduino IDEへ書き込む

cds2ProcBYTE

```
int MIC = 5; // sensor connected

int val = 0;
int sound = 0;
float MAX = 900.0;
float MIN = 600.0;

void setup (){
  Serial.begin(38400);
}

void loop(){
  val = analogRead(MIC);
  sound = int( (MAX-float(val))/(MAX-MIN)*255.0 );
  if(sound > 255) {
     sound = 255;
  } else if (sound < 0) {
     sound = 0;
  }
   if( Serial.available() > 0 ){
    Serial.write(sound);
    Serial.read();
   }
}
```



このとき，必ず接続しているシリアルポートを確認する

センサと連動

s8_001p
＜デモのみ＞

Processing側

```
import processing.serial.*;

Serial port;
D2Graph tG;    //2Dgraph Class

int width = 600, height = 400;
boolean mousePressBtn = false;
int sdata = 0;

void setup() {
  size(width, height, P3D);
  frameRate(60);
  background(255);

  tG = new D2Graph(int(width*0.75),
      int(height*0.75), "Time[S]", "Output[V]");

  println("Available serial ports:");
  println(Serial.list());

  // Please input your COM port
  port = new Serial(this, "COM3", 38400);
}
```

早くする

先ほどのポート

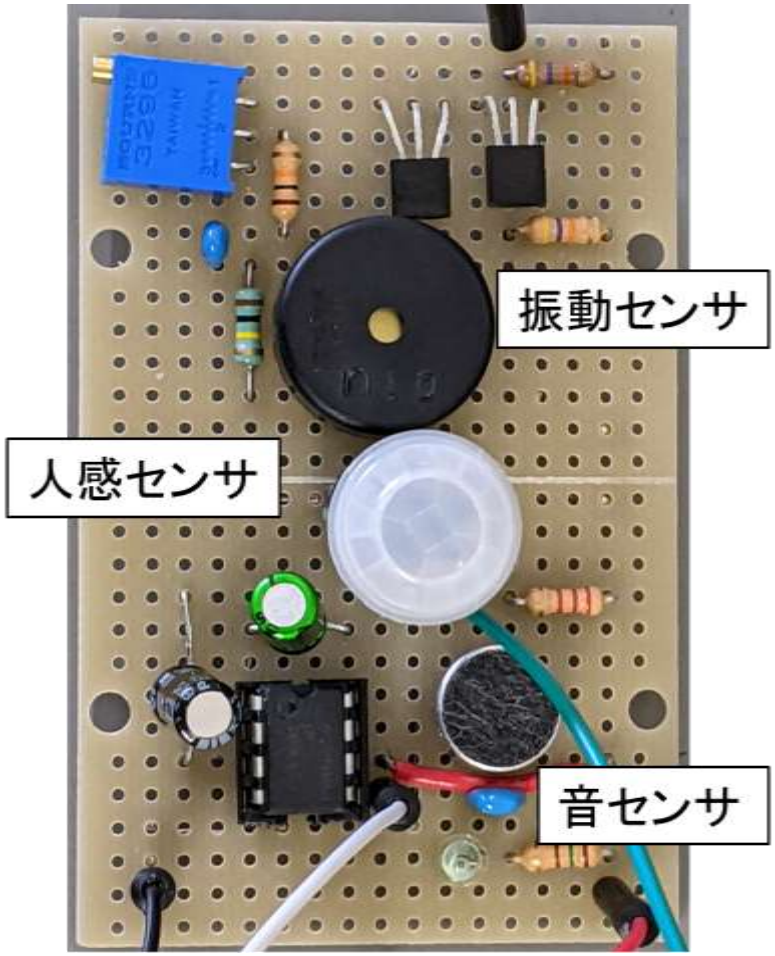
```
void draw(){
  fill(0);
  textSize(16);
  text("Please click !",  200, 30);

  if( mousePressBtn ){
    fill(255, 0, 0);
    text("Sampling!", 330, 30);
  } else {
    fill(255, 255, 255);
    noStroke();
    rect(280, 20, 100, 30);
  }

  tG.graphPointDraw((float)sdata);
}

void serialEvent(Serial p){
  if( port.available( )> 0){
    sdata = port.read();
    println(sdata);

    if( mousePressBtn ){
      port.write(65);  //sign for Arduino
    }
  }
}
```

