

# システム実験

## 実験後期第4回レポート

6119019056 山口力也

2019/10/31 日提出

### 1 レポート 11.4.1

課題 11.4.1(基本動作の確認) では, カラーセンサーのキャリブレーションを行なっている. 具体的にどのようなキャリブレーションを行なっているか, プログラムの中身を読んで説明せよ.

センサーが値を読み込むたびにそれぞれ red, green, blue のセンサーの値を足して行き, それをカウントで割って行く. カウントもセンサーの値を読み込むたびに増えて行くためカラーセンサの読み取った値の平均値を調べていることになる.

### 2 レポート 11.4.2

課題 11.4.2(bang-bang 制御によるライントレース) において,  $K_p$  の値の違いによってライントレースがどのように変化したか記述せよ. また, bang-bang 制御の問題点を説明せよ.

$K_p$  の値を増やすと zumo の左右に動く幅が広がりより蛇行するようになった.  $K_p$  の値を小さくすると zumo の左右に動く幅は小さくなったが, 急カーブなどは曲がれなくなった.

bang-bang 制御では目標値に対して常に一定の倍率 ( $K_p$ ) でしか速度を制御できないので, 目標値を超えるか目標値を下回るかでつねに振動してしまうのが問題点である.

### 3 レポート 11.4.3

課題 11.4.3(P 制御によるライントレース) に関し, 下記 (a) (d) ごとに分けてレポートを作成せよ. (a) 作成したプログラムを報告せよ. (b) bang-bang 制

御と P 制御の場合でライントレースの動作がどのように違ったか記述し, なぜそのようになるのか原理を説明せよ.(c)  $K_p$  の値の違いによってライントレースがどのように変化したか記述せよ.(d) 高速かつスムーズにライントレースを行うことができた  $K_p$  の値と speed の値を報告せよ. また, 赤, 緑, 緑, 赤の区間を通過した際の RGB の値の変化のグラフを貼り付け, どのような特徴があるか考察せよ.

### 3.1 (a)

以下ソースコード 1 に作成したプログラムを報告する.

ソースコード 1: P 制御

```
1 if ( lightNow < (lightMin + lightMax) / 2.0 ) // 右回転
2     //P 制御,lightNow の値に応じて speedDiff の値を線形に変化
3     speedDiff = map(lightNow,lightMin,(lightMin+lightMax) /
4         2.0, -Kp*speed,0);
5 else // 左回転
6     //P 制御,lightNow の値に応じて speedDiff の値を線形に変化
7     speedDiff = map(lightNow,(lightMin+lightMax) / 2.0,lightMax
8         ,0,Kp*speed);
9 motorL_G = speed - speedDiff;
10 motorR_G = speed + speedDiff;
```

### 3.2 (b)

bang-bang 制御では目標値に対して振動していたが,P 制御では目標値に近ければ近いほど操作量を減らすため, 直線でも目標値に近づくと振動せずに一定の動きをしていた.

### 3.3 (c)

$K_p$  の値が大きくなると急カーブには対応できるようになったが, 少し bang-bang 制御に近いような動きになった. $K_p$  の値が小さいとスムーズではあるが急カーブなどは対応できなかった.

### 3.4 (d)

高速かつスムーズにライントレースできた値はそれぞれ  $K_p = 1.9$ , speed=100 であった. また, 以下図 1, 図 2 に直線と曲線の場合の図を示す.

