

# 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)

# 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	プログラム概要 . . . . .	3
1.2	ソースコード . . . . .	3
1.3	フローチャート . . . . .	4
1.4	実行結果 . . . . .	5
1.5	考察 . . . . .	5
1.6	練習問題 5.6 . . . . .	6
1.6.1	問題 . . . . .	6
1.6.2	回答 . . . . .	6
1.7	練習問題 5.7 . . . . .	7
1.7.1	問題 . . . . .	7
1.7.2	回答 . . . . .	7
2	リスト 5-6(光が流れるプログラム (往復バージョン))	8
2.1	プログラム概要 . . . . .	8
2.2	ソースコード . . . . .	8
2.3	フローチャート . . . . .	10
2.4	実行結果 . . . . .	11
2.5	考察 . . . . .	12
2.6	練習問題 5.8 . . . . .	12
2.7	練習問題 5.9 . . . . .	12
2.8	練習問題 5.10 . . . . .	12

# 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 プログラム概要

8個あるLEDの1個を右端や左端から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 ソースコード

```
;5-5
;header
LIST      P=PIC16F84A
INCLUDE   "P16F84A.INC"

;var
LEDD      EQU      80H
CNT1      EQU      0CH
CNT2      EQU      0DH
CNT3      EQU      0EH
;main
ORG       0

BSF       STATUS,RPO
CLRF      TRISB
BCF       STATUS,RPO
BCF       STATUS,C

MOVLW     LEDD
MOVWF     PORTB
REPEAT    CALL    TIMER3
RRF       PORTB,1      ;右方向
;         RLF     PORTB,1    ;左方向
GOTO      REPEAT

TIMER1    MOVLW     D'62'      ;0.1ms
MOVWF     CNT1
LOOP1     NOP
DECFSZ    CNT1,F
GOTO      LOOP1
RETURN

TIMER2    MOVLW     D'100'     ;10ms
MOVWF     CNT2
LOOP2     NOP
CALL      TIMER1
DECFSZ    CNT2,F
GOTO      LOOP2
RETURN

TIMER3    MOVLW     D'50'      ;0.5s
MOVWF     CNT3
LOOP3     NOP
CALL      TIMER2
DECFSZ    CNT3,F
```

```
GOTO    LOOP3
RETURN

END
```

### 1.3 フローチャート



図 1: フローチャート

## 1.4 実行結果

$$80_{16} = 10000000_2$$

$$40_{16} = 01000000_2$$

$$20_{16} = 00100000_2$$

$$10_{16} = 00010000_2$$

$$8_{16} = 00001000_2$$

$$4_{16} = 00000100_2$$

$$2_{16} = 00000010_2$$

$$1_{16} = 00000001_2$$

*C* フラグ

: 点灯    : 消灯

このように、0.5 秒毎に光る場所が、右に動いていく。右端に到達すると、すべての LED が消えるタイミングがある。

## 1.5 考察

ローテイト (RRF) 命令は 1 ビットずつ右にシフトさせるもので、16 進数において 1 ビット右にシフトさせることは 2 で割ることになる。RRF 命令は *C* フラグを経由してデータを回転するので端に到達したら一時的に全部が消える瞬間がある。

PIC のクロック周波数は 10MHz なので、1 クロックあたりは

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

4 クロックで 1 サイクルなので、1 サイクルあたりは

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

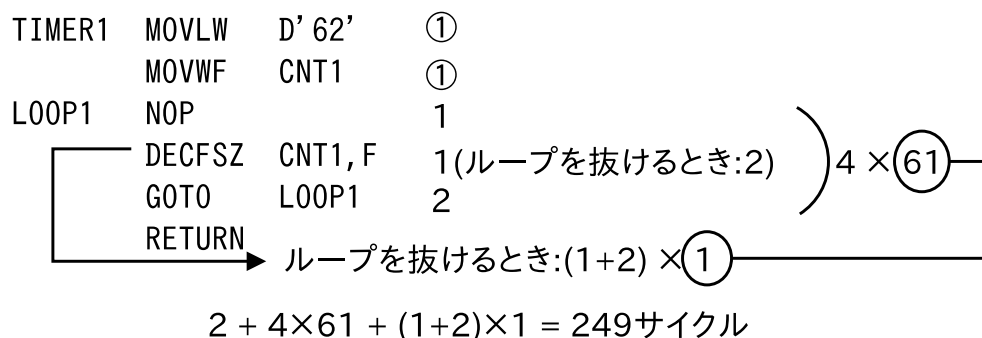


図 2: サイクル数

TIMER1 のサイクル数は図 2 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 消費される。

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

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

合計で 0.5 s のタイマルーチンである。

この操作を 0.5 秒のタイマルーチンで呼び出すことで、0.5 秒毎に光が動いているように見える。

## 1.6 練習問題 5.6

### 1.6.1 問題

リスト 5-5 を点灯が左方向に移動するように変更せよ。

### 1.6.2 回答

;	RRF	PORTB, 1	; 右方向
	RLF	PORTB, 1	; 左方向

このように RRF を RLF に変更する。

RLF 命令は 1 ビットずつ左にシフトさせるもので、16 進数で 1 ビット左にシフトさせることは 2 をかけることになる。RLF 命令も RRF 命令と同様に、C フラグを経由してデータを回転するので、端に到達したら一時的に全部が消える瞬間がある。

