

マイクロコンピュータ 後期期末レポート

電気情報工学科 2 年

E1533 西総一郎

2017 年 2 月 10 日提出

- 光が流れるプログラム (片道バージョン)
- 光が流れるプログラム (往復バージョン)
- パルスモータ 1-2 相励磁プログラム
- 電子サイコロのプログラム
- タイマ割り込み制御プログラム

目 次

1	リスト 5-5(光が流れるプログラム (片道バージョン))	3
1.1	プログラム説明	3
1.2	フローチャート	3
1.3	ソースコード	4
1.4	実行結果・考察	5
2	リスト 5-6(光が流れるプログラム (往復バージョン))	7
2.1	プログラム説明	7
2.2	フローチャート	7
2.3	ソースコード	8
2.4	実行結果・考察	9
3	リスト 5-12(パルスモータ 1-2 相励磁プログラム)	11
3.1	プログラム説明	11
3.2	フローチャート	11
3.3	ソースコード	11
3.4	実行結果・考察	14

1 リスト 5-5(光が流れるプログラム (片道バージョン))

1.1 プログラム説明

8 個ある LED の内 1 個を左端から右へ 0.5 秒間隔で順次点灯することによって、光が流れるように見えるプログラム。

1.2 フローチャート

このプログラムのフローチャートを図 1 に示す。

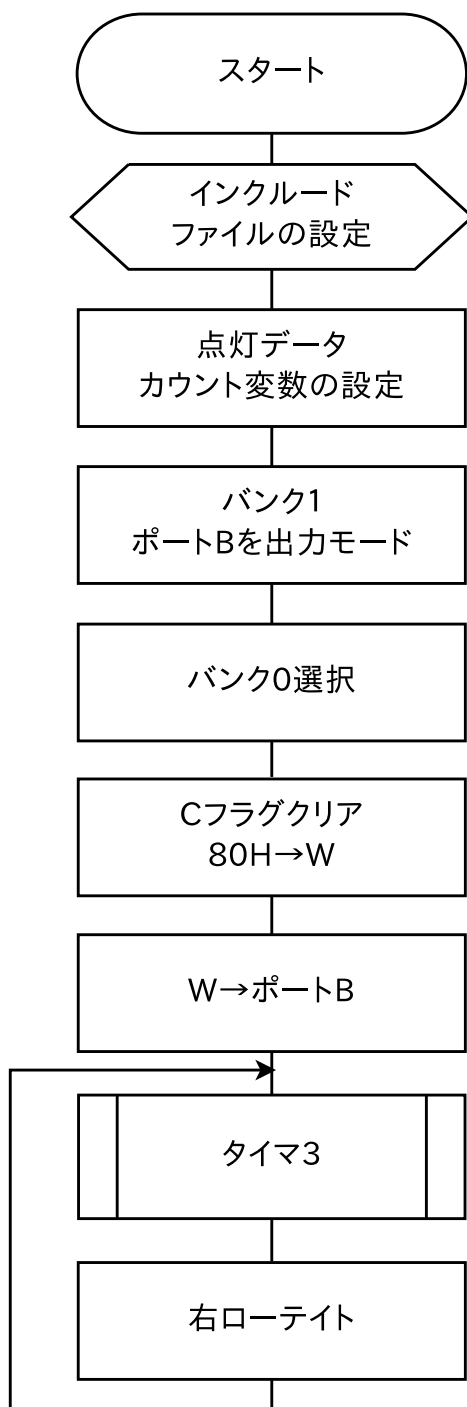


図 1: 光が流れるプログラム (片道バージョン) のフローチャート

1.3 ソースコード

```
1 ; *****
2 ; リスト5.5
3 ; 光が流れるプログラム（片道バージョン）
4 ; （右方向）
5 ; *****
6 LIST P=P16F84A ;使用するPICを指定
7 INCLUDE "P16F84A.INC" ;読み込む設定ファイルを指定
8 ; *****
9 LEDD EQU 80H ;LEDの点灯データの設定
10 CNT1 EQU 0CH ;タイマ1用のカウンタ変数
11 CNT2 EQU 0DH ;タイマ2用のカウンタ変数
12 CNT3 EQU 0EH ;タイマ3用のカウンタ変数
13 ; *****
14 ORG 0 ;プログラムを格納する先頭アドレス
15
16 BSF STATUS,RP0 ;バンク1を選択
17 CLRF TRISB ;ポートBをすべて出力モードに設定
18 BCF STATUS,RP0 ;バンク0を選択
19 BCF STATUS,C ;Cフラグをクリア
