

気象モニターを作ろう（発展編）

—誰にでもできるプロトタイピング—

第10回 Processingでシリアル通信

<http://cms.db.tokushima-u.ac.jp/DAV/person/S10723/気象モニターを作ろう/>

川上 博

2015/10/22

今日のテーマ：シリアル通信

1. Arduinoからのセンサー値をProcessingに表示

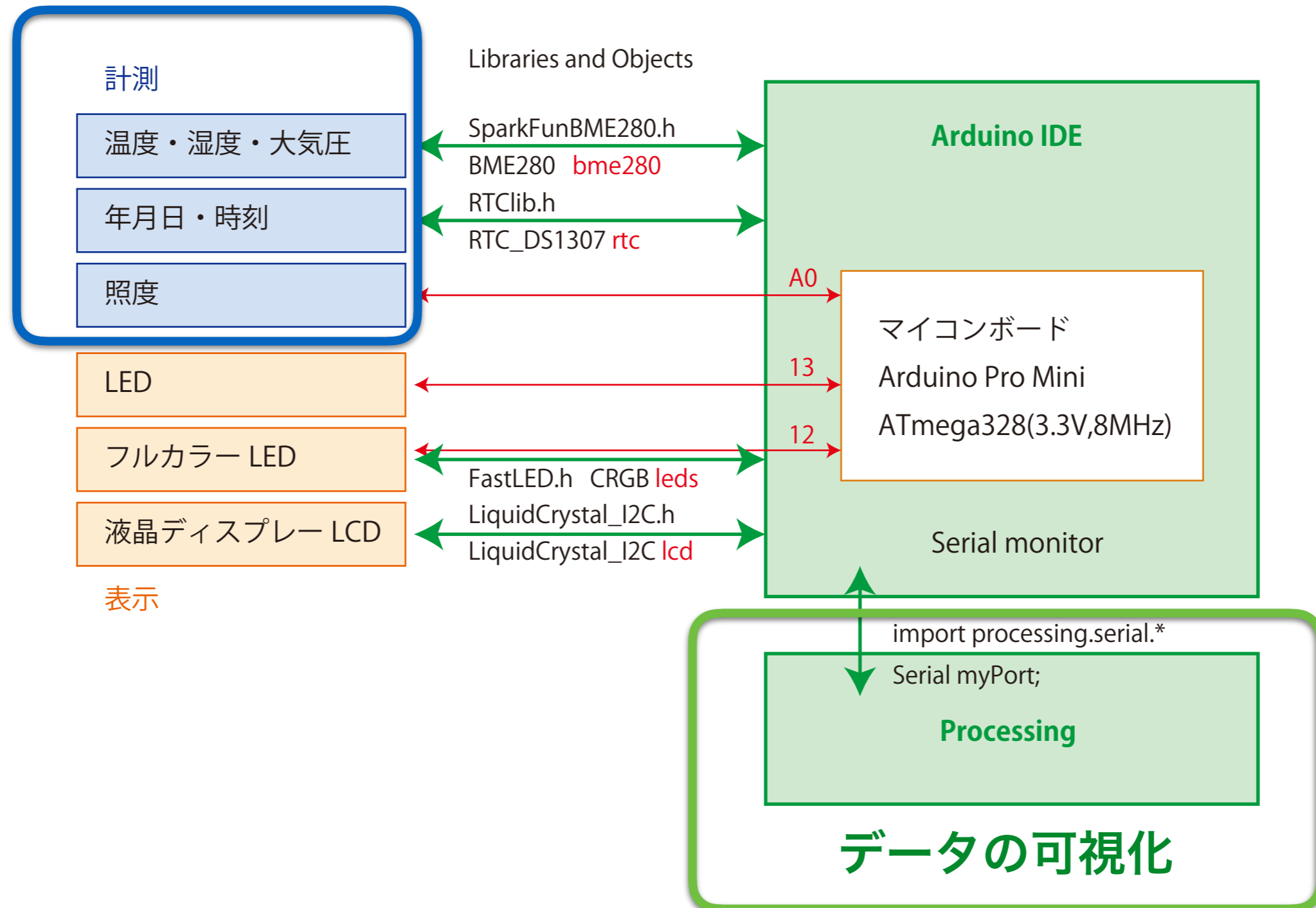
- ◎ Arduinoからのセンサー値をシリアル通信で受信

2. ProcessingのOOP : Object Oriented Programming

- ◎ スケッチを読みやすくする

JJ4 での計測・表示

無線で通信



Processing をシリアル・ポートにつなぐ

Processing からシリアル・ポートを使う