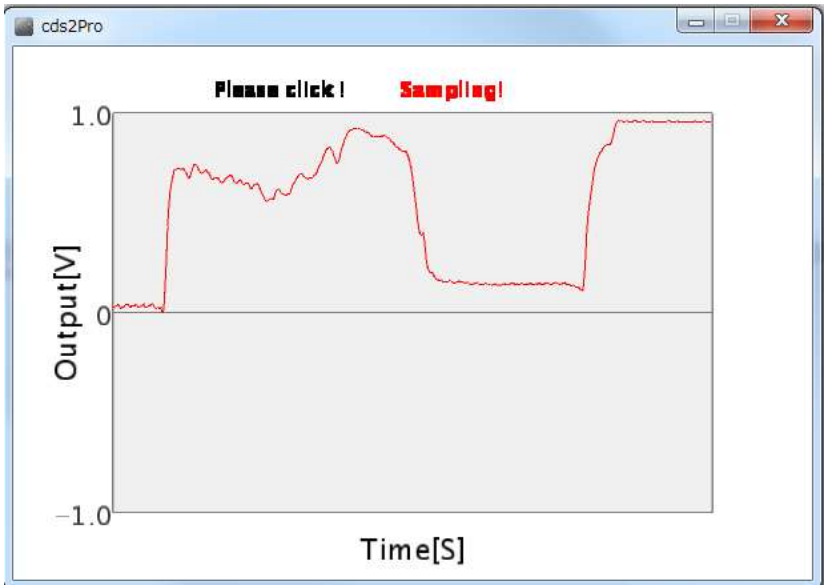
```
void mousePressed(){
  if( !mousePressBtn ){
    port.write(65);
    mousePressBtn = true;
  } else {
    mousePressBtn = false;
  }
}
```

D2GraphR.class
も追加されている

センサと連動

＜デモのみ＞





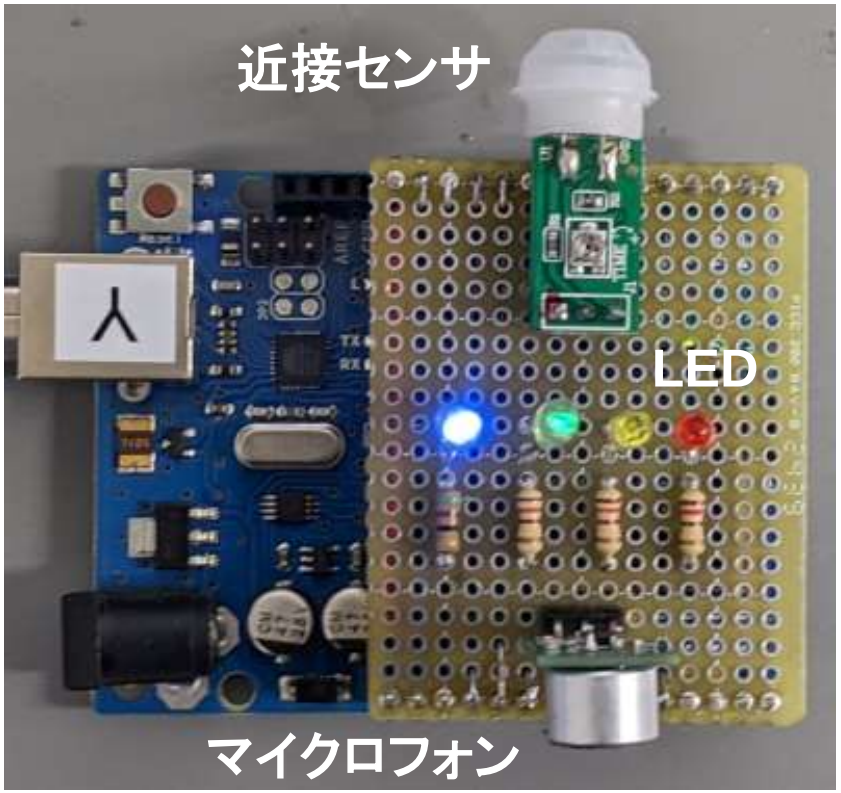
ArduinoのA/D出力は10bitだが、今回は8bitのみを使用する。全情報を送りたいときには上位下位で分けて転送するか，文字列として送受信する.