20
21 MOVLW LEDD ;点灯データをWレジスタにセット
22 MOVWF PORTB ;点灯データをポートBに出力（LEDが点灯）
23 REPEAT CALL TIMER3 ;0.5秒タイマの呼び出し
24 RRF PORTB,1 ;ポートBを1ビット右にローテイト
25 GOTO REPEAT
26
27 ;止めるまで永遠に繰り返す
28
29 TIMER1 MOVLW D'62' ;0.1ミリ秒タイマサブルーチン
30 MOVWF CNT1
31 LOOP1 NOP
32 DECFSZ CNT1,F
33 GOTO LOOP1
34 RETURN
35
36 TIMER2 MOVLW D'100' ;10ミリ秒タイマサブルーチン
37 MOVLW CNT2
38 LOOP2 NOP
39 CALL TIMER1
40 DECFSZ CNT2,F
41 GOTO LOOP2
42 RETURN
43
44 TIMER3 MOVLW D'50' ;0.5秒タイマサブルーチン
45 MOVWF CNT3
46 LOOP3 NOP
47 CALL TIMER2
48 DECFSZ CNT3,F
49 GOTO LOOP3
50 RETURN
51
52 END ;プログラムの終わり
```

1.4 実行結果・考察

状態	7 6 5 4 3 2 1 0 C
1	$80_{16} = 10000000_2$
2	$40_{16} = 01000000_2$
3	$20_{16} = 00100000_2$
4	$10_{16} = 00010000_2$
5	$08_{16} = 00001000_2$
6	$04_{16} = 00000100_2$
7	$02_{16} = 00000010_2$
8	$01_{16} = 00000001_2$
9	$00_{16} = 00000000_2$

:LED 点灯 :LED 消灯, :C フラグ 1 :C フラグ 0

図 2: 光が流れるプログラム (片道バージョン) の実行結果。上の数字はポート B の値を表す。

図 2 はポート B の出力状態を表している。0.5 秒毎に LED の点灯箇所が右に移動していき (状態 1 ~ 9)、右端に到達する (状態 9) と全ての LED が消灯する。この一連の動作を繰り返す。ローテイト (RRF) 命令は C フラグを経由してデータを 1 ビット右にシフトするので C フラグにデータが存在するときには全ての LED が消灯している。