```
import processing.serial.*;

Serial myPort;

void setup() {
  size(500, 500);

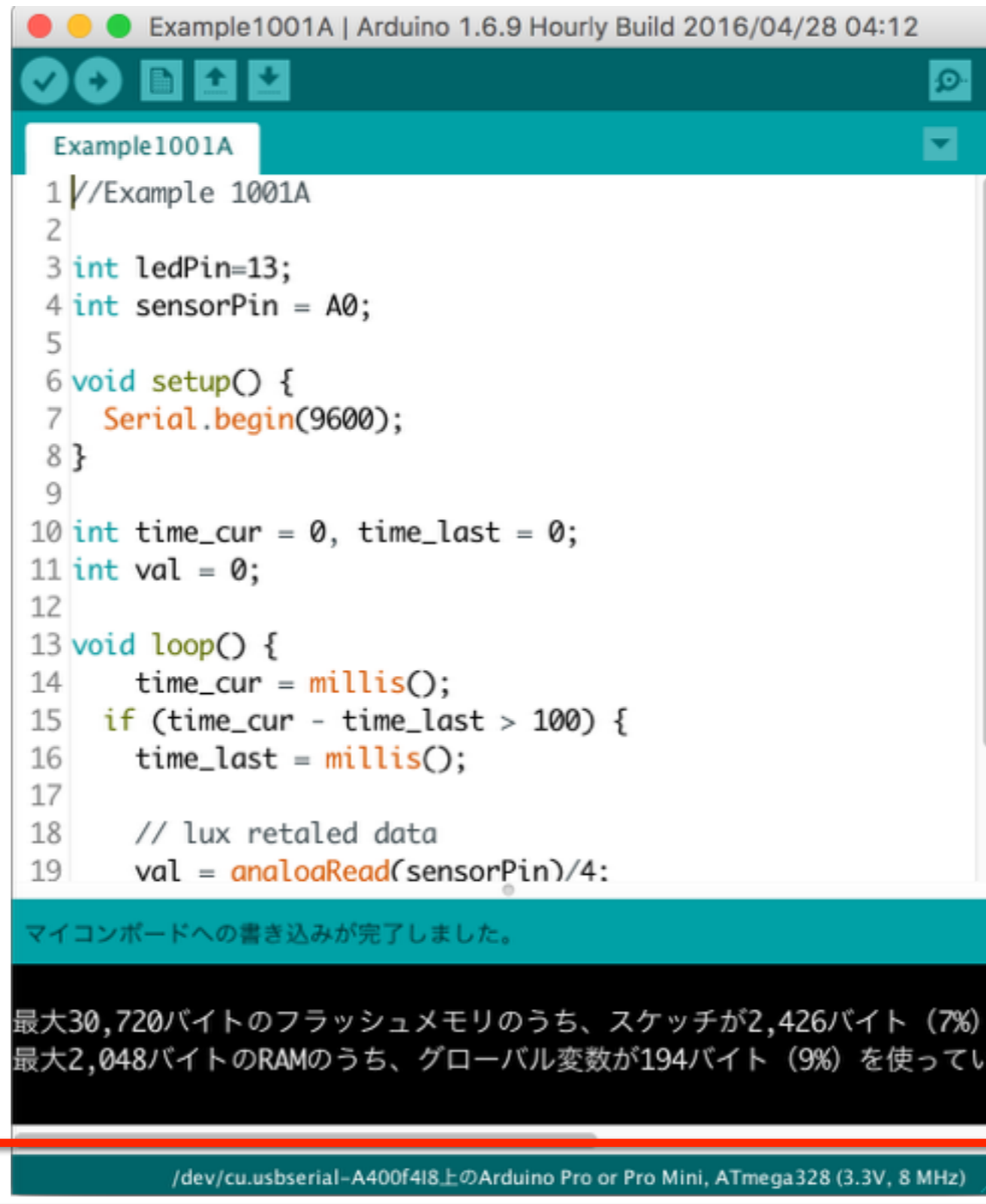
  println(Serial.list());
  // Serial.list()[0]="COM3"
  String arduinoPort = Serial.list()[0];
  myPort = new Serial(this, arduinoPort, 9600);
  // myPort = new Serial(this, "COM3", 9600);
  // myPort = new Serial(this, Serial.list()[0], 9600);
  background(255);
}
```

