

Report on the Experiment

No. 1

Subject ワンチップマイコン基礎実験I

Date 2020. 12. 09

Weather 晴れ Temp 24 °C Wet 65.5 %

Class E4
Group
Chief
Partner

No 14
Name 小畠 一泰

Kure National College of Technology

1 目的

PIC16F84A についてプログラム方法と基本的な入出力制御の習得を目的とする.

2 課題

下記ソースコードを共通部分とし, すべてのソースコードは下記を含むものとする.

コード 1 共通部分

```
1  list p=16F84A
2  #include <p16F84A.inc>
3  __CONFIG _CP_OFF & _WDT_OFF & _PWRTE_ON & _HS_OSC
4
5  ORG 0x000
6
7  bsf STATUS, RP0 ; バンク 0 → 1 に切り替え
8  clrf TRISB      ; TRISB をすべて 0, 全ビット出力設定に
9  bcf STATUS, RP0 ; バンク 1 → 0 に切り替え
10
11 clrf PORTB      ; すべて消灯
```

また下記を標準タイマーとし, 明示しない限りこれらも含み使用できるものとする.

コード 2 タイマー

```
1  COUNTER1 EQU 0x0C ; 作業用変数の設定
2  COUNTER2 EQU 0x0D ; 汎用ファイルレジスタ 0Ch から
3
4  WAIT clrf COUNTER1 ; 変数 1 をクリア
5
6  WAIT1
7  clrf COUNTER2      ; 変数 2 をクリア
8  call WAIT2         ; さらにサブルーチンへ
9  decfsz COUNTER1, 1 ; 変数から 1 を引く
10 goto WAIT1         ; 結果が 0 でなければ実行
11 return             ; 結果が 0 ならばスキップしこの行を実行
12
13 WAIT2
14 decfsz COUNTER2, 1 ; サブルーチンのサブルーチン
15 goto WAIT2
16 return
```

コード 3 [課題 1] RB1 に接続された LED を点灯させるプログラムを作成せよ

```
1    bsf PORTB, 1    ;0 番目の LED 点灯
2    STP goto STP    ;無限ループ
```

コード 4 [課題 2] List3 の点滅周期を約 1 秒になるように調整したプログラムを作成せよ

```
1  TODO
```

コード 5 [課題 3] List3 の点滅を 10 回に規定して動作させるプログラムを作成せよ

```
1  TODO
```

コード 6 [課題 4] RB0 から RB7 まで順次点灯するように List4 を書き換えよ

```
1  MAIN
2      movlw B'00000001' ;Wレジスタに B'00000000' を代入
3      movwf PORTB      ;PORTB に値を転送→ 0 番目の LED 点灯
4      call WAIT        ; 時間待ちサブルーチン呼び出し
5
6      movlw B'00000011'
7      movwf PORTB
8      call WAIT
9
10     movlw B'00000111'
11     movwf PORTB
12     call WAIT
13
14     movlw B'00001111'
15     movwf PORTB
16     call WAIT
17
18     movlw B'00011111'
19     movwf PORTB
20     call WAIT
21
22     movlw B'00111111'
23     movwf PORTB
24     call WAIT
25
26     movlw B'01111111'
27     movwf PORTB
28     call WAIT
29
30     movlw B'11111111'
31     movwf PORTB
32     call WAIT
33
34     goto MAIN          ; 無限ループ
```

コード 7 [課題 5] 課題 4 をローテートを用いる方法に書き換えよ

```
1  MAIN
2      movlw B'00000001' ;Wレジスタに B'00000000' を代入
3      movwf PORTB      ;PORTB に値を転送→ 0 番目の LED 点灯
4      call WAIT        ; 時間待ちサブルーチン呼び出し
5      call SHIFT
6      call SHIFT
7      call SHIFT
8      call SHIFT
9      call SHIFT
10     call SHIFT
11     call SHIFT
12     goto MAIN        ; 無限ループ
13
14  SHIFT
15     rlf PORTB        ; 左に 1 つシフト
16     addwf PORTB, 1    ; インクリメントし PORTB に値を転送
17     call WAIT
18     return
```

コード 8 [課題 6] LED を上位・下位半分ずつ交互に点滅するプログラムを作成せよ

```
1  ;init
2      movlw B'00001111' ;Wレジスタに B'00000000' を代入
3      movwf PORTB      ;PORTB に値を転送→ 0 番目の LED 点灯
4      call WAIT        ; 時間待ちサブルーチン呼び出し
5      call WAIT
6      call WAIT
7
8  MAIN
9      swapf PORTB, 1    ;PORTB の上位ビットと下位ビットを交換
10     call WAIT
11     call WAIT
12     call WAIT
13     goto MAIN        ; 無限ループ
```