このプログラムのタイマについて考えてみる。PIC のクロック周波数は 10MHz なので、1 クロックは

$$\frac{1}{10\text{MHz}} = 0.1 \mu\text{s}$$

4 クロックで 1 サイクルなので、1 サイクルは

$$0.1 \mu\text{s} \times 4 = 0.4 \mu\text{s}$$

TIMER1 のサイクル数は図 3 より、249 サイクルだとわかり、

$$0.4 \mu\text{s} \times 249 = 99.6 \mu\text{s} \approx 0.1 \text{ms}$$

TIMER1 では 0.1 ms かかる。

この TIMER1 を TIMER2 では 100 回呼び出し、TIMER3 は TIMER2 を 50 回呼び出しているので、

$$0.1 \text{ms} \times 100 \times 50 = 50 \text{ms} = 0.5 \text{s}$$

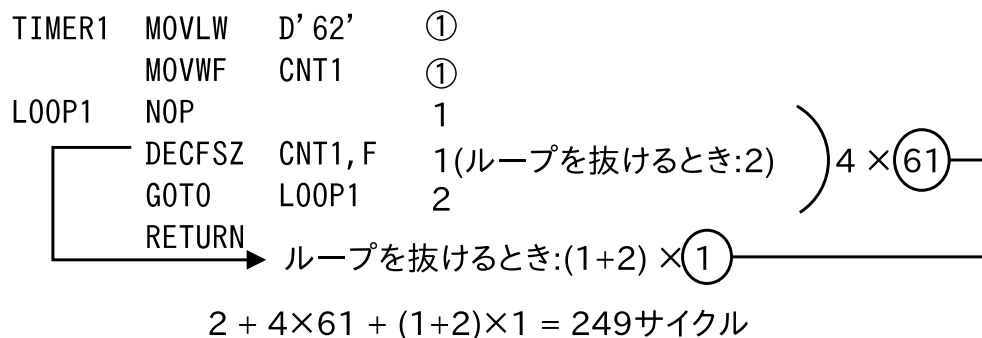


図 3: サイクル数の考え方

よって TIMER1, TIMER2, TIMER3 の合計で 0.5s のタイマールーチンであることがわかる。

次に光の移動方向について考えてみる。このプログラムにおいて、光の移動方向を決めているのは

9 行 目	LEDD	EQU	80H	; LED の点灯データの設定
24 行 目	RRF	PORTB, 1		; ポート B を 1 ビット右にローテイト

この 2 つなので次のように変更すれば左方向に移動するようになる。

9 行 目	LEDD	EQU	01H	; LED の点灯データの設定
24 行 目	RLF	PORTB, 1		; ポート B を 1 ビット右にローテイト

RLF 命令も RRF 命令と同様に、C フラグを経由してデータを 1 ビット左にシフトするので C フラグにデータが存在するときには全ての LED が消灯している。

LED の移動時間 S を調整するには TIMER1 の 0.1 ミリ秒を基準にして、

$$S = 0.1 \text{ ms} \times \text{TIMER2} \times \text{TIMER3}$$

$$\text{TIMER2} \leq 255, \text{TIMER3} \leq 255$$

TIMER2 と TIMER3 とで移動時間 S を調整する。

2 リスト 5-6(光が流れるプログラム (往復バージョン))

2.1 プログラム説明

8 個ある LED の内 1 個を左端から右へ 0.5 秒間隔で順次点灯していき、右端に到達すると移動方向を逆にして左端に向かう。光が左から右へ流れるように見え、右端に到達すると右から左へ流れるように見えるプログラム。