Arduinoが接続されているポートの番号



シリアル・ポートの番号

Example1001A (Arduinoのスケッチ) を実行してポート番号の表示を見る



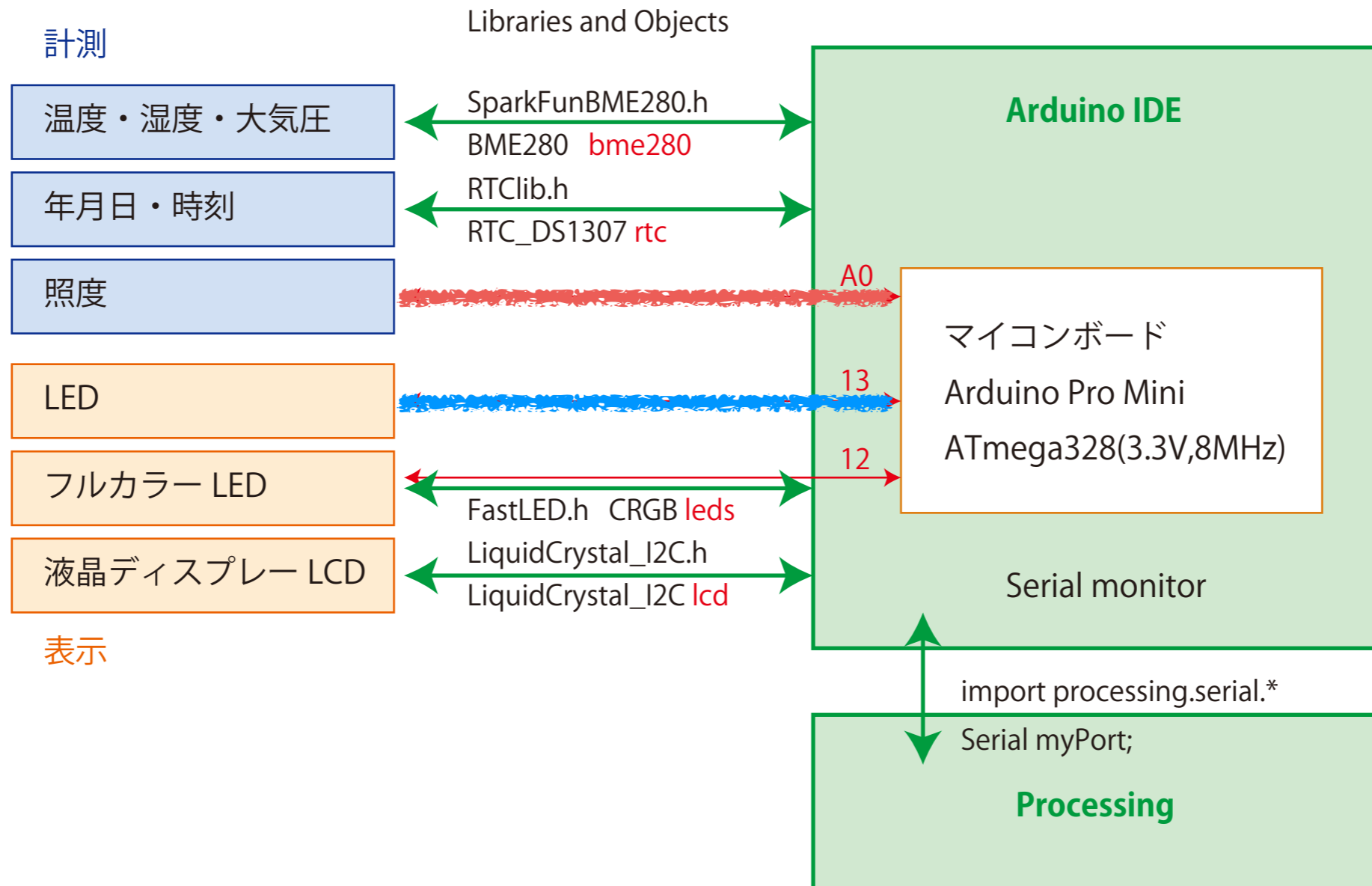
```
Example1001A | Arduino 1.6.9 Hourly Build 2016/04/28 04:12
Example1001A
1 //Example 1001A
2
3 int ledPin=13;
4 int sensorPin = A0;
5
6 void setup() {
7   Serial.begin(9600);
8 }
9
10 int time_cur = 0, time_last = 0;
11 int val = 0;
12
13 void loop() {
14   time_cur = millis();
15   if (time_cur - time_last > 100) {
16     time_last = millis();
17
18     // lux retaled data
19     val = analogRead(sensorPin)/4;
```

マイコンボードへの書き込みが完了しました。

最大30,720バイトのフラッシュメモリのうち、スケッチが2,426バイト (7%)
最大2,048バイトのRAMのうち、グローバル変数が194バイト (9%) を使っている

`/dev/cu.usbserial-A400f418上のArduino Pro or Pro Mini, ATmega328 (3.3V, 8 MHz)`

Example 1001A <-> Example 1001P



Example1001A

```
int ledPin=13;
int sensorPin = A0;
```

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

```
int time_cur = 0, time_last = 0, val=0;
```

```
void loop() {
  time_cur = millis();
  if (time_cur - time_last > 100) {
    time_last = millis();

    val = analogRead(sensorPin)/4;
    Serial.write(val);
    // Serial.println(val); //for serial monitor
    analogWrite(ledPin, val);
  }
}
```