コード 9 [課題 7] List5 をポート A の押されたスイッチに対応して, LED を消灯するように変更せよ

```
1
2 movlw B'00011111' ; 下位 5 ビットを 1 に
3 movwf TRISA ; TRISA に代入, RA0 から RA4 が入力設定に
4 movlw B'00000000' ; Wレジスタにすべて 0 に
5 movwf TRISB ; TRISB をすべて 0, 出力設定に
6 bcf STATUS, RPO ; バンク 1 → 0 に切り替え
7
8 clrw ; Wレジスタをクリア
9
10 MAIN
11 bcf STATUS,C ; Cフラグクリア
12 movf PORTA, 0 ; 入力データを Wレジスタに転送
13 comf PORTA, 0 ; PORTA の値を反転させる
14 movwf PORTB ; ポート B に出力
15 goto MAIN
```

コード 10 [課題 8] List5 を対応して点灯する LED を RB1 から RB5 となるように変更せよ

```
1
2 movlw B'00011111' ; 下位 5 ビットを 1 に
3 movwf TRISA ; TRISA に代入, RA0 から RA4 が入力設定に
4 movlw B'00000000' ; Wレジスタにすべて 0 に
5 movwf TRISB ; TRISB をすべて 0, 出力設定に
6 bcf STATUS, RPO ; バンク 1 → 0 に切り替え
7
8 clrw ; Wレジスタをクリア
9
10 MAIN
11 bcf STATUS,C ; Cフラグクリア
12 movf PORTA, 0 ; 入力データを Wレジスタに転送
13 rlf PORTA, 0 ; 左にシフト
14 movwf PORTB ; ポート B に出力
15 goto MAIN
```

3 調査課題

- 今回使用しているセラミック発振子とクリスタル発振子の長所, 短所を調べよ。
 - セラミック発振子
 - 発振周波数の安定性が高い (水晶振動子と LC もしくは CR 発振回路との中間に位置する)
 - 小型, 軽量 (通常的水晶振動子の約 1/2 以下のサイズ)
 - 発振回路の無調整化, 低価格化が可能
 - セラミック発振子は CR, LC など電氣的共振を応用したものと異なり, 機械的共振を利用
 - 外部回路や電源電圧の変動などの影響を受けにくい
 - 無調整で高安定な発振回路が得られる
 - クリスタル (水晶) 発振子
 - 周波数誤差が少ない (セラミック発振子よりも安定している)
 - Q 値が大きい
 - 均一で安定した高品質材料を確保できる
 - 温度特性も良好で, 弾性率の温度変化と熱膨張率が相殺する関係にあるので, 温度変化に対して特性変動が小さい
 - ウェハーを切り出す方向次第で, さまざまな発振周波数を生み出せる
 - 発振させたときの内部損失が極めて小さい
- マイコンの入力端子には, 多くの場合プルアップ抵抗もしくはプルダウン抵抗が付けられる. 今回 RA0~RA4 はプルダウン, MCLR はプルアップである. 抵抗値の選択を行う場合留意する点について調べよ.

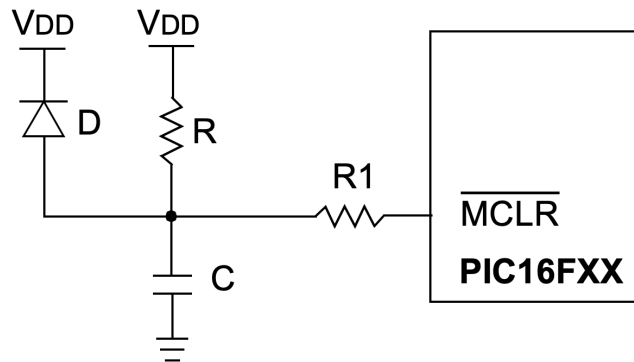


図 1: MCLR

たとえば 図 1 のように MCLR を利用してリセットを作動させるためには次の条件を満たす必要がある。

- 抵抗 R: 抵抗での電圧降下が大きくなり過ぎて, 論理の判定が出来なくなってしまうように 40 [kΩ] 以下であること
- ダイオード D: 電源が off になった時にリセットが早めにかかるように, 急速にコンデンサ C を放電させる必要があるため容量は大きすぎないこと
- 抵抗 R1: MCLR 端子にコンデンサからの過電流が流れるのを防止するためのもので, 100~1 [kΩ] 程度とすること

このようにプルダウン, プルアップ抵抗には電圧を安定化させるための役割があるが, 入力として使う場合には閾値を上回ったり, 逆に下回ってしまわないように適切な電圧値, 電流値を得る必要がある. したがってマイコンや IC のデータシートなどを読み抵抗値を決定しなければならない.

- マイコンの出力端子では, 電流をプルダウン抵抗に出力するソース出力と電流をプルアップ抵抗などから吸い込むシンク出力の両方もしくはどちらか一方が可能となっている. PIC16F84A の端子内部の構造を調べ, 各端子

を出力設定とした場合にどのような出力が可能か調べよ (トータムポール型とオープンドレイン型について理解すること).