2.2 フローチャート

このプログラムのフローチャートを図 4 に示す。

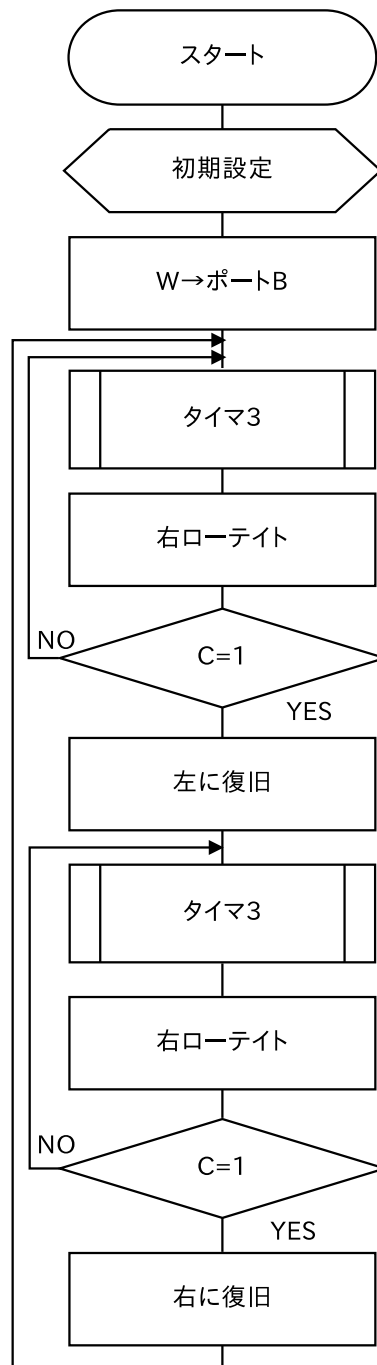


図 4: 光が流れるプログラム (往復バージョン) のフローチャート

2.3 ソースコード

```

1 ; *****
2 ; リスト5.6
3 ; 光が流れるプログラム（往復バージョン）
4 ; （Cフラグで判定バージョン）
5 ; *****
6     LIST      P=P16F84A      ;使用するPICを指定
7     INCLUDE   "P16F84A.INC"  ;読み込む設定ファイルを指定
8 ; *****
9 LEDD      EQU      80H      ;LEDの点灯データの設定
10 CNT1      EQU      0CH      ;タイマ1用のカウンタ変数
11 CNT2      EQU      0DH      ;タイマ2用のカウンタ変数
12 CNT3      EQU      0EH      ;タイマ3用のカウンタ変数
13 ; *****
14     ORG       0              ;プログラムを格納する先頭アドレス
15
16     BSF       STATUS,RP0     ;バンク1を選択
17     CLRF      TRISB          ;ポートBをすべて出力モードに設定
18     BCF       STATUS,RP0     ;バンク0を選択
19     BCF       STATUS,C       ;Cフラグをクリア
20     MOVLW     LEDD           ;点灯データをWレジスタにセット
21     MOVWF     PORTB          ;点灯データをポートBに出力（LEDが点灯）
22
23 RIGHT     CALL    TIMER3      ;0.2秒タイマの呼び出し
24           RRF      PORTB,1    ;ポートBを1ビット右にローテイト
25           BTFSS   STATUS,C    ;Cフラグが1なら次の命令をスキップ
26           GOTO    RIGHT
27
28           RLF      PORTB,1    ;過分ローテイトの復旧
29           RLF      PORTB,1    ;過分ローテイトの復旧
30
31 LEFT      CALL    TIMER3      ;0.2秒タイマの呼び出し
32           RLF      PORTB,1    ;ポートBを1ビット左にローテイト
33           BTFSS   STATUS,C    ;Cフラグが1なら次の命令をスキップ
34           GOTO    LEFT
35
36           RRF      PORTB,1    ;過分ローテイトの復旧
37           RRF      PORTB,1    ;過分ローテイトの復旧
38           GOTO    RIGHT
39
40 TIMER1    MOVLW    D'62'      ;0.1ミリ秒タイマサブルーチン
41           MOVWF    CNT1
42 LOOP1     NOP
43           DECFSZ   CNT1,F
44           GOTO     LOOP1
45           RETURN
46
47 TIMER2    MOVLW    D'100'     ;10ミリ秒タイマサブルーチン
48           MOVWF    CNT2
49 LOOP2     NOP
50           CALL     TIMER1
51           DECFSZ   CNT2,F
52           GOTO     LOOP2
53           RETURN
54
55 TIMER3    MOVLW    D'20'      ;0.2秒タイマサブルーチン
56           MOVWF    CNT3
57 LOOP3     NOP
58           CALL     TIMER2
59           DECFSZ   CNT3,F
60           GOTO     LOOP3
61           RETURN

```


2.4 実行結果・考察

状態	7 6 5 4 3 2 1 0	状態	7 6 5 4 3 2 1 0
1	$80_{16} = 10000000_2$	8	$01_{16} = 00000001_2$
2	$40_{16} = 01000000_2$	9	$02_{16} = 00000010_2$
3	$20_{16} = 00100000_2$	10	$04_{16} = 00000100_2$
4	$10_{16} = 00010000_2$	11	$08_{16} = 00001000_2$
5	$08_{16} = 00001000_2$	12	$10_{16} = 00010000_2$
6	$04_{16} = 00000100_2$	13	$20_{16} = 00100000_2$
7	$02_{16} = 00000010_2$	14	$40_{16} = 01000000_2$

:LED 点灯 :LED 消灯

図 5: 光が流れるプログラム (往復バージョン) の実行結果。上の数字はポート B の値を表す。

図 5 はポート B の出力状態を表している。0.2 秒毎に LED の点灯箇所が右に移動していき (状態 1 ~ 7) 右端 (状態 8) に到達したら、移動方向を左に反転する (状態 8 ~ 14)。左端 (状態 1) に到達したら再びこの一連の動作を繰り返す。ローテイト (RRF, RLF) 命令は C フラグを経由してデータを 1 ビット (右, 左) にシフトするので C フラグにデータが存在するかどうかで、光が右端 (0 ビット目) または左端 (7 ビット目) に移動したことを判定している。C フラグが 1 の場合はオーバーフローかアンダーフローしているので、図 6 に過分ローテートの復旧を示す。状態 9 の次に状態 U がくるが、状態 U に到達したらすぐに過分ローテートの復旧するために RLF 命令を 2 回実行することで、状態 10 に移行し光がなめらかに移動するように見える。

