

無線で動くロボットを作ろう 第2回

前回第1回では ロボットの開発環境について学び、LED を点滅させるスケッチを作成し学んだ。

ロボット上にはLED1, LED2, LED3 が取り付けられている。それらを光らせてみた。

次のようなスケッチ例となる。

Baics01	
<pre>1 //Example01 2 const int LED_PIN=3; 3 const int LED_PIN2=4; 4 5 void setup() { 6 // put your setup code here, to run once: 7 pinMode(LED_BUILTIN, OUTPUT); 8 pinMode(LED_PIN,OUTPUT); 9 pinMode(LED_PIN2,OUTPUT); 10 } 11 12 void loop() { 13 // put your main code here, to run repeatedly: 14 digitalWrite(LED_BUILTIN, HIGH); 15 digitalWrite(LED_PIN,HIGH); 16 digitalWrite(LED_PIN2,LOW); 17 delay(1000); 18 19 digitalWrite(LED_BUILTIN, LOW); 20 digitalWrite(LED_PIN,LOW); 21 digitalWrite(LED_PIN2,HIGH); 22 delay(1000); 23 }</pre>	<p>LED_PIN を 3 とする LED_PIN2 を 4 とする LED の接続されているポート番号が分かっている。</p>
	<p>LED_BUILTIN は 13 と指定されている。 それらのポートを出力 OUTPUT のモードにする。</p>
	<p>LED_BUILTIN と LED_PIN を HIGH 点灯とする。 LED_PIN2 を LOW 消滅にする。 1000ms 1 秒待つ</p>
	<p>次に LED_BUILTIN と LED_PIN を LOW 消滅とする。 LED_PIN2 を HIGH 点灯にする。 1000ms 1 秒待つ。 全体が loop() で繰り返すようになっているので、 これで LED が点灯・消滅を繰り返す</p>

第2回 ロボットのモーターの動作の基本

1. ロボットの車輪を動かす例

今回の目標は

- ・ ロボットの車輪を動かし、前進・後退、向きの回転を行う。
- ・ 車輪の動きをいろいろ試してみる。これらの応用を考える。
- ・ スケッチのプログラミングの基本を学ぶ。

Example0101 を起動する 右の車輪の前進・後退と速度を指定して動かす。

もしくは Example0001a

Example0101

```
1 // 01: motor
2 void setup() {
3   /* モーター回転方向 */
4   /* 後退: HIGH */
5   /* 前進: LOW */
6   pinMode(7, OUTPUT); // 右回転方向
7 }
8
9 void loop() {
10  digitalWrite(7, LOW); // 右回転方向
11  analogWrite(9, 100); // 右PWM
12  delay(1000);
13
14  digitalWrite(7, HIGH); // 右回転方向
15  analogWrite(9, 100); // 右PWM
16  delay(1000);
17 }
18
```

pin7 を OUTPUT 出力用とする

pin7 に LOW(前進)を指定

pin9 に回転速度 100 を指定

delay 続ける時間 1000ms とする

pin7 に HIGH(後退)を指定
同様に実行

これで右車輪が前進と後退を繰り返す。

練習1) モーターの回転速度を変えてみるとどうなるか。例、200, 50 など

練習2) delay を変えるとどうなるか。delay はその時間動作を継続する時間となっている。

Example102 左に曲がる

Example0102

```
1 // 01: motor
2 const int MOTOR_R_CWCCW = 7;
3 const int MOTOR_R_PWM = 9;
4
5 void setup() {
6   /* モーター回転方向 */
7   /* 後退: HIGH */
8   /* 前進: LOW */
9   pinMode(MOTOR_R_CWCCW, OUTPUT); // 右回転方向
10 }
11
12 void loop() {
13   digitalWrite(MOTOR_R_CWCCW, LOW); // 右回転方向
14   analogWrite(MOTOR_R_PWM, 100); // 右PWM
15   delay(1000);
16
17   digitalWrite(MOTOR_R_CWCCW, HIGH); // 右回転方向
18   analogWrite(MOTOR_R_PWM, 100); // 右PWM
19   delay(1000);
20 }
```

Example102a 右に曲がる

Example0102a

```
1 // 01: motor
2 const int MOTOR_L_CWCCW = 8;
3 const int MOTOR_L_PWM = 10;
4
5 void setup() {
6   /* モーター回転方向 */
7   /* 後退: HIGH */
8   /* 前進: LOW */
9   pinMode(MOTOR_L_CWCCW, OUTPUT); // 右回転方向
10 }
11
12 void loop() {
13   digitalWrite(MOTOR_L_CWCCW, LOW); // 右回転方向
14   analogWrite(MOTOR_L_PWM, 100); // 右PWM
15   delay(1000);
16
17   digitalWrite(MOTOR_L_CWCCW, HIGH); // 右回転方向
18   analogWrite(MOTOR_L_PWM, 100); // 右PWM
19   delay(1000);
20 }
```

