

1. 実験状況

日付：1月10日

天気：晴れ

2. 目的

C言語を用いて Arduino の理解を深める。

3. 使用機器

表 1. 使用機器

機材名	型番	数量
ブレッドボード		1
マブチモーター	FA-130RA	1
パワー MOS FET	2SK4017	1
ダイオード	1N4007	2
抵抗	22 Ω 、0.1 Ω 5W	各 1
LED	-	1
基板用タクトスイッチ	-	1
3V 電池ボックス	-	1
Arduino	-	1

4. 実験内容と結果

4.1 目的物

次の回路の実現

① 停止状態

LED が 5 秒毎に点滅する

② 高速状態

スイッチを押すと、LED は 0.5 秒毎に点滅する。モータは高速に回転する。

③ 低速状態

5 秒後に、LED は 2 秒毎に点滅するようになり、モータは低速に回転する。

さらにスイッチを押すと、停止状態になる。

4.2 回路図

作成した回路図を図 1 に示す。

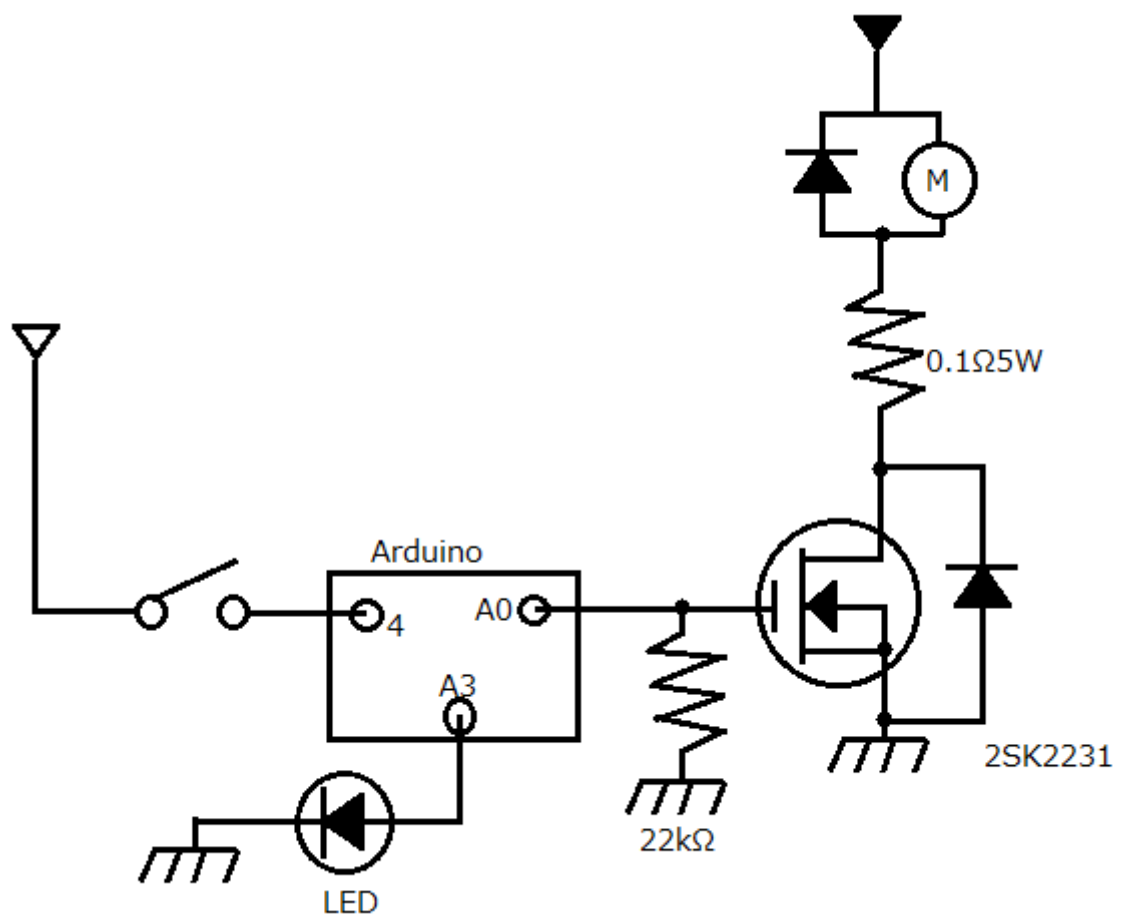


図 1. 回路図

4.3 プログラム

今回のテスト課題は C 言語で行ったため、動作確認した C 言語のプログラムを以下に記す。

```
#include <MsTimer2.h>
#include <avr/sleep.h>
const byte led = A3;
const byte mot = A0;
const byte button = 4;
static byte a;
static byte b;

void setup() {
    a = 0;
    b = 0;
    pinMode(led, OUTPUT);
    digitalWrite(led, LOW);
    pinMode(mot, OUTPUT);
    digitalWrite(mot, LOW);
    pinMode(button, INPUT_PULLUP);
    MsTimer2::set(500, vect);
    MsTimer2::start();
}

void vect(){
    b = b + 1;
    if(a == 0 && b%10 == 0){
        right();
    }
    else if(a == 1){
        right();
        if(b == 10){
            a = 2;
        }
    }
}
```

```

    else if(a == 2 && b%4 == 0){
        right();
    }
}

```

```

void right(){
    digitalWrite(led, HIGH);
    delay(100);
    digitalWrite(led, LOW);
}

```

```

void loop() {
    static int i = 2;
    static bool prev = HIGH;
    static bool now = HIGH;

    now = digitalRead(button);
    if(now == HIGH && prev == LOW){
        if(a == 0){
            a = 1;
            b = 0;
            delay(20);