状態	7 6 5 4 3 2 1 0 C	説明
9	$01_{16} = 00000001_2$	0 ビット目点灯
U	$00_{16} = 00000000_2$	C フラグにアンダーフロー
10	$02_{16} = 00000010_2$	過分ローテート復旧 (RLF × 2)

:LED 点灯 :LED 消灯, :C フラグ 1 :C フラグ 0

図 6: 過分ローテート復旧の例。上の数字はポート B の値を表す。

光の移動方向については、Cフラグでの判定と過分ローテートの復旧を行っているため、LEDの点灯データを変えるだけで、移動方向を変えられる。プログラムの9行目

9行目	LEDD	EQU	80H	;左端から右方向にスタート
-----	------	-----	-----	---------------

を次のように変更すれば、スタート時の移動方向を反対にできる。

9行目	LEDD	EQU	01H	;右端から左方向にスタート
-----	------	-----	-----	---------------

次に過分ローテートの復旧がない場合について考えてみる。

状態	7 6 5 4 3 2 1 0 C	説明
7	$02_{16} = 00000010_2$	1ビット目点灯
8	$01_{16} = 00000001_2$	0ビット目点灯
U	$00_{16} = 00000000_2$	Cフラグにアンダーフロー
8	$01_{16} = 00000001_2$	0ビット目点灯
9	$02_{16} = 00000010_2$	1ビット目点灯

:LED点灯 :LED消灯, :Cフラグ1 :Cフラグ0

図 7: 過分ローテート復旧がない場合。上の数字はポート B の値を表す。

図 7 のように過分ローテートの復旧がないと、全ての LED が 0.5 秒間点灯しないので不自然に移動してるように見える。

3 リスト 5-12(パルスモータ 1-2 相励磁プログラム)

3.1 プログラム説明

1 相励磁と 2 相励磁を交互に繰り返して、パルスモータを制御するプログラム。時計回りに配置されている 4 つのコイル X, Y, \bar{X}, \bar{Y} について順に電流を流していく。4 つのコイル X, Y, \bar{X}, \bar{Y} はこの順に 90° ずつ配置されている。励磁の順序は最初に X 、次に X と Y 、次に Y のように行う。すなわち、コイルを 1 つ励磁し、次に隣のコイルと合わせて 2 つ励磁し、次に隣のコイル 1 つだけを励磁するのを繰り返す。この動作をまとめると表 1 になる。表 1 の X, Y, \bar{X}, \bar{Y} の組をデータとして予めファイルレジスタに入れておいて、これらを順に呼び出して出力する。

表 1: 1-2 相励磁の場合のコイルに流す電流の組の順序

	1	2	3	4	5	6	7	8
X	1	1	0	0	0	0	0	1
Y	0	1	1	1	0	0	0	0
\bar{X}	0	0	0	1	1	1	0	0
\bar{Y}	0	0	0	0	0	1	1	1
データ	8_{16}	C_{16}	4_{16}	6_{16}	2_{16}	3_{16}	1_{16}	9_{16}

3.2 フローチャート

図 8 にこのプログラムのフローチャートを示す。

3.3 ソースコード

```
1 ; *****
2 ; リスト 5.12
3 ; パルスモータ 1-2 相励磁 プログラム
4 ; 間接アドレッシング使用
5 ; *****
6 LIST P=P16F84A ;使用するPICを指定
7 INCLUDE "P16F84A.INC" ;読み込む設定ファイルを指定
8 ; *****
9 WORK0 EQU 0CH ;回転データ格納領域
10 WORK1 EQU 0DH
11 WORK2 EQU 0EH
12 WORK3 EQU 0FH
13 WORK4 EQU 10H
14 WORK5 EQU 11H
15 WORK6 EQU 12H
16 WORK7 EQU 13H
17 TMP EQU 14H ;Wレジスタ待避用
18 CNT1 EQU 15H ;タイマ1用カウンタ変数
19 CNT2 EQU 16H ;タイマ2用カウンタ変数
20 CNT3 EQU 17H ;タイマ3用カウンタ変数
21 ; *****
22 ORG 0 ;プログラムを格納する先頭アドレス
23
24 BSF STATUS,RP0 ;バンク1を選択
25 CLRF TRISB ;ポートBを出力モードに設定
```

