

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

【作業内容】

文字の担当を決める

決まった担当の文字をどのようにするか検討する

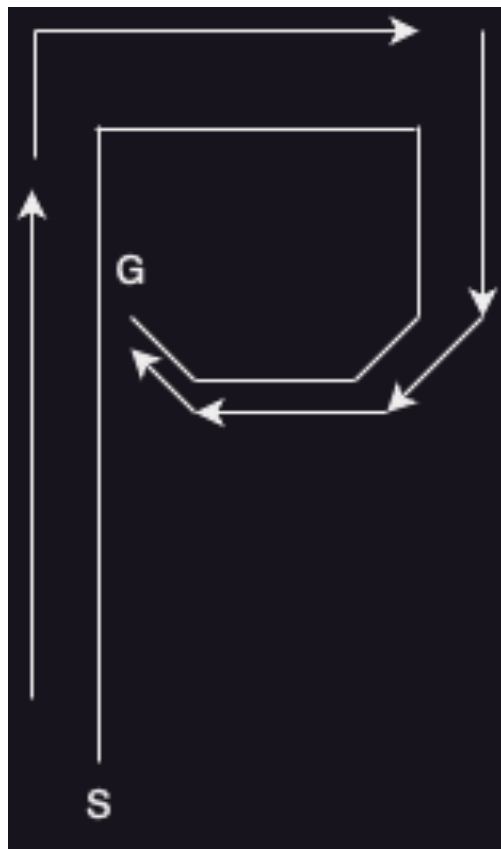
【作業項目】

- ① 文字の担当を決める。

自分は P の文字になった。

- ② P の文字を攻略するための方法を検討する

P は比較的簡単な文字だと考える。やり方としていくつか考えようかと思ったが、どう考えても 1 つしかやり方が浮かばないのでそのやり方でプログラムを変えてみる。



このように右回りで回っていく。前にラインが繋がっていなければ左のモーターの出力を強くする。このようにやれば右回りでコースを全て走破することができる。

- ③ このコースをクリアするプログラムを以下に示す。

④ `#include <stdio.h>`

⑤ `#include <stdlib.h>`

⑥ `#include <wiringPi.h>`

⑦ `#include <wiringPiI2C.h>`

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

```
⑧
⑨ // PWM ユニットの I2C アドレス
⑩ #define PWMI2CADR 0x40
⑪
⑫ // PWM 制御に使う。（1 でブリッジ動作、0 はブリッジオフ）
⑬ #define ENA_PWM 8
⑭
⑮ // IN1 と IN2 は右車輪の回転方向を決める（後進 : 0, 1, 前進 : 1, 0）（0, 0 と 1, 1 はブレーキ）
⑯ #define IN1_PWM 9
⑰ #define IN2_PWM 10
⑱
⑲ // 左側のモーター : パワーユニットの K3 または K4 に接続
⑳ // ENB は PWM 駆動に使う（1 でブリッジ動作、0 はブリッジオフ）
21 #define ENB_PWM 13
22
23 // IN3 と IN4 は左車輪の回転方向を決める（後進 : 0, 1, 前進 : 1, 0）（0, 0 と 1, 1 はブレーキ）
24 #define IN3_PWM 11
25 #define IN4_PWM 12
26
27 // PWM モジュールのレジスタ番号
28 #define PWM_MODE1 0
29 #define PWM_MODE2 1
30 #define PWM_SUBADR1 2
31 #define PWM_SUBADR2 3
32 #define PWM_SUBADR3 4
33 #define PWM_ALLCALL 5
34
35 // PWM 番号 * 4 + PWM_0_??_? でレジスタ番号は求まる
36 #define PWM_0_ON_L 6
37 #define PWM_0_ON_H 7
38 #define PWM_0_OFF_L 8
39 #define PWM_0_OFF_H 9
40
41 // PWM 出力定数
42 #define PWMFULLON 16
```