100ミリ秒毎に送信



Example1001P

```
import processing.serial.*;

Serial myPort;

void setup() {
  size(500, 500);
  String arduinoPort = Serial.list()[1];
  myPort = new Serial(this, arduinoPort, 9600);
  background(255);
}

int x;
float val, lux, data;

void draw() {
  if ( myPort.available() > 0) {
    val = myPort.read();
    lux=222.0*val*3.3/256.0;
    data = map(lux, 0, 1000, height-50, 50);
  }
  showData(data);
}
```

受信が可能になった？



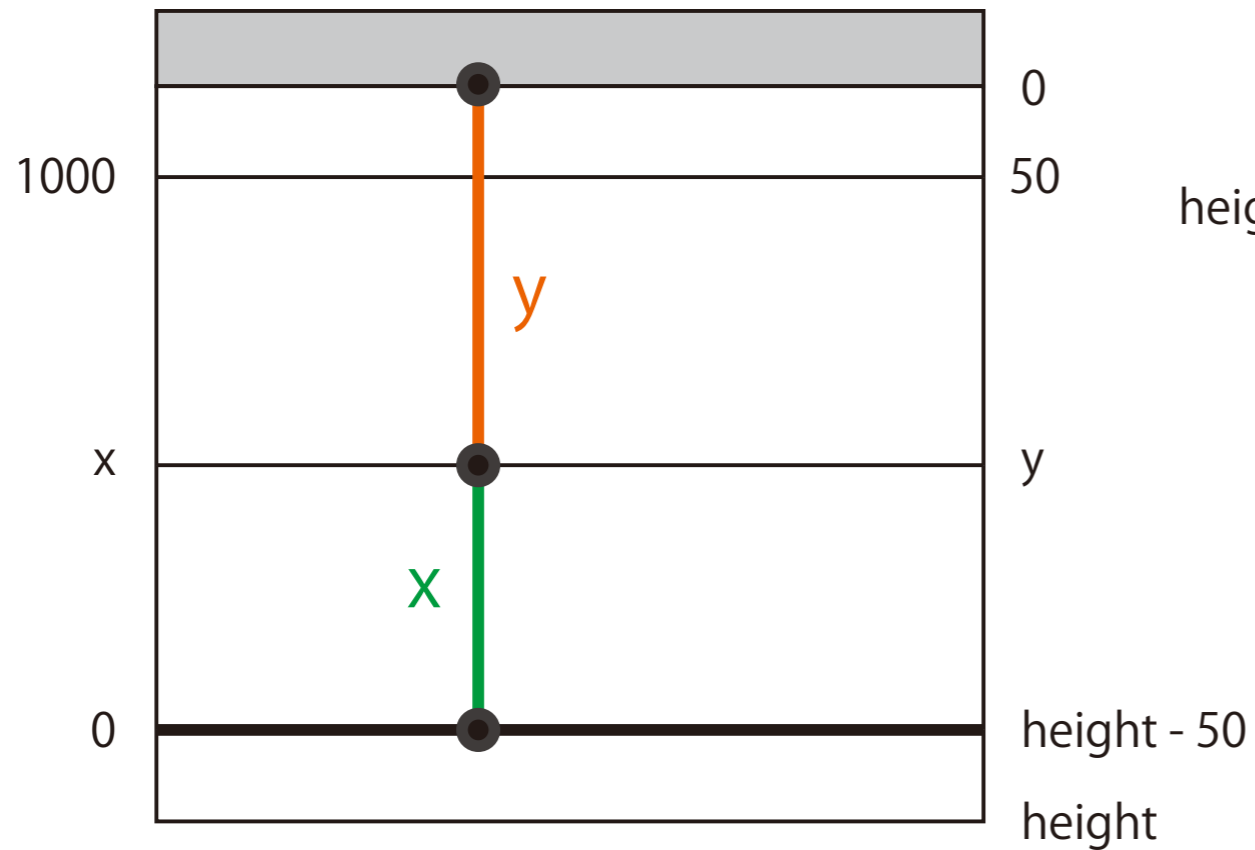
データを読む



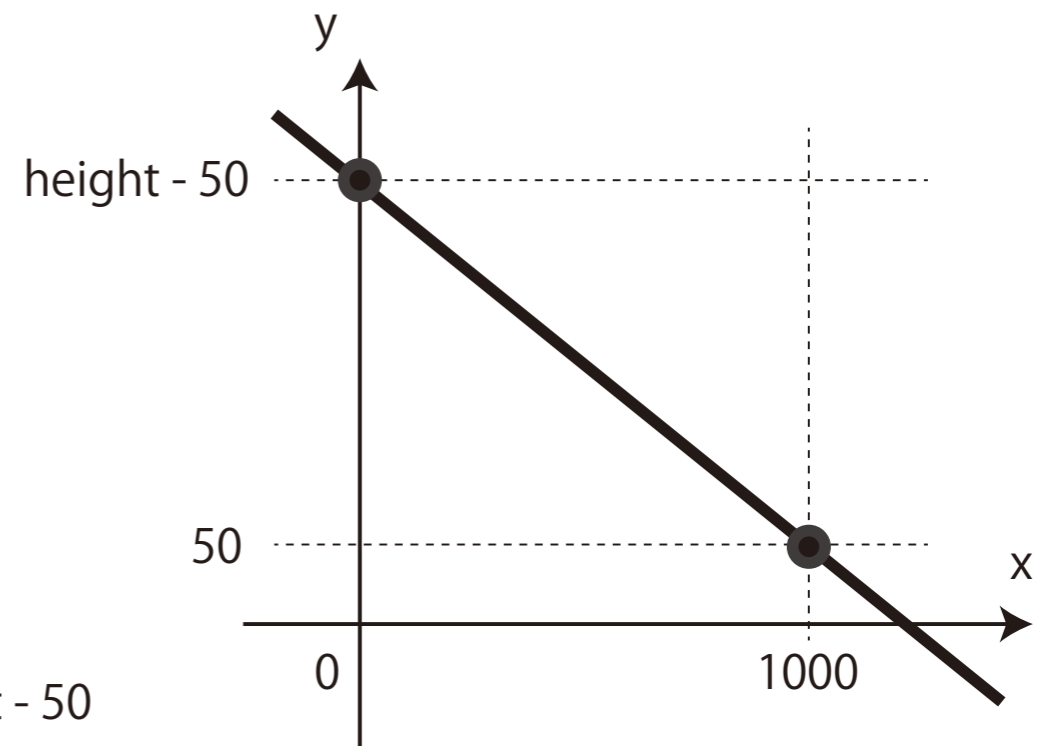
luxに変換

この変換で縦軸を反転させている

map関数を使う



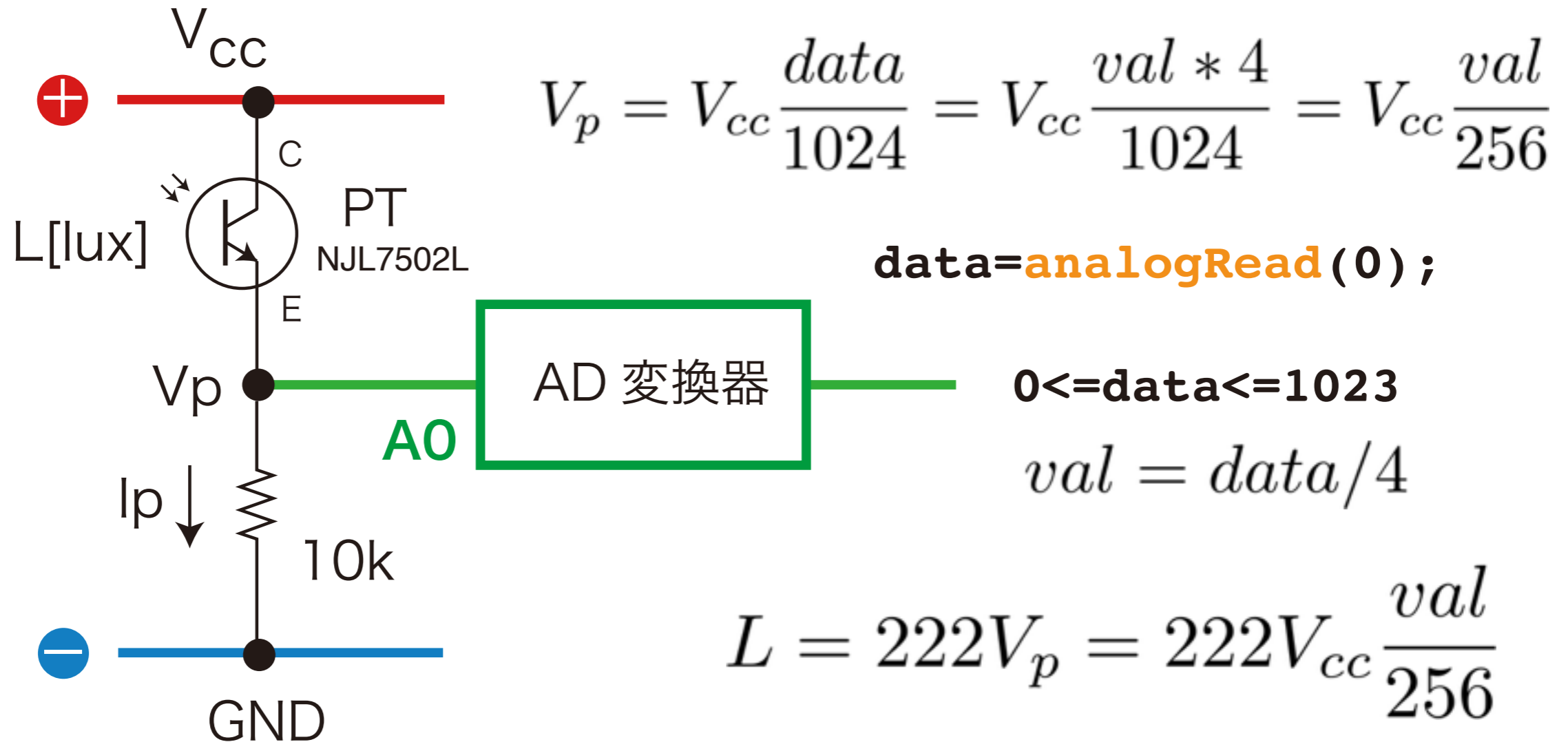
$$y = \text{map}(x, 0, 1000, \text{height} - 50, 50)$$