- シンク (吸い込み) ロジック: NPN トランジスタ (主に国内での使用が多い)
- ソース (吐き出し) ロジック: PNP トランジスタ (主に海外での使用が多い)

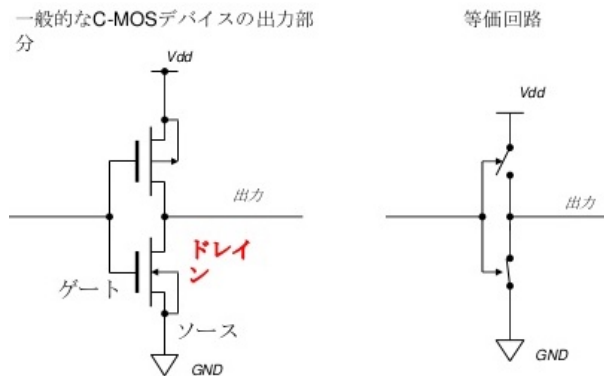


図 2: トータムポール型 (<https://www.slideshare.net/minorunatsutani/kdmsopen-drain> より抜粋)

- トータムポール型
 - VCC と GND 間に 2 つのトランジスタを組み合わせた出力回路
 - 出力にプルアップ抵抗をつけなくてもハイまたはローレベルが確定する
 - シンクとソースの両方が可能

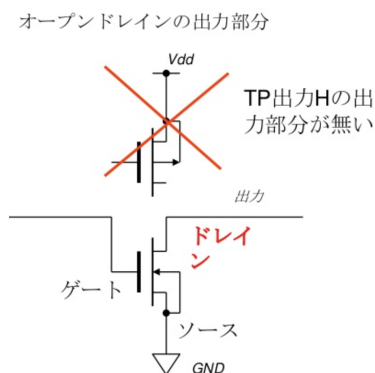


図 3: オープンドレイン型 (<https://www.slideshare.net/minorunatsutani/kdmsopen-drain> より抜粋)

- オープンドレイン型
 - 出力はローの出力しかできない
 - 信号線をハイにするためにはプルアップ抵抗が必要
 - シンクのみ

PIC16F84A ではデータシートを見ると RA, RB ピンは次の表のようになっている。

表 1: I/O Pin's Buffer Types

Pin Name	Buffer Type	Function
RA0	TTL	
RA1	TTL	
RA2	TTL	

Pin Name	Buffer Type	Function
RA3	TTL	
RA4	ST	Output is open drain type.
RB0	TTL	Internal software programmable weak pull-up.
RB1	TTL	Internal software programmable weak pull-up.
RB2	TTL	Internal software programmable weak pull-up.
RB3	TTL	Internal software programmable weak pull-up.
RB4	TTL	Internal software programmable weak pull-up.
RB5	TTL	Internal software programmable weak pull-up.
RB6	TTL	Internal software programmable weak pull-up.
RB7	TTL	Internal software programmable weak pull-up.

表 2: I/O Pin's Threshold

Symbol	Characteristic	Min	Max	Units
V_{IL}	with TTL buffer	VSS	0.8	V
V_{IL}	with Schmitt Trigger buffer	VSS	0.2VDD	V
V_{IH}	with TTL buffer	2.0	VDD	V
V_{IH}	with Schmitt Trigger buffer	0.8VDD	VDD	V

4. List3 の WAIT ルーチンにおいて, 変数をクリアしてから減算処理に入っている. この理由を説明せよ.
1. クリアすることで B'00000000' となる.
 2. 減算をすると B'11111111'(D'255') となり, 最大値が得られる.
 3. 最大値を COUNTER1, COUNTER2 で得られたので 255×255 の回数分得られる.

4 参考文献

- PIC 命令一覧表 (<https://www.mlab.im.dendai.ac.jp/~assist/PIC/appendix/instruction/>)
- セラミック発振子 (セラロック) | 村田製作所 (<https://www.murata.com/ja-jp/products/timingdevice/ceralock/basic/summary>)
- 水晶振動子 | マルツオンライン (<https://www.marutsu.co.jp/contents/shop/marutsu/mame/46.html>)
- 第2回 水晶を発振器に使う5つの理由 - EE Times Japan (<https://eetimes.jp/ee/articles/1011/24/news121.html>)
- PIC16F84A Data Sheet (<https://akizukidenshi.com/download/PIC16F84A.pdf>)
- リセットの使い方 | 電子工作の実験室 (<http://www.picfun.com/pic21.html>)
- オープンコレクター出力型とトータムボール出力型では何が違うのですか? | 東芝デバイス&ストレージ株式会社 (<https://toshiba.semicon-storage.com/jp/semiconductor/knowledge/faq/opto/opto-038.html>)
- Kdms 勉強会 (open drain) (<https://www.slideshare.net/minorunatsutani/kdmsopen-drain>)
- RA4 の罟 | PIC の罟 (<http://mhage.zombie.jp/PicTrap/RA4.html>)
- TTL or ST? What is it? | PICBASIC Community Discussion (<http://www.picbasic.co.uk/forum/showthread.php?t=8704>)
- what is TTL input and Schmitt Trigger input? | microchip Forums (<https://www.microchip.com/forums/m103724.aspx>)