        }
        else if(a == 2){
            a = 0;
            delay(20);
            digitalWrite(mot, LOW);
        }
    }

    prev = now;

    if(a == 0);
    else if(a == 1){
        digitalWrite(mot, HIGH);
    }
    else if(a == 2){

```

```
if(i == 0){  
    digitalWrite(mot, LOW);  
    i = 2;  
}  
else digitalWrite(mot, HIGH);  
}  
if(i == 0){  
    i = 2;  
}  
i = i-1;  
}
```

4.4 プログラム説明

今回は led を光らせるためにポート A3、モータを回すためにポート A0、ボタンを動作させるためにポート 4 を使用した。停止状態、高速回転、低速回転の状態の区別をつけるためにグローバル変数 a、それぞれの状態において条件通りに動作するようにするためグローバル変数 b を作成した。

setup()ではグローバル変数にそれぞれ 0 を代入する。これは停止状態を意味する。割り込みを使用し 0.5 秒毎に関数 vect()を呼び出す。

vect()では呼び出させるたびに変数 b に 1 を足す。a=0 のとき 10 回中 1 回だけ right()を呼び出す。つまり 5 秒に 1 回 right()を呼び出す。a=1 のとき毎回 right()を呼び出す。つまり 0.5 秒に 1 回 right()を呼び出す。そして 10 回呼び出されたとき、つまり 5 秒経過したときに変数 a に 2 が代入されて低速回転へと移行する。a=2 のとき 4 回に 1 回 right()を呼び出す。つまり 2 秒に 1 回 right()を呼び出す。よって vect()で LED の点滅の周期を変化させる。

right()は LED の点滅を示している。

loop()ではスイッチの動作とモータの回転速度の管理を行っている。変数 i はモータの回転速度の変化をつけるための変数であり loop()が呼び出されるたびに 1 を引かれ、i=0 になったら再び 2 に戻る。1 つ目の if 文ではスイッチの動作を行う。a=0 のとき、つまり停止状態のときにスイッチを押すと変数 a に 1 が代入される。これで停止状態から高速回転へ移行。a=2 のとき、つまり低速回転のときにスイッチを押すと変数 a に 0 が代入される。これで低速回転から停止状態へ移行。2 つ目の if 文ではモータの回転速度の管理を行う。a=1 のときは常に HIGH となり高速回転となる、a=2 のときは i=0 のとき LOW となり i=1,2 のときは HIGH となる。Duty 比によりこれで低速回転を実現する。

5. 考察

今回の実験により `delay` では実現不可能なことも割り込みを使用することにより複数のプログラム操作が可能であることが理解できた。これより割り込みに関する重要性和理解が深まった。しかし作成したプログラムでは 0.5 秒毎に関数が呼び出されるため、必要のないタイミングでも関数が呼び出されてしまう。これでは無駄が生じてしまう。割り込みのタイミングを変化させることができればこの無駄が省けると考えられる。