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

```
43 #define PWMFULLOFF 0
44
45 // プリスケーラのレジスタ番号
46 // PWM 周波数を決めるレジスタ番号、100Hz なら 61 をセット
47 #define PWM_PRESCALE 254
48
49 // 光センサーのピン番号
50 #define GPIO_L 5
51 #define GPIO_ML 6
52 #define GPIO_M 13
53 #define GPIO_MR 19
54 #define GPIO_R 26
55
56 // motor_drive()から呼ばれる関数、PWM ユニットへの書き込みをする。
57 int set_pwm_output(int fd, int pwmch, int outval) {
58     int ef = 0;
59     int regno;
60
61     if ((pwmch < 0) || (pwmch > 15)) ef = 1; // チャンネルの指定違反チェック
62     if ((outval < 0) || (outval > 16)) ef = ef + 2; // 出力値の指定違反チェック
63     if (ef == 0) {
64         regno = PWM_0_ON_L + pwmch * 4; // 1ch あたり 4 レジスタで 16ch 分あるので
65         if (outval == 16) {
66             wiringPiI2CWriteReg8(fd, regno + 3, 0);
67             wiringPiI2CWriteReg8(fd, regno + 1, 0x10);
68         } else {
69             wiringPiI2CWriteReg8(fd, regno + 1, 0);
70             wiringPiI2CWriteReg8(fd, regno + 3, outval);
71         }
72     }
73     return ef; // エラーがなければ 0 が返る
74 }
75
76
77 // モーターを制御するための関数。
78 // fd は I2C 初期化時のファイルディスクリプタ（デバイス番号のようなもの）
79 // lm は左モーター、rm は右モーターの駆動数値で、-16~+16 の範囲で指定
```

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

```
80 // 負の場合は後ろ方向に回転、生の場合は前方向に回転
81 // 全体値が大きいほど、パワーが大きくなる
82 // PWM ユニット自体は 12 ビット制度だが、上位 4 ビット分を制御
83 int motor_drive(int fd, int lm, int rm) {
84     set_pwm_output(fd, ENA_PWM, 0); // 右のモーター無効化
85     set_pwm_output(fd, ENB_PWM, 0); // 左のモーター有効化
86     // 右モーターの制御
87     if (rm < 0) {
88         set_pwm_output(fd, IN1_PWM, 0); // OUT -> GND
89         set_pwm_output(fd, IN2_PWM, rm); // OUT2 -> +Vs
90         rm = abs(rm);
91     } else {
92         set_pwm_output(fd, IN1_PWM, rm); // OUT1 -> +Vs
93         set_pwm_output(fd, IN2_PWM, 0); // OUT2 -> GND
94     }
95
96     // 左モーターの制御
97     if (lm < 0) {
98         set_pwm_output(fd, IN3_PWM, 0); // OUT3 -> GND
99         set_pwm_output(fd, IN4_PWM, lm); // OUT -> +Vs
100        lm = abs(lm);
101    } else {
102        set_pwm_output(fd, IN3_PWM, lm); // OUT3 -> +Vs
103        set_pwm_output(fd, IN4_PWM, 0); // OUT4 -> GND
104    }
105    if (lm > 16) lm = 16;
106    if (rm > 16) rm = 16;
107    set_pwm_output(fd, ENA_PWM, rm); // 右モータースタート
108    set_pwm_output(fd, ENB_PWM, lm); // 左モータースタート
109    return 0;
110}
111
112int main() {
113    int fd;
114    wiringPiSetupGpio();
115    fd = wiringPiI2CSetup(PWMI2CADR);
116    if (fd < 0) {
```

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

```
117     printf("I2C の初期化に失敗しました。終了します。¥n");
118     exit(EXIT_FAILURE);
119 }
120 wiringPiI2CWriteReg8(fd, PWM_PRESCALE, 61);
121 wiringPiI2CWriteReg8(fd, PWM_MODE1, 0x10);
122 wiringPiI2CWriteReg8(fd, PWM_MODE1, 0);
123 delay(1);
124 wiringPiI2CWriteReg8(fd, PWM_MODE1, 0x80);
125
126 while(1){
127     if(digitalRead(GPIO_L) == LOW && digitalRead(GPIO_R) == LOW) break;
128 }
129
130 int ms, ls, rs;
131 while (1) {
132     ms = 0;
133     ls = 0;
134     rs = 0;
135     if (digitalRead(GPIO_L) == HIGH) {
136         printf("right¥n");
137         rs = 5;
138     }
139     else if (digitalRead(GPIO_ML) == HIGH) {
140         printf("middle right¥n");
141         rs = 3; ms=2;
142     }
143     else if (digitalRead(GPIO_M) == HIGH) {
144         printf("middle¥n");
145         ms = 6;
146     }
147     else if (digitalRead(GPIO_MR) == HIGH) {
148         printf("middle left¥n");
149         ls = 3; ms=2;
150     }
151     else if (digitalRead(GPIO_R) == HIGH) {
152         printf("left¥n");
153         ls = 5;
```

作業報告書（2023 年 6 月 1 日）

J20413 北野正樹

```
154     }  
155     else {  
156         printf("not_read¥n");  
157         ls=6;  
158     }  
159     motor_drive(fd, ms+ls, ms+rs);  
160     delay(50);  
161 }  
162 return 0;  
163}
```

④ 改善点

このプログラムを実行すると、カーブ部分でコースから外れてしまった。次はパラメータの調整をしていきたいと思う。

【作業時間】

- ・ 作業時間：90 分
- ・ 報告書作成時間：30 分