

```

/*
 * Robotrace Gate System シリアル通信
 * Processing 側
 * 開発環境 Ver3.4
 */

// シリアルライブラリを取り入れる
import processing.serial.*;

// myPort (任意名) というインスタンスを用意
Serial myPort;

/*===== 変数宣言 =====*/
int x; //図形の X 座標の変数を用意
int NUM = 3; //センサーの数
int[] receivedData = new int[NUM]; //センサーの値を格納する配列
int secondIcntM = 0; // OUT コース:分
int secondIcntS = 0; // OUT コース:秒
int secondITIME = 0; // OUT コース:センチ秒

int couseoutOUT = 0;
int cntOUT = 0;

/*===== 初期設定 =====*/
void setup(){
  // 画面サイズ
  size(1080, 780); // 1024 x 768 だと下が切れる
  textFont(createFont("x", 15)); // 日本語を表示用。バージョンアップで不要?
  //frameRate(10); // 1 秒間に 10 回 draw()実行 (使うかもしれないので残す)

  // シリアルポートの設定
  // Arduino が接続されているシリアルポートにあわせて書き換える
  myPort = new Serial(this, "COM6", 250000);
}

/*===== 描画ループ =====*/
void draw(){

  // 受信バッファから描画用変数に格納
  secondIcntM = receivedData[2];
  secondIcntS = receivedData[1];
  secondITIME = receivedData[0];

```

```

// 分,秒,センチ秒のバラバラのデータを時計形式に結合
String outTime = nf(secondIcntM, 2)+":"+nf(secondIcntS, 2)+":"+nf(secondITIME, 2);
println(outTime);          // デバッグ用に出力

background(255);          // 背景を指定色:白で塗りつぶす

//===OUT コースタイム表示===
textSize(90);             // 文字サイズ 90
fill(0, 0, 255);         // 文字色:BLUE
text("TIME", 30, height*25/100); // 表示するテキスト, x 座標, y 座標 30,15/100

textSize(220);           // 文字サイズ 220
fill(0, 0, 255);         // 文字色:BLUE
text(outTime, 100, height*60/100); //80,40/100

cntOUT++;

}

void serialEvent(Serial p){
  if ( myPort.available() >= 4 ) { // ヘッダ + 時間データ で合計7バイト
    if ( myPort.read() == 'H' ) { // ヘッダ文字を見つけたところから読み取る
      receivedData[0] = myPort.read(); // OUT:分読み込み
      receivedData[1] = myPort.read(); // OUT:秒読み込み
      receivedData[2] = myPort.read(); // OUT:センチ秒読み込み
    }
  }
}

void keyPressed(){
  if(key == 'o'){
    if(couseoutOUT == 0 && cntOUT > 10){
      cntOUT = 0;
      couseoutOUT = 1;
    }
    if(couseoutOUT == 1 && cntOUT > 10){
      cntOUT = 0;
      couseoutOUT = 0;
    }
  }
}
}

```