$$y = -\frac{\text{height}-100}{1000}x + \text{height} - 50$$

光センサーの値を照度に換算する

春夏第3回スライド pp.23-26



$$V_p = I_p \times 10 \times 10^3 \times 10^{-6} \Rightarrow I_p = 100V_p$$

$$L = 2.22 \times I_p = 222V_p$$

```
void showData(float data){
    strokeWeight(1);
    stroke(255);
    line(x, 0, x, height);
    stroke(0,255,0);
    line(x+1, 0, x+1, height);
    strokeWeight(4);
    stroke(255,0,0);
    point(x,data);
    x++;
    if (x > width) {
        x = 0;
    }
}
```



Example1002A

Example1002A (Arduinoのスケッチ) 照度データをASCII形式で送る

```
void loop() {  
  time_cur = millis();  
  if (time_cur - time_last > 1000) {  
    time_last = millis();
```

```
    // lux retaled data  
    val = analogRead(sensorPin);  
    lux = 222.0*val*3.3/1024.0;  
    Serial.print("H"); // data[0]  
    Serial.print(",");  
    Serial.println(lux, 1);
```

H,318.3\LF

```
    analogWrite(ledPin, val/4);
```

こちらはLED明滅

```
  }  
}
```

Example1002P

```
import processing.serial.*;

Serial myPort;
Indicator myInd;

char HEADER='H';
short LF=10;

volatile float[] thp=new float[2];

void setup() {
  size(400, 300);
  background(0);
  myPort=new Serial(this, Serial.list()[1], 9600);
  myPort.bufferUntil('\n');

  myInd=new Indicator(200,200,300);
}
```



```
volatile float[] thp=new float[2];
volatile int received = 0;
```

