

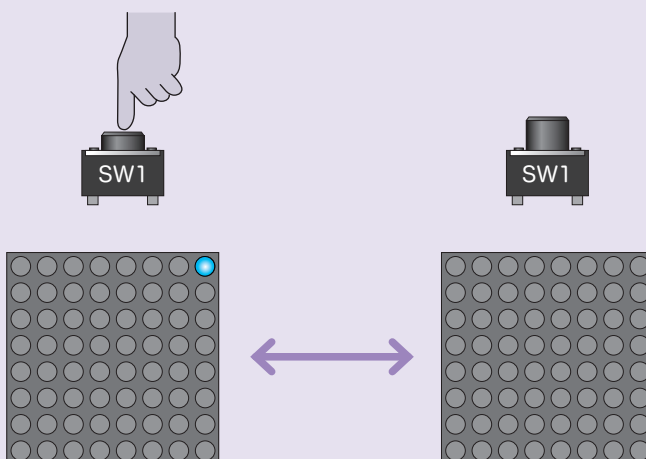
## SW を使ってドットマトリクス LED を点灯させる

このSTEPでは、入力装置と出力装置を使ったプログラムを組んでいきます。  
入力装置と出力装置の扱い方の違いをしっかりと抑えながら学習を進めてください。

SWの構造や接続するポートについては前STEPで学習しました。  
では、実際にSWのON/OFFによるDRの変化をどのようにプログラムで読み取るのか、  
このSTEPで学習していきましょう。

### 課題 9-1

SW1 を押し続けている間だけドットマトリクス LED が点灯する。



フローチャートを描きましょう。

※ 今回は、入力装置を初めて使うので初期化関数の中身を示しておきます。

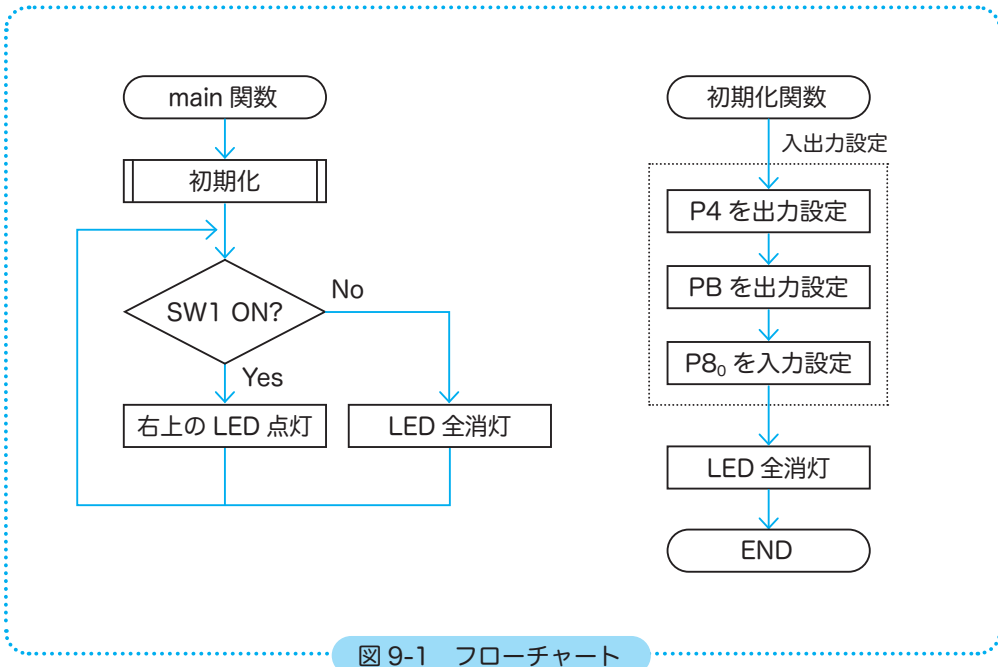


図 9-1 フローチャート

初期化関数では、ドットマトリクス LED を出力に設定するだけでなく SW を入力に設定することを忘れないようにしましょう。

さて、図 9-1 のフローチャートですが、SW1 が ON の時、OFF の時にそれぞれ違う処理を行うこととなります。この形は条件分岐でしたね。STEP02 では、for 文を条件分岐として紹介しましたが、汎用性のある条件分岐は if 文を使います。

## 【if文】

if 文には以下のような型があります。用途によって使い分けましょう。

```
if(条件式)
```

```
{  
    処理  
}
```

条件式が真なら処理を実行する。  
条件式が偽なら処理は実行されない。

```
if(条件式)
```

```
{  
    処理 1  
}  
else  
{  
    処理 2  
}
```

条件式が真なら処理 1 を実行する。

条件式が偽なら処理 2 を実行する。

```
if(条件式 A)
```

```
{  
    処理 1  
}  
else if(条件式 B)  
{  
    処理 2  
}  
else  
{  
    処理 3  
}
```

条件式 A が真なら処理 1 を実行する。

条件式 A が偽で条件式 B が真なら処理 2 を実行する。

条件式 A も条件式 B も偽なら処理 3 を実行する。

条件式は以下のような演算子を用いて記述します。用途によって使い分けましょう。

| 演算子 | 記述例    | 意味                       |
|-----|--------|--------------------------|
| ==  | a == b | a と b の値が等しければ真、それ以外は偽   |
| !=  | a != b | a と b の値が等しくなければ真、それ以外は偽 |
| <   | a < b  | a が b より小さければ真、それ以外は偽    |
| >   | a > b  | a が b より大きければ真、それ以外は偽    |
| <=  | a <= b | a が b 以下であれば真、それ以外は偽     |
| >=  | a >= b | a が b 以上であれば真、それ以外は偽     |

### 問題 9-1

SW1 が ON の時に LED を点灯させるには、条件式に何を書き込めばいいでしょうか？

下記の四角に当てはまる条件式を考えましょう。

ヒントは、「SW はアクティブ L で接続している」「SW1 は P8<sub>0</sub> に接続している」ことです。

```
if (  )
{
    P4.DR.BYTE = 0x80; // LED 点灯
}
else
{
    P4.DR.BYTE = 0x00; // LED 消灯
}
```

答えは p.107 参照

## プログラム例 9-1

```
05 #include <3052f.h> // 3052F 固有の定数
06
07 /*
08  * 初期化関数
09  */
10 void initIO(void)
11 {
12     // 入出力設定
13     P4.DDR = 0xFF; // 出力 LED 横行
14     PB.DDR = 0xFF; // 出力 LED 縦行
15     P8.DDR = 0xFC; // 入力 SW1,2
16
17     // 出力レベル設定
18     P4.DR.BYTE = 0x00; // L アノード
19     PB.DR.BYTE = 0xFF; // H カソード
20 }
21
22 /*
23  * main 関数
24  */
25 int main(void)
26 {
27     initIO(); // 初期化関数の呼び出し
28
29     PB.DR.BYTE = 0xFE; // 1111 1110 カソード
30
31     // 永続ループ
32     while (1)
33     {
34         if (P8.DR.BIT.B0 == 0) // SW1 が ON なら
35             P4.DR.BYTE = 0x80; // 右上の LED だけ点灯
36         else // その他の場合
37             P4.DR.BYTE = 0x00; // 消灯
38     }
39
40     return 0;
41 }
```