26		BCF	STATUS ,RPO	;バンク0を選択
27		CLRF	PORTB	;ポートBをクリア(モータ停止)
28				
29		MOVLW	08H	;回転データの格納
30		MOVWF	WORK0	
31		MOVLW	0CH	
32		MOVWF	WORK1	
33				
34		MOVLW	04H	
35		MOVWF	WORK2	
36		MOVLW	06H	
37		MOVWF	WORK3	
38		MOVLW	02H	
39		MOVWF	WORK4	
40		MOVLW	03H	
41		MOVWF	WORK5	
42		MOVLW	01H	
43		MOVWF	WORK6	
44		MOVLW	09H	
45		MOVWF	WORK7	
46				
47	NEW	MOVLW	WORK0	;WORK0をアドレスとして読み取る
48		MOVWF	FSR	;FSRにアドレスを書き込む
49	MADA	MOVF	INDF,W	;INDFレジスタを読み取る
50		MOVWF	PORTB	;回転データをポートBに出力
51		MOVWF	TMP	;Wレジスタの待避
52		CALL	TIMER3	;タイマの呼び出し
53		MOVF	TMP,W	;Wレジスタの回復
54		SUBWF	WORK7,W	;最後の回転データとの照合
55		BTFSC	STATUS,Z	;照合結果チェック
56		GOTO	NEW	;最後まで行っていれば最初から
57				
58		INCF	FSR,F	;まだなら、FSRを1番地分進める
59		GOTO	MADA	;次のWORK領域を読み取る
60				
61	TIMER1	MOVLW	D'62'	;0.1ミリ秒タイマサブルーチン
62		MOVWF	CNT1	
63	LOOP1	NOP		
64		DECFSZ	CNT1,F	
65		GOTO	LOOP1	
66		RETURN		
67				
68	TIMER2	MOVLW	D'100'	;10ミリ秒タイマサブルーチン
69		MOVWF	CNT2	
70	LOOP2	NOP		
71		CALL	TIMER1	
72		DECFSZ	CNT2,F	
73		GOTO	LOOP2	
74		RETURN		
75				
76	TIMER3	MOVLW	D'50'	;0.5秒タイマサブルーチン
77		MOVWF	CNT3	
78	LOOP3	NOP		
79		CALL	TIMER2	
80		DECFSZ	CNT3,F	
81		GOTO	LOOP3	
82		RETURN		
83				
84		END		;プログラムの終わり