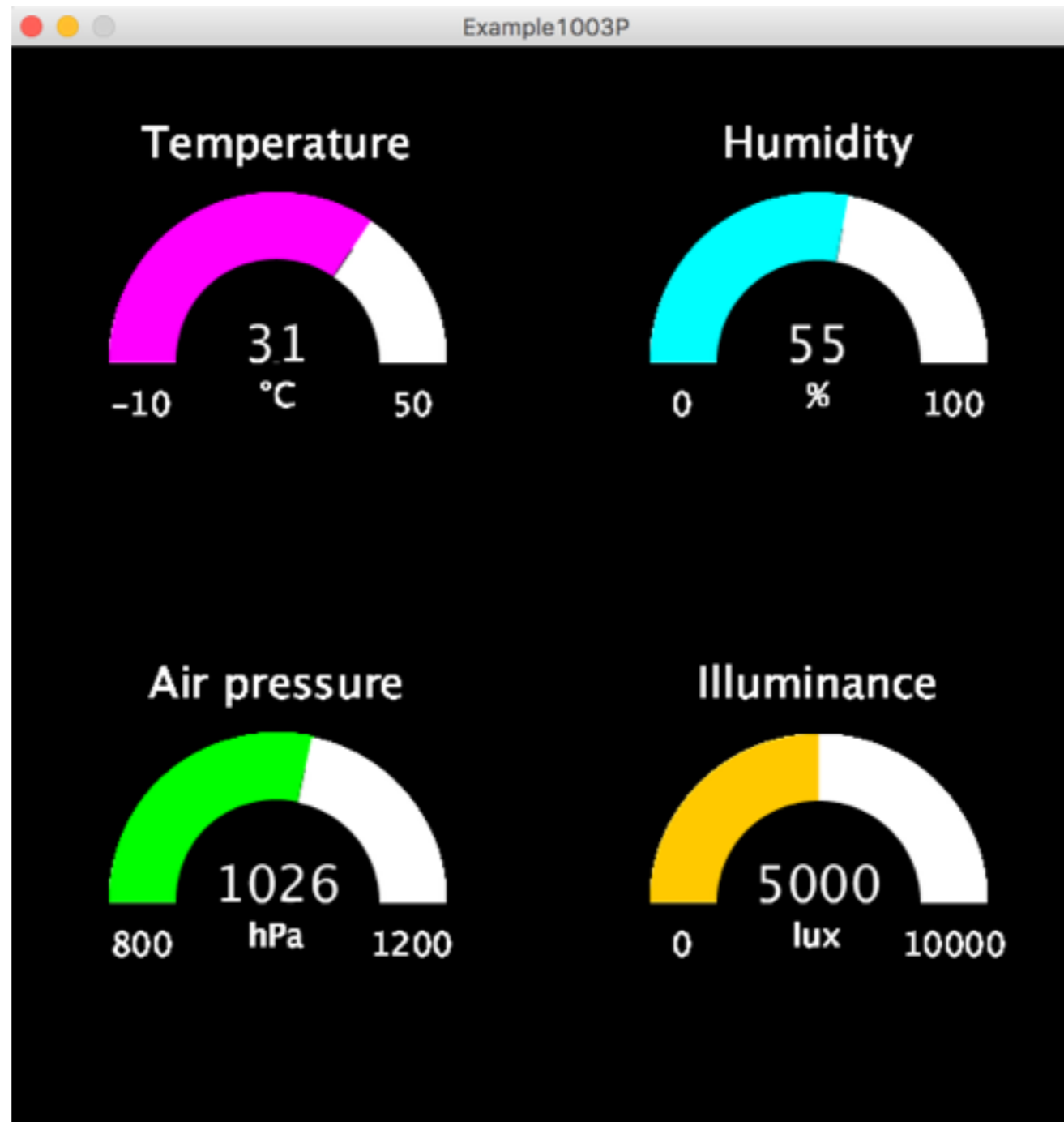
```
void draw() {
    if (received == 1) {
        received = 0;
        myInd.showData(thp[1]);
    }
}
```

```
void serialEvent(Serial myPort) {
    String message=myPort.readStringUntil(LF);
    if (message != null) {
        print(message);
        String [] data=message.split(",");
        if (data[0].charAt(0)==HEADER && data.length > 1) {
            for (int i=0; i < data.length; i++) {
                thp[i]=float(data[i]);
            }
        }
        received = 1;
        myPort.clear();
    }
}
```



H,318.3\\LF

Example1003Pの実行結果



照度の表示を完成させてください

Example 1003A

```
void loop() {  
  time_cur = millis();  
  if (time_cur - time_last > 1000) {  
    time_last = millis();  
    // temperature, humidity, pressure  
    Serial.print("H"); // data[0]  
    Serial.print(",");  
    Serial.print(bme280.readTempC(), 1); // data[1]  
    Serial.print(",");  
    Serial.print(bme280.readFloatHumidity(), 1); // data[2]  
    Serial.print(",");  
    Serial.println(bme280.readFloatPressure() / 100.0, 0); // data[3]  
  }  
}
```



**アナログポートA0から照度データを入力し、
シリアル通信で送る**

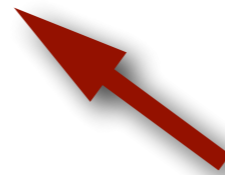
Example 1003P

```
void draw() {
    if (received == 1) {
        received = 0;
        thp[0]=thp[0]+noise(t)*5.0;
        thp[1]=thp[1]+noise(t)*10.0;
        thp[2]=thp[2]+noise(t)*20.0;
        myTemp.showData(thp[0]);
        myHum.showData(thp[1]);
        myPres.showData(thp[2]);
        myLux.showData(5000.0);
        t+=0.1;
    }
}
```