練習 3. Example102 でほぼ直角に向きを変えるにはどうしたら良いか。

直進をして、直角に曲がる。これらの動作ができれば、繰り返すことで、四角形をつくることができる。

練習 4. うまいかない点がある。どんな点か述べよ。

Example103 前進後退を繰り返す

```
Example0103
1 // 01: motor
2 void setup() {
3   /* モーター回転方向 */
4   /* 後退: HIGH */
5   /* 前進: LOW */
6   pinMode(7, OUTPUT); // 右回転方向
7   pinMode(8, OUTPUT); // 左回転方向
8 }
9
10 void loop() {
11   digitalWrite(7, LOW); // 右回転方向 前
12   digitalWrite(8, LOW); // 左回転方向 前
13   analogWrite(9, 200); // 右PWM
14   analogWrite(10, 200); // 左PWM
15   delay(1000);
16
17   digitalWrite(7, HIGH); // 右回転方向 後
18   digitalWrite(8, HIGH); // 左回転方向 後
19   analogWrite(9, 200); // 右PWM
20   analogWrite(10, 200); // 左PWM
21   delay(1000);
22 }
```

練習 5. 左右の違いを補正するにはどうするか。速度を左右で少し変える？

1000m 秒進んで、直角に曲がり、また 1000m 秒進むようにした、どうすればよいか

練習 6. 真っすぐに進むようにするには？ 直角に曲がるには？

実際に上のような動作を行うには、ケーブルがつながっていると自由に移動できないので、後の講座で学ぶ、無線を利用する方法のときに実行してみたい。

サブプログラムを用いるスケッチ 参考

前進後退 プログラムが分かりやすくなる

Example104

Example0104

```
1 // 01: motor
2 const int MOTOR_R_CWCCW = 7;
3 const int MOTOR_L_CWCCW = 8;
4 const int MOTOR_R_PWM = 9;
5 const int MOTOR_L_PWM = 10;
6
7 void motor_forward(int left, int right) {
8   digitalWrite(MOTOR_L_CWCCW, LOW); // 左回転方向
9   digitalWrite(MOTOR_R_CWCCW, LOW); // 右回転方向
10  analogWrite(MOTOR_L_PWM, right); // 左PWM
11  analogWrite(MOTOR_R_PWM, left); // 右PWM
12 }
13
14 void motor_backward(int left, int right) {
15   digitalWrite(MOTOR_L_CWCCW, HIGH); // 左回転方向
16   digitalWrite(MOTOR_R_CWCCW, HIGH); // 右回転方向
17   analogWrite(MOTOR_L_PWM, right); // 左PWM
18   analogWrite(MOTOR_R_PWM, left); // 右PWM
19 }
20
21 void setup() {
22   /* モーター回転方向 */
23   /* 後退: HIGH */
24   /* 前進: LOW */
25   pinMode(MOTOR_L_CWCCW, OUTPUT); // 左回転方向
26   pinMode(MOTOR_R_CWCCW, OUTPUT); // 右回転方向
27 }
28
29 void loop() {
30   motor_forward(100, 100); // 前進
31   delay(1000);
32   motor_backward(100, 100); // 後退
33   delay(1000);
34 }
```

左速度と右速度を指定する

サブプログラム モーター前進

左速度と右速度を指定する

サブプログラム モーター後退

左速度と右速度を指定する

主プログラム

初めのセットアップ

前進と後退を繰り返す

前進後退 別のプログラム

Example104a

Example0104a

```
1 // 01: motor
2 const int MOTOR_R_CWCCW = 7;
3 const int MOTOR_L_CWCCW = 8;
4 const int MOTOR_R_PWM = 9;
5 const int MOTOR_L_PWM = 10;
6
7 void motor(int left, int right, int left_c, int right_c) {
8     if (left_c == 1) {
9         digitalWrite(MOTOR_L_CWCCW, LOW); // 左回転方向
10    } else {
11        digitalWrite(MOTOR_L_CWCCW, HIGH); // 左回転方向
12    }
13
14    if (right_c == 1) {
15        digitalWrite(MOTOR_R_CWCCW, LOW); // 右回転方向
16    } else {
17        digitalWrite(MOTOR_R_CWCCW, HIGH); // 右回転方向
18    }
19
20    analogWrite(MOTOR_L_PWM, right); // 左PWM
21    analogWrite(MOTOR_R_PWM, left); // 右PWM
22 }
23
24 void setup() {
25     /* モーター回転方向 */
26     /* 後退: HIGH */
27     /* 前進: LOW */
28     pinMode(MOTOR_L_CWCCW, OUTPUT); // 右回転方向
29     pinMode(MOTOR_R_CWCCW, OUTPUT); // 右回転方向
30 }
31
32 void loop() {
33     motor(100, 100, 1, 1);
34     delay(1000);
35     motor(100, 100, 0, 0);
36     delay(1000);
37 }
38
```

left_c が 1 なら左車輪前進にセット

right_c が 1 なら右車輪前進にセット

左速度 100、右速度 100、左車輪前進、右車輪前進にセット
1000 ミリ秒維持

左速度 100、右速度 100、左車輪後退、右車輪後退にセット
1000 ミリ秒維持

構文) もし～なら～a～そうでなければ～b～ の構文

```
if (left_c ==1 ) { a } else { b }
```

もし left_c が1 ならば a の部分の命令を実行、そうでなければ b の部分の命令を実行する。

if の後の条件により、実行される命令が異なる。