http://kousaku-kousaku.blogspot.jp/2008/05/arduino-processing_28.html
http://kousaku-kousaku.blogspot.jp/2008/06/arduino-processing.html

センサと連動

＜デモのみ＞

Arduino＝リアルタイムな動作　Processing＝多彩な表現



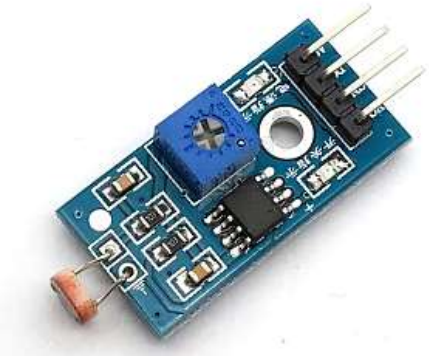
人が近づいてきたら、Arduinoで接近を捉え、Processingにトリガを送り、PCで映像と音で接客。更に流れる音楽の音量をArduinoで捉えてLEDを光らせる。

センサ紹介

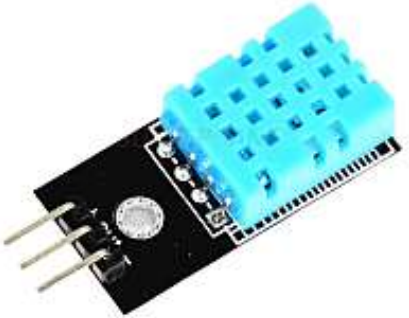
＜皆さんは買い物で効率的に開発を＞

振動センサ

音センサ



照度センサ



温湿度センサ

人感センサ

圧力センサ

距離センサ

加速度センサ

2Dオブジェクト描画

Photoshopやillustrator等で作成した作品を披露したい

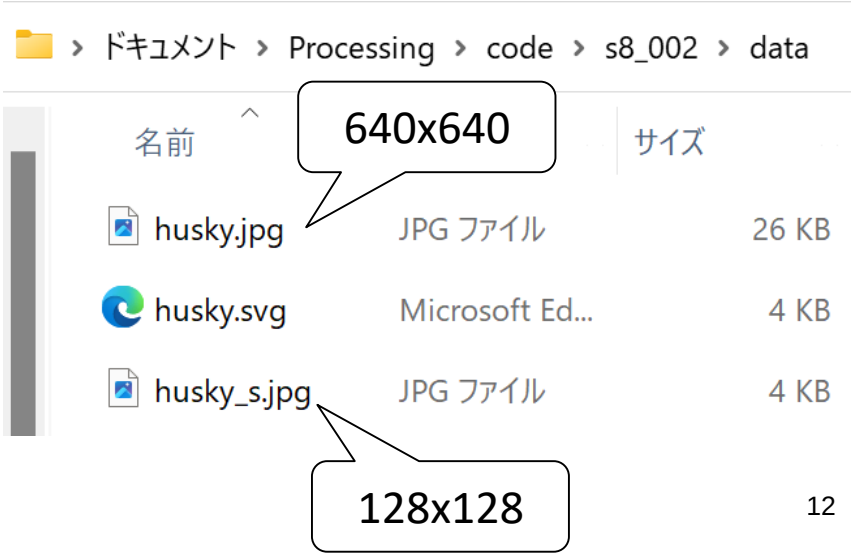
⇩

ベクターデータ(.svg)で出力して，Processingで再生

ベクターデータ

- スケール拡大しても解像度を保つことができる
- ファイルサイズが非常に軽量でアニメーション表現にも対応可


```
<g
   inkscape:label="Layer 1"
   inkscape:groupmode="layer"
   id="layer1"
   transform="translate(0,-38)">
  <path
     inkscape:connector-curvature="0"
     style="fill:#f79018;stroke:#f79018;s
     class="st0"
     d="m 238.92671,161.47369
18.12996,-3.7647 -2.3777,-15.45507 -1
15.75227,-12.97832 -9.21358,-12.68105
11.9876,-15.355987 -12.68106,-9.90711
3.76472,-16.445787 -14.3653,-6.538681
9.516968 h -17.1393 c 0,0