class MyLux extends Indicator

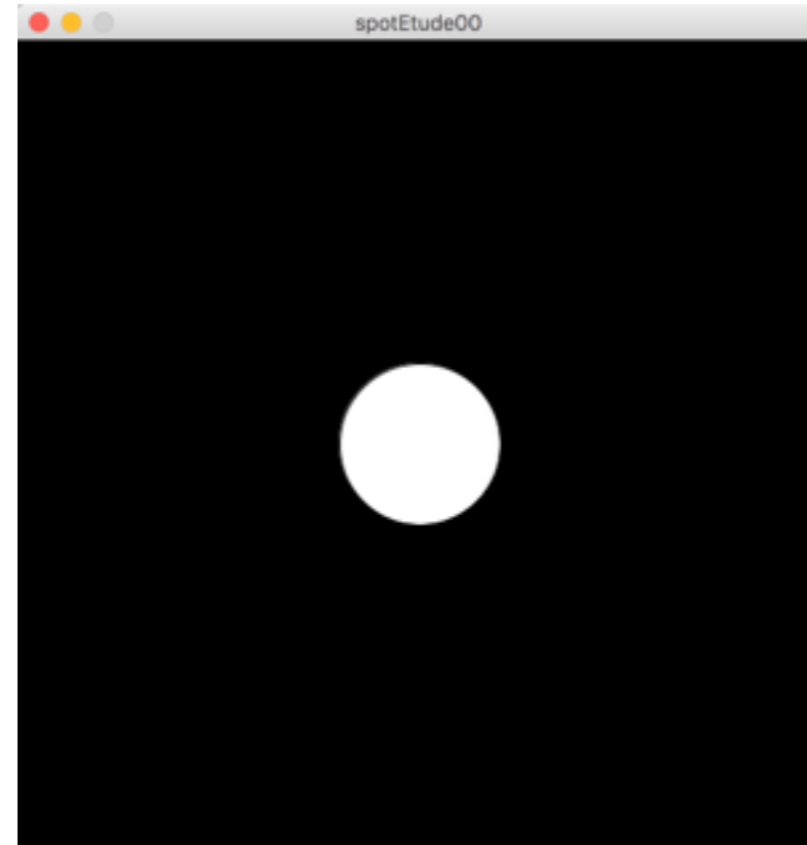
```
void showData(float data){
    super.displayText(min, max, name);
    super.displayData(data, min, max, col, illUnit);
}
```



Processing と OOP

```
//00P01P
void setup(){
  size(500,500);
}

void draw(){
  background(0);
  ellipse(width/2, height/
2, 100,100);
}
```



```
//00P02P
float x=250, y=250, diameter=100;

void setup(){
  size(500,500);
}

void draw(){
  background(0);
  ellipse(x,y,diameter,diameter);
}
```

Object = 変数 + 関数

```
//00P03P
Spot sp;

void setup(){
  size(500,500);
  sp=new Spot(250.0,250.0,100.0);
}

void draw(){
  background(0);
  ellipse(sp.x,sp.y,sp.diameter,sp.diameter);
}

class Spot{
  float x, y, diameter;
  Spot(float tx, float ty, float td){
    x=tx; y=ty; diameter=td;
  }
}
```

```
//00P04P
Spot sp;

void setup(){
    size(500,500);
    sp=new Spot(250.0,250.0,100.0);
}

void draw(){
    background(0);
    sp.display();
}

class Spot{
    float x, y, diameter;
    Spot(float tx, float ty, float td){
        x=tx; y=ty; diameter=td;
    }

    void display(){
        ellipse(x,y,diameter,diameter);
    }
}
```

```
//00P05P
Spot sp1,sp2;

void setup(){
    size(500,500);
    sp1=new Spot(150.0,250.0,100.0);
    sp2=new Spot(350.0,250.0,100.0);
}

void draw(){
    background(0);
    sp1.display();
    sp2.display();
}

class Spot{
    float x, y, diameter;
    Spot(float tx, float ty, float td){
        x=tx; y=ty; diameter=td;
    }

    void display(){
        ellipse(x,y,diameter,diameter);
    }
}
```

```
//00P06P
Spot[] sp=new Spot[2];

void setup(){
    size(500,500);
    sp[0]=new Spot(150.0,250.0,100.0);
    sp[1]=new Spot(350.0,250.0,100.0);
}

void draw(){
    background(0);
    sp[0].display();
    sp[1].display();
}

class Spot{
    float x, y, diameter;
    Spot(float tx, float ty, float td){
        x=tx; y=ty; diameter=td;
    }

    void display(){
        ellipse(x,y,diameter,diameter);
    }
}
```



```
//00P07P
Spot[] sp=new Spot[20];

void setup(){
  size(500,500);
  background(0);
  for(int i=0; i<sp.length;i++){
    float u=random(500),v=random(500),w=random(100) ;
    sp[i]=new Spot(u,v,w);
    sp[i].display();
  }
}

void draw(){}

class Spot{
  float x, y, diameter;
  Spot(float tx, float ty, float td){
    x=tx; y=ty; diameter=td;
  }

  void display(){
    ellipse(x,y,diameter,diameter);
  }
}
```