

# ECE 376 - Homework #2

Assembler & Flow Charts - Due Monday, September 9th

## Assembler Programming

1) Determine the contents of registers W, A, and B after each assembler command:

| Command   | W   | A  | B   |
|-----------|---|----|---|
| ; Start   | 11  | 12 | 13  |
| incf B,W  | 14  | 12 | 13  |
| decf A,F  | 14  | 11 | 13  |
| iorwf A,W | $\begin{array}{r} 14 = 0000\ 1110 \\ 11 = 0000\ 1011 \\ \hline 15 = 0000\ 1111 \end{array}$ | 11 | 13  |
| andwf B,F | 15  | 11 | $\begin{array}{r} 13 = 0000\ 1101 \\ 15 = 0000\ 1111 \\ \hline 13 = 0000\ 1101 \end{array}$ |
| movlw 15  | 15  | 11 | 13  |
| subwf A,W | $\begin{array}{r} 11 - 15 = -4 \\ -4\ \text{or}\ +252 \end{array}$                          | 11 | 13  |

2) Convert the following C code to assembler (8-bit operations)

```
; unsigned char A, B, C;
A equ 0
B equ 1
C equ 2

; A = B + 2*C + 3;
    movlw    3
    addwf   B
    addwf   C
    addwf   C
    movwf  A
```

3) Convert the following C code to assembler: (16-bit operations)

```
;unsigned int A, B, C;
A equ 0
B equ 2
C equ 4

;A = B + 2*C + 3;

    movlw    0
    movwf   A
    movlw    3
    movwf   A+1

    movwf   B+1
    addwf   A+1,F
    movf    B
    addwfc  A,F

    movwf   C+1
    addwf   A+1,F
    movf    C
    addwfc  A,F

    movwf   C+1
    addwf   A+1,F
    movf    C
    addwfc  A,F
```

#### 4) Convert the following C code to assembler (if-statements)

```
;unsigned char A, B, C;  
A equ 0  
B equ 1  
C equ 2
```

```
;if(A > B) C += 1  
    movf    B,W  
    cpfgt   A  
    goto    L1  
    incf    C,F
```

```
;if(A < B) C -= 1  
L1:   movf    B,W  
      cpfslt  A  
      goto    L2  
      decf    C,F
```

```
L2:   nop
```

5) The flow chart on the left is for turning your PIC into a JK flip-flop:

- RB0 = CLK (output changes on the rising edge of RB0)
- RB1 = K
- RB2 = J
- PORTC = Q (all eight pins)

Write the corresponding assembly code

```

movlw    0xFF
movwf    TRISB
clrf     TRISC

L1:      btfsc    PORTB, 0
        goto    L1

L2:      btfss    PORTB, 0
        goto    L2

L3:      btfsc    PORTB, 2
        goto    L3a
        btfsc    PORTB, 1
        goto    L5
        goto    L1

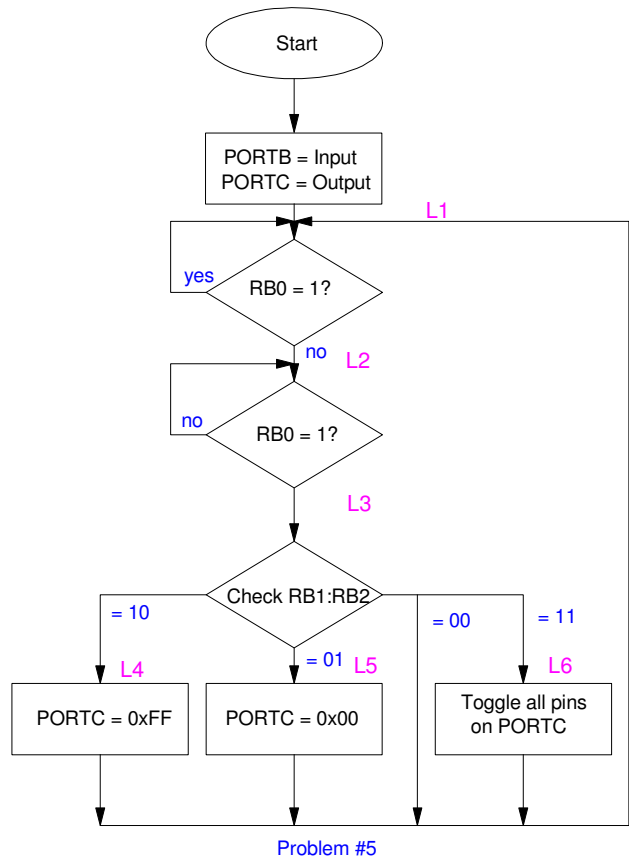
L3a:     btfsc    PORTB, 1
        goto    L6

L4:      movlw    0xFF
        movwf    PORTC
        goto    L1

L5:      clrf     PORTC
        goto    L1

L6:      comf     PORTC, F
        goto    L1

```



Problem #5

6) The flow chart to the right generates random numbers when you press a button:

- RB0: 6-sided die
- RB1: 8 sided die
- RB2: 16-sided die

Write the corresponding assembly code

```

L1:  movlw    0xFF
     movwf   TRISB
     clrf    TRISC

L2:  movlw    0
     cpfseq  PORTB
     goto    L3
     goto    L2

L3:  btfsc   PORTB, 0
     goto    L4
     btfsc   PORTB, 1
     goto    L5
     goto    L6

L4:  movlw    5
     cpfeq   PORTC
     goto    L4a
     clrf    PORTC
     goto    L7

L4a: incf    PORTC, F
     goto    L7

L5:  incf    PORTC, W
     andlw   0x07
     movwf   PORTC
     goto    L7

L6:  incf    PORTC, W
     andlw   0x0F
     movwf   PORTC
     goto    L7

L7:  movlw    0
     cpfseq  PORTB
     goto    L3
     goto    L2

```