```



2Dオブジェクト描画

s8_002.txt

```
PShape svg;

void setup(){
  size(640, 640, P2D);
  svg = loadShape("husky.svg");
}

void draw(){
  background(255);
  push();
  scale(map(mouseY, 0, height, 0, 1));
  shape(svg);

  push();
  noFill();
  stroke(0);
  strokeWeight(1);
  rect(0, 0, svg.width, svg.height);
  pop();

  pop();
}
```

```
void push(){
  pushMatrix();
  pushStyle();
}

void pop(){
  popStyle();
  popMatrix();
}
```

拡大してもきれいな画像で表示することができる

3Dオブジェクト描画

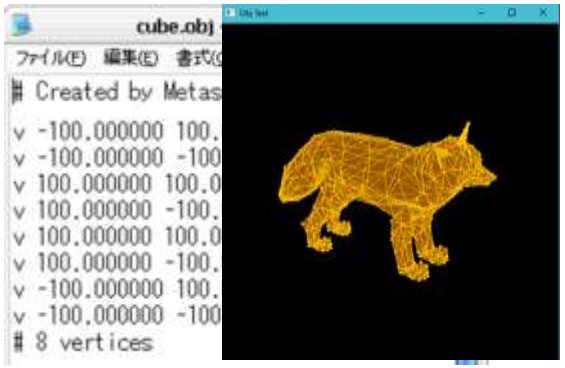
3D-CAD等で作成した自分の作品を披露したい

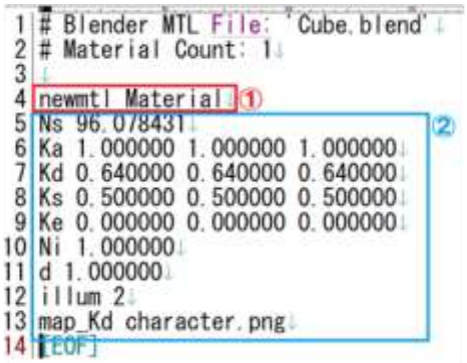
⇩

Wavefront(.obj)形式で出力して，Processingで再生

Wavefront形式

.obj：3Dオブジェクトを構成するメッシュの形状及び法線情報が格納
.mtl：objファイルで使用するマテリアル（材質）ファイルを宣言している
.jpg or png：テクスチャーマッピングする際の画像情報





https://yttm-work.jp/model_render/model_render_0001.html

3Dオブジェクト描画

描画プログラムでの留意点

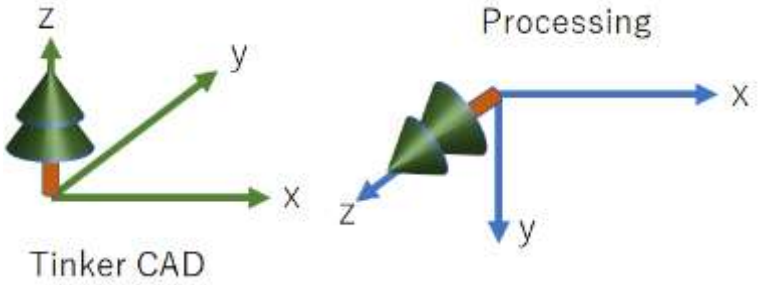
①必要な書類を必要な個所に保存するべき

AutoCADやBlenderはWavefront(OBJ)形式がExportできる
ネット上から得られたファイルの.obj形式への変換は変換できるサイトを探す
.objから.mtlファイルを生成することは可能（反射具合は自ら設定が必要）
https://qiita.com/umi_kappa/items/7f6a613a2b028afff45e