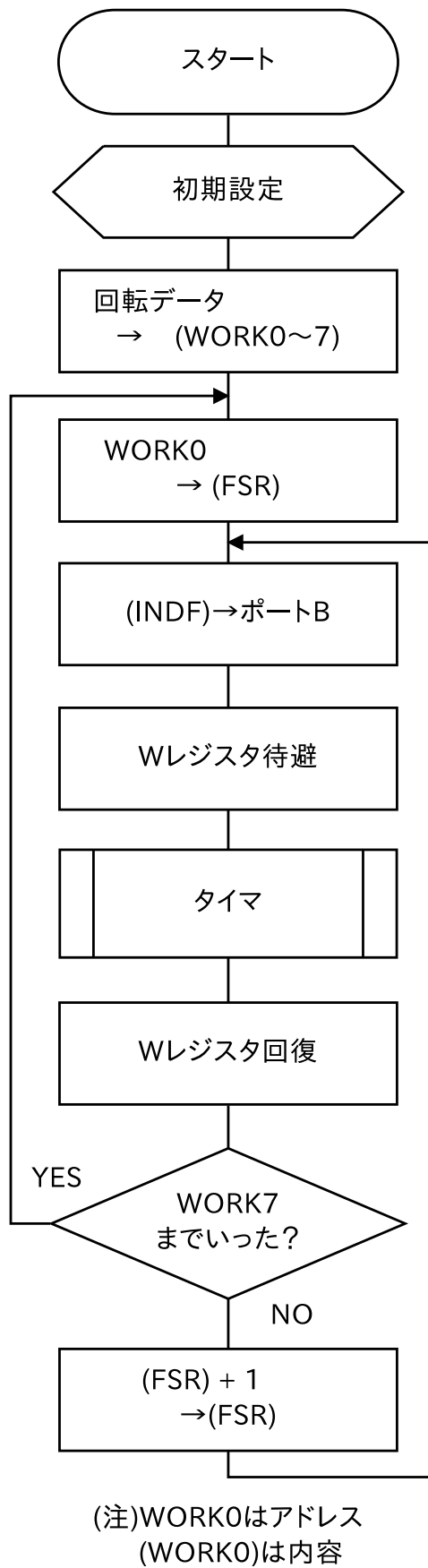


図 8: パルスモータ 1-2 相励磁プログラムのフローチャート

3.4 実行結果・考察

実行する前にボードのRB0をDOWNにスイッチしておく。実験で使ったボードにはパルスモータが接続されていないのでLEDで動作を確認する。

状態	7 6 5 4 3 2 1 0 $X Y \bar{X} \bar{Y}$	データ
1		$8_{16} = 00001000_2$
2		$C_{16} = 00001100_2$
3		$4_{16} = 00000100_2$
4		$6_{16} = 00000110_2$
5		$2_{16} = 00000010_2$
6		$3_{16} = 00000011_2$
7		$1_{16} = 00000001_2$
8		$9_{16} = 00001001_2$

:LED 点灯 :LED 消灯

図 9: パルスモータ 1-2 相励磁プログラムの実行結果

図 9 はポート B の出力状態を表している。8 個の LED の内右側の 4 つがそれぞれコイル X, Y, \bar{X}, \bar{Y} に対応している。状態 1~8 は LED の点灯状態で、0.5 秒毎に移行していく。状態 8 の次は状態 1 に戻る。各コイルでみると対応する LED が 1.5 秒点灯して、2.5 秒消灯するように変化している。各コイルの動作は 1 秒ずれているので、1 相励磁と 2 相励磁が交互に繰り返す様子が確認できた。

この動作を実現するためにあらかじめデータをメモリに入れておいて、それを読みだして PORTB にセットしている。FSR レジスタの内容を 1 ずつ変えて、INDF レジスタを読みだすと FSR レジスタの内容が指すメモリアドレスにアクセスすることで読み出しを実現している。つまり間接アドレッシング方式を用いている。

読みだす順序を変えるだけで、モーターを反転させることができる。そのためには読みだす最初のアドレスを反対からにして読みだすアドレスを 1 ずつ減らしていく。

47 行 目	NEW	MOVLW	WORK0
54 行 目		SUBWF	WORK7
58 行 目		INCF	FSR, F

このコードを

47 行 目	NEW	MOVLW	WORK7
54 行 目		SUBWF	WORK0
58 行 目		DECF	FSR, F

このように変えるとよい。

回転の速さは TIMER3 のカウント数で調整すればよい。

1 相制御にするには WORK0 ~ WORK7 に格納するデータを表 2 のようにして、WORK5 ~ WORK7 は使わず、コードの 54 行目を

54 行 目		SUBWF	WORK3
--------	--	-------	-------

のように変えれば良い。

2 相制御なら、同様に表 3 のデータを与えればよい。

表 2: 1 相励磁のデータ

WORK0	$08_{16} = 00001000_2$
WORK1	$04_{16} = 00000100_2$
WORK2	$02_{16} = 00000010_2$
WORK3	$01_{16} = 00000001_2$

表 3: 2 相励磁のデータ

WORK0	$0C_{16} = 00001100_2$
WORK1	$06_{16} = 00000110_2$
WORK2	$03_{16} = 00000011_2$
WORK3	$09_{16} = 00001001_2$

このようにテーブルを与えて順に読みだす方式は非常に汎用的な制御方法と考えられる。