## 1.7 練習問題 5.7

### 1.7.1 問題

リスト 5-5 で、点灯の移動するスピードを早くして、目で確認できる最小の移動時間を調べなさい。

### 1.7.2 回答

TIMER1	MOVLW	D'62'	;0.1ms
	MOVWF	CNT1	
LOOP1	NOP		
	DECFSZ	CNT1,F	
	GOTO	LOOP1	
	RETURN		
TIMER2	MOVLW	D'100'	;10ms
	MOVLW	CNT2	
LOOP2	NOP		
	CALL	TIMER1	
	DECFSZ	CNT2,F	
	GOTO	LOOP2	
	RETURN		
TIMER3	MOVLW	D'10'	;0.1s
	MOVWF	CNT3	
LOOP3	NOP		
	CALL	TIMER2	
	DECFSZ	CNT3,F	
	GOTO	LOOP3	
	RETURN		

このように TIMER1 のカウントはそのままにして、TIMER2 と TIMER3 のカウントのところを変更すると、

$$0.1\text{ ms} \times 100 \times 10 = 10\text{ ms} = 0.1\text{ s}$$

と、0.1 秒間隔になりこれが目で確認できる最小の移動時間であるとわかった。

## 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 プログラム概要

8個LEDの1個を左端や右端から順次点灯していき、端に到達したら逆方向に点灯させることでLEDの点灯が、往復して流れるように見えるプログラム。

### 2.2 ソースコード

```
;5-6
;header
LIST      P=P16F84A
INCLUDE   "P16F84A.INC"

;var
;LEDD     EQU      80H           ;左端から右方向にスタート
LEDD      EQU      01H           ;右端から左方向にスタート
CNT1      EQU      0CH
CNT2      EQU      0DH
CNT3      EQU      0EH
;main
ORG       0

BSF       STATUS,RP0
CLRF      TRISB
BCF       STATUS,RP0
BCF       STATUS,C
MOVLW     LEDD
MOVWF     PORTB

RIGHT     CALL      TIMER3
RRF       PORTB,1
BTFSS     STATUS,C
GOTO      RIGHT

RLF       PORTB,1           ;過分ローテイトの復旧
RLF       PORTB,1

LEFT      CALL      TIMER3
RLF       PORTB,1
BTFSS     STATUS,C
GOTO      LEFT

RRF       PORTB,1           ;過分ローテイトの復旧
RRF       PORTB,1
GOTO      RIGHT

TIMER1    MOVLW     D'62'
MOVWF     CNT1
LOOP1     NOP
DECFSZ    CNT1,F
GOTO      LOOP1
RETURN
```



```

TIMER2  MOVLW    D'100'
        MOVWF    CNT2
LOOP2   NOP
        CALL     TIMER1
        DECFSZ   CNT2,F
        GOTO     LOOP2
        RETURN

TIMER3  MOVLW    D'20'
        MOVWF    CNT3
LOOP3   NOP
        CALL     TIMER2
        DECFSZ   CNT3,F
        GOTO     LOOP3
        RETURN

        END

```

## 2.3 フローチャート

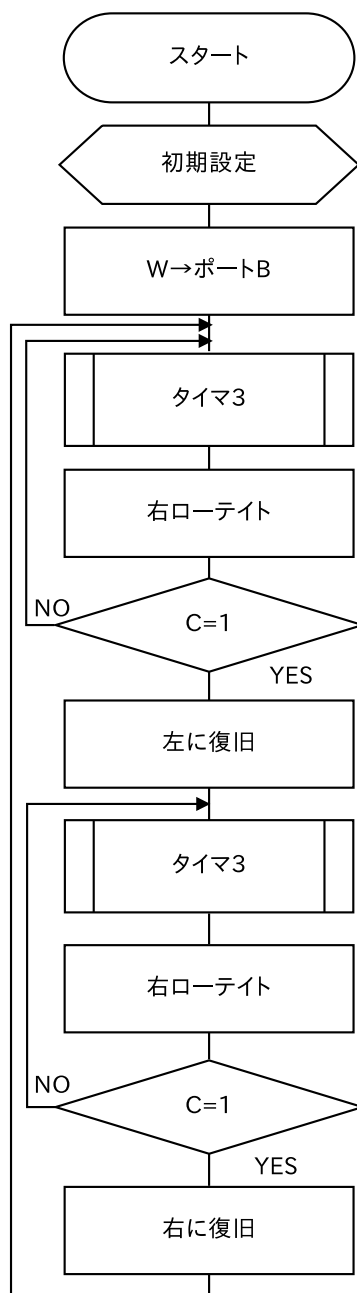


図 3: フローチャート

## 2.4 実行結果

$$80_{16} = 10000000_2$$

$$40_{16} = 01000000_2$$

$$20_{16} = 00100000_2$$

$$10_{16} = 00010000_2$$

$$8_{16} = 00001000_2$$

$$4_{16} = 00000100_2$$

$$2_{16} = 00000010_2$$

$$1_{16} = 00000001_2$$

$$2_{16} = 00000010_2$$

$$4_{16} = 00000100_2$$

$$8_{16} = 00001000_2$$

$$10_{16} = 00010000_2$$

$$20_{16} = 00100000_2$$

$$40_{16} = 01000000_2$$

$$80_{16} = 10000000_2$$

: 点灯    : 消灯

左端から、0.2 秒毎に光るところが右に動いていき右端になったら、左方向に戻ってくる。

## 2.5 考察

ローテイト命令は、Cフラグを含めてシフトするので、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Cフラグで判定。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、過分ローテイトの復旧（2ビット復旧）することで、なめらかに移動するように見える。

## 2.6 練習問題 5.8

LEDD	EQU	80H	; 左端から右方向にスタート
;LEDD	EQU	01H	; 右端から左方向にスタート

このように、LEDD のデータを 80H から 01H に変更する。

## 2.7 練習問題 5.9

過分ローテイトの復旧がないと、すべての LED が点灯しない瞬間が生じる。

## 2.8 練習問題 5.10