.obj, .mtl, .jpg(.png)の一式を直下のdataの中に名前の整合性を取って格納

②3D-CADの座標系との違いを修正すべき

ソフトウェアによって物体に対する軸方向が
異なる場合があるため，3Dデータを出力する
前にCADソフト上で修正しておくことが必要



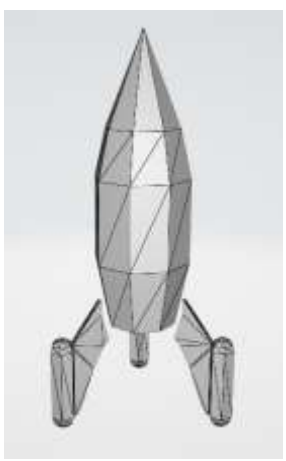
③あまり多データ（多ポリゴン数）の再現は行わない

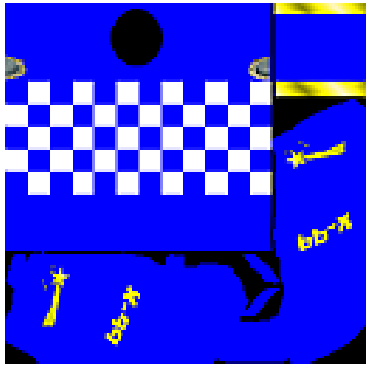
注意：全ての3Dデータを再現できるわけではない（特にダウンロードファイル）

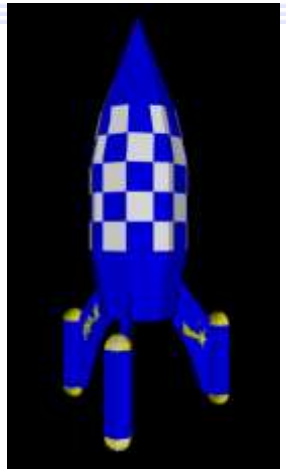
3Dオブジェクト描画

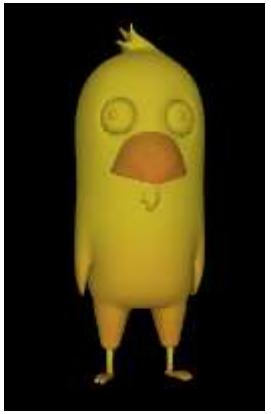
s8_003.txt

```
PShape model;

Void setup(){
  size(640, 640, P3D);
  model = loadShape("rocket.obj");
  model.scale(1.0);
  translate(0, 0, 0);
}

void draw(){
  background(0);
  lights();

  translate(width/2, height/1.5, 0);
  rotateZ(PI);
  rotateY(ry);
  shape(model, 0, 0);

  ry += 0.02;
}
```









