


```

void loop(){

    for(int i=0; i < picsize; i++){
        display(picture[i]);
        sensval = analogRead(sensor);
        y = sensval;
        Serial.print("y,");
        Serial.println(y);
    }
}

void display(char* pic){
    delaytime = ave/picsize;

    int digits[picsize];
    for(int i = 0; i <10; i++){
        digits[i] = pic[i]-'0';
    }

    for(int i = 0; i <10; i++){
        if(digits[i]==0){
            digitalWrite(ledarray[i],LOW);
        }
        else if(digits[i]==1){
            digitalWrite(ledarray[i],HIGH);
        }
    }

    delay(delaytime);
}

void calibrate(){
    for(int i=0; i<10; i++){
        digitalWrite(ledarray[i],HIGH);
        delay(30);
    }
    for(int i=9; i>-1; i--){
        digitalWrite(ledarray[i],LOW);
        delay(30);
    }

    for(int i=0; i<10; i++){
        digitalWrite(ledarray[i],HIGH);
        time = millis();
        //set
        //基準
        int base = analogRead(sensor);
        delay(100);

        //right to left

        while(1){
            y = analogRead(sensor)-base;
            if(y<-150){
                time = millis();
                break;
            }
        }

        i++;
        digitalWrite(ledarray[i],HIGH);
        //left to right

        while(1){
            y = analogRead(sensor)-base;
            if(y>150){
                time = millis() - time;
                break;
            }
        }
    }
    ave =ave+time;
}
ave = ave/5;
Serial.println(ave);

```

```
}
```

4.実行結果

加速度センサーをもとにしたデータより, `delay()`の時間を指定できるようにした.

5.考察

プログラムでは加速度センサーを用いて, POVを振るタイミングにキャリブレーションできるようにした.

その手順として

- POVを作る前に, 10回ほど一定の速度で振ってもらう
- 加速度の変換を用いて, 右から左移動している時間, その逆を`millis()`を用いて計測する.
- 計測した時間の平均は, およそ1 振幅の長さである.
- 1 振幅の間にPOVを出し切りたいので, 図を格納している`picture`配列の大きさと1 振幅の長さを割る.

`picture`配列の大きさと1 振幅の長さを割った時間が一回一回`delay()`すべき時間となる.

6.参考文献

「情報科学基礎実験! 第2章Arduinoを用いた基礎的な実験」テキスト

7.謝辞

この実験をレポートとして形にすることが出来たのは、ペアの杉崎さん、TAの皆様にご協力いただいたおかげです。協力していただいた皆様へ心から感謝の気持ちと御礼を申し上げたく、謝辞にかえさせていただきます。