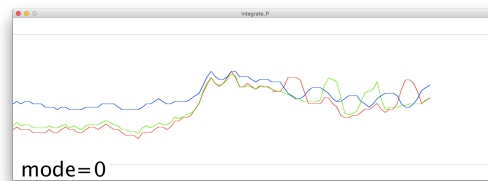


図 1: 直線の場合

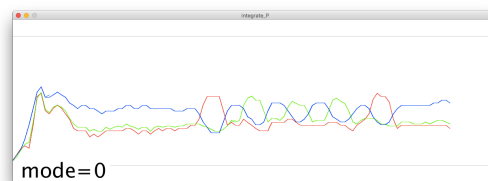


図 2: 曲線の場合

曲線の方が全体的な RGB 値が直線に比べて低かった。

## 4 レポート 11.4.4

課題 11-4.4 では loop 関数を制御の基本単位としたプログラミング方針を学んだ。このプログラム方針の輪転を記述せよ。

mode によって状態を遷移するので他の記述部分は見なくてよいので 1 回の loop にかかる実行速度が短くなる。また、コードも読みやすくわかりやすい。

## 5 レポート 11.4.5

課題 11-4.5 で作成したプログラムを報告せよ。また、どのような状態遷移を行なったか説明せよ。

以下ソースコード 2 に作成したプログラムを報告する。

### ソースコード 2: 課題 11-4.5

```
1 switch ( mode_G ) {
2     case 0:
3         mode_G = 1;
```

```

4      break; // break 文を忘れない（忘れるとその下も実行される）
5
6      case 1:
7          linetrace_P(); // ライントレース(各自で作成)
8          color = identify_RGB(); // ラインの色を推定 (R:赤,G:緑,B:
           青, -:それ以外)
9          if ( color == 'R' ) { // red //一回目の赤
10             mode_G = 2;
11         }
12         break;
13     case 2:
14         linetrace_P();
15         color = identify_RGB();
16         if( color == 'B' ) {
17             mode_G = 3;
18         }
19         break;
20     case 3:
21         linetrace_P(); // ライントレース
22         color = identify_RGB();
23         if ( color == 'R' ) { // 2 回目も赤だったら
24             startTime = timeNow_G; // mode_G=3に遷移した時刻を記録
25             mode_G = 4;
26         }
27         if ( color == 'G' ) { //2回目が緑だったら
28             green_count++;
29             mode_G = 5;
30         }
31         break;
32
33     case 4:
34         motorL_G = 100;
35         motorR_G = 100;
36         if ( timeNow_G - startTime > run_period ) // 指定時間経
           過したら
37             mode_G = 1;
38         break;
39     case 5:
40         linetrace_P();
41         color = identify_RGB();
42         if (color == 'B') {
43             mode_G = 6;
44         }
45         break;
46     case 6:
47         linetrace_P();

```

```

48     color = identify_RGB();
49     if( color == 'G') { //緑だったらカウント
50         green_count++;
51         mode_G=5;
52     }
53     if ( color == 'R') { //赤だったら終了
54         motorL_G = -100;
55         motorR_G = 100;
56         delay(500);
57         mode_G = 7;
58     }
59     break;
60 case 7:
61     color = identify_RGB();
62     motors.setSpeeds(100,100);
63     if( color == 'B') {
64         mode_G = 8;
65     }
66     break;
67 case 8:
68     color = identify_RGB();
69     motors.setSpeeds(100,100);
70     if( color == 'W') {
71         mode_G = 9;
72     }
73     break;
74 case 9:
75     delay(1000);
76     for(int i = 0; i < green_count; i++) {
77         motors.setSpeeds(100, 100); // 直進
78         delay(500);
79         motors.setSpeeds(0, 0); // 停止
80         delay(500);
81     }
82     mode_G = 10;
83     break;
84 case 10:
85     motors.setSpeeds(0,0);
86     break;
87 }

```

---

また, 以下図 3 にフローチャート図を示す.

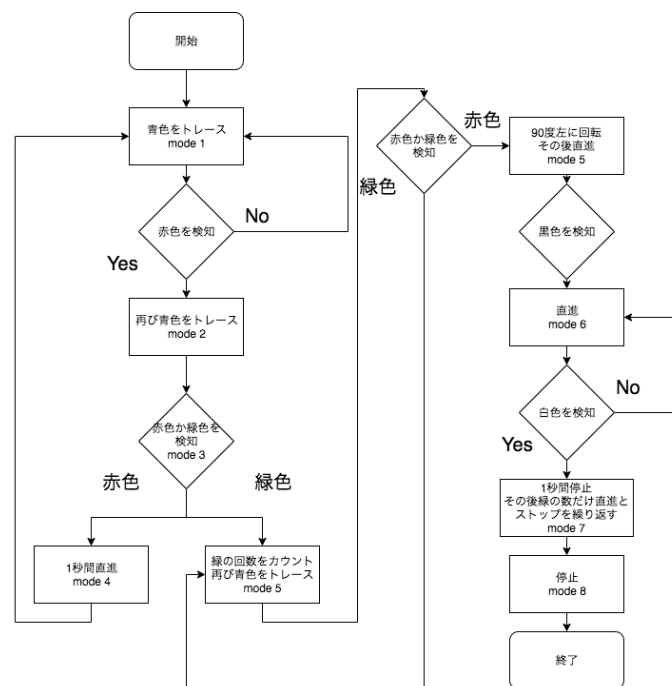


図 3: フローチャート図