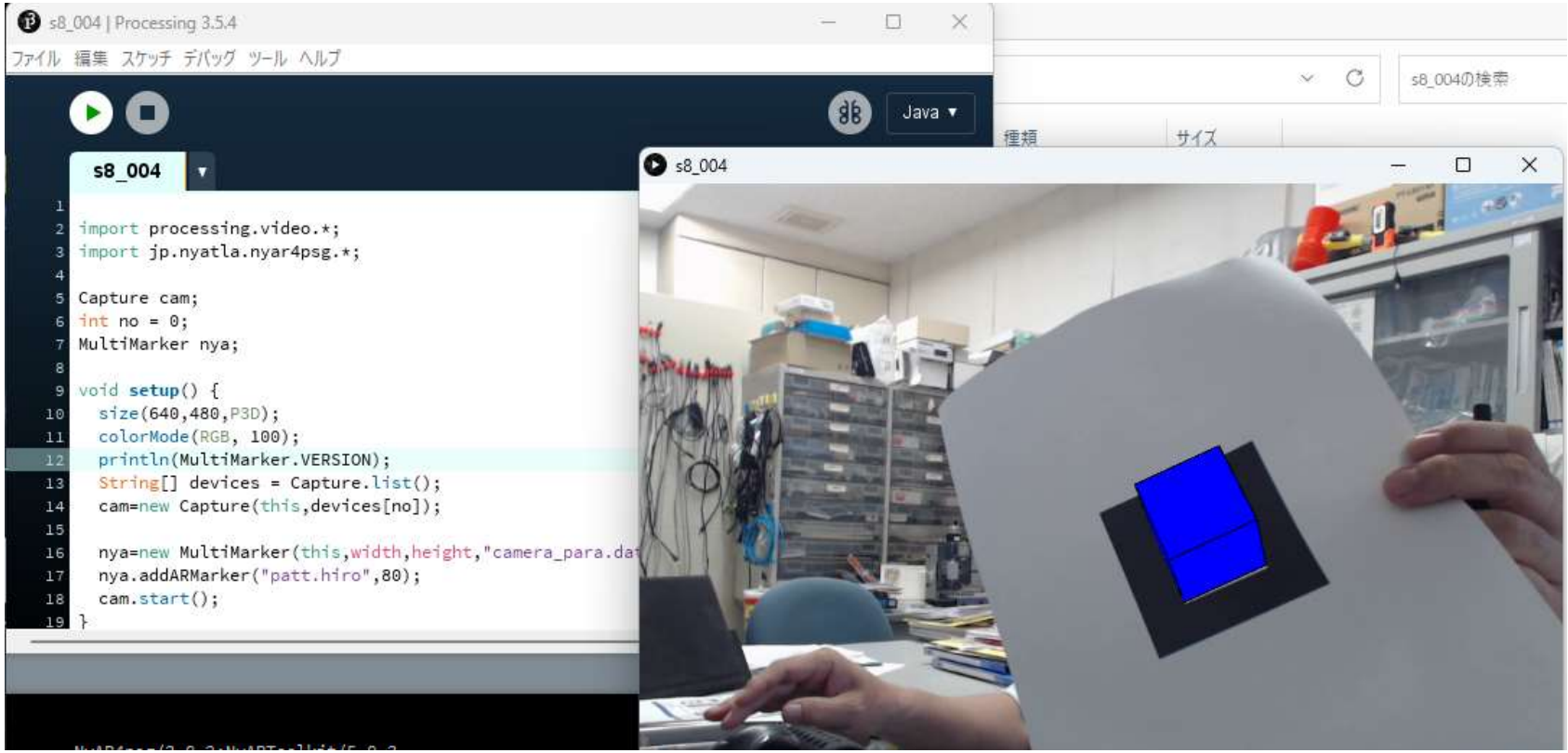
とても簡単！

https://www.future-coders.net/blog/6bv71vyscuamgan81yjcpl/

https://qiita.com/hextomino/items/3e6821bc0da133dae531

AR toolkitで複合現実

＜デモのみ＞



video libraryとの相性あり？
2.0（Library Managerでインストール）でないとErrorが起きる
しかし2.0は内部カメラでエラー。一度、外部カメラで動作させるとOK。

ArToolkitはNyar4psg-3.0.2.zipで動作。最新版3.0.10は×（Javaバージョン新しい）
https://github.com/nyatla/NyARToolkit-for-Processing/releases

作品を感動的にするために

いろいろな物理現象とシンクロすることが面白いが…

リアルタイムな応答処理 → アニメーション＋Midi
プログラミング能力が必要となる

イベントドリブン応答処理 → 画像処理＋Midi
動画の変化をトリガとして音再生

→ アニメーション＋音信号処理+センサ
センサ変化をトリガとして映像/音再生

環境的な映像/音の利用 → ビデオ再生＋mp3再生

いずれもコンピュータ処理の重さを意識した作りこみが重要となる
（特にビデオは過負荷なので，多用はお勧めしない）

適材適所で“驚き”を付け加えるワンポイントとして活用すべき

Final sample

学んだことがどこかで皆さんの役に立てば幸いです