

16.317: Microprocessor Systems Design I

Fall 2013

Exam 3

December 16, 2013

Name: _____ ID #: _____

For this exam, you may use a calculator and one 8.5" x 11" double-sided page of notes. All other electronic devices (e.g., cellular phones, laptops, PDAs) are prohibited. If you have a cellular phone, please turn it off prior to the start of the exam to avoid distracting other students.

The exam contains 4 questions for a total of 100 points. Please answer the questions in the spaces provided. If you need additional space, use the back of the page on which the question is written and clearly indicate that you have done so.

Please note that Question 4 has three parts, but you are only required to complete two of the three parts. You may complete all three parts for up to 10 points of extra credit. If you do so, **please clearly indicate which part is the extra one—I will assume it is part (c) if you mark none of them.**

You will be provided with seven pages (4 double-sided sheets) of reference material for the exam: the PIC16F684 instruction set, memory map, and block diagram, as well as a list of the 80386 instructions and condition codes we have covered this semester. You do not have to submit these pages when you turn in your exam.

You will have three hours to complete this exam.

Q1: Multiple choice	/ 20
Q2: Reading PIC assembly	/ 16
Q3: PIC programming sequences	/ 24
Q4: Complex operations	/ 40
TOTAL SCORE	/ 100
EXTRA CREDIT	/ 10

1. (20 points, 5 points per part) **Multiple choice**

For each of the multiple choice questions below, clearly indicate your response by circling or underlining the single choice you think best answers the question.

a. Given the PIC assembly code below, what is the final value of the variable x?

```
    clrf    x
    movlw  0x0E
    movwf  COUNT
L:   incf   x
    decfsz COUNT, F
    goto   L
```

- i. 0
- ii. 1
- iii. 14 (0xE)
- iv. 15 (0xF)
- v. 241 (0xF1)

b. Under what conditions will the following code jump to the label L1?

```
    btfss  STATUS, C
    goto   END
    movlw  0x11
    subwf  x, F
    btfsc  STATUS, Z
    goto   L1
END:
```

- i. C = 0
- ii. C = 0 and x = 0x11
- iii. C = 1
- iv. C = 1 and x = 0x11
- v. C = 1 and x ≠ 0x11

1 (continued)

c. Which of the following statements about shift and rotate operations using 16-bit values (and therefore multiple registers) on the PIC16F684 are true? Note that these statements assume the C bit is modified before performing any rotate operations to ensure the correct bit is shifted in.

- a. In a 16-bit right shift, the most significant byte should be shifted first to ensure that the correct data is transferred between bytes.
- b. In a 16-bit rotate operation that does not include the carry, the initial value of the C bit should always be 1.
- c. Every 16-bit shift or rotate operation requires at least three PIC instructions, even if the shift amount is 1.
- d. In a 16-bit arithmetic right shift, the initial value of the C bit depends on the most significant bit (bit 15) of the value being shifted.

- i. Only A
- ii. B and C
- iii. A and C
- iv. B and D
- v. A and D

d. Circle one (or more) of the choices below that you feel best “answers” this “question.”

- i. “Thanks for the free points.”
- ii. “I don’t REALLY have to answer the last three questions, do I?”
- iii. “It’s about time we have a test that doesn’t start at 8:00 AM.”
- iv. None of the above.

2. (16 points) **Reading PIC assembly**

Show the result of each PIC 16F684 instruction in the sequences below. Be sure to show not only the state of updated registers, but also the carry (C) and zero (Z) bits.

a. cblock 0x20

 x
 endc

 movlw 0x10

 sublw 0x31

 clrf x

 decf x, F

 xorwf x, F

 swapf x, W

 btfsc x, 7

 bcf x, 3

3. (24 points) **PIC programming sequences**

Complete each program below by writing the appropriate PIC instruction into each of the blank spaces. The purpose of each instruction is described in a comment to the right of the blank.

a. (12 points)

This short program uses an 8-bit variable, COUNT, to count from 0 to 100.

```
_____ ; Set COUNT = 0  
  
L: _____ ; Increment COUNT  
  
movlw    100                ; W = 100  
  
_____ ; Use two instructions  
_____ ; to compare current  
_____ ; COUNT value to 100  
_____ ; If COUNT is 100, skip  
_____ ; goto instruction and  
_____ ; exit loop  
  
goto    L                  ; Return to start of loop
```

3 (continued)

b. (12 points)

This program snippet calls a function, SetBits, which reads the current state of PORTA and isolates the lowest bits to produce a value between 0 and 3. Based on that value, SetBits returns a value that can be used to set 0, 1, 2, or 3 bits in register PORTC without changing any other bits. The short snippet below shows how the function is called; you have to determine how the return value is used.

; Code below is short snippet from main program

```
call    SetBits                ; Call SetBits function
_____ ; Use return value from SetBits
; to set appropriate bits
; in PORTC
... ; Remainder of main program
; (not shown)
```

SetBits:

```
movf    PORTA, W              ; Read state of PORTA into W
_____ ; Isolate lowest bits of PORTA to
; get a value between 0 and 3
addwf   PCL, F                ; Add that value to PCL
; to pick instruction below
_____ ; Return value used to set no
; bits in PORTC
retlw   b'00000001'          ; Return value that will set
; lowest bit in PORTC
_____ ; Return value used to set
; lowest two bits in PORTC
_____ ; Return value used to set
; lowest three bits in PORTC
```

4. (40 points, 20 points per part) **Complex operations**

For each of the following complex operations, write a sequence of PIC 16F684 instructions that performs an equivalent operation. **CHOOSE ANY TWO OF THE THREE PARTS** and fill in the space provided with appropriate code. **You may complete all three parts for up to 10 points of extra credit, but must clearly indicate which part is the extra one—I will assume it is part (c) if you mark none of them.**

Assume that 8-bit variables “TEMP” and “COUNT” have been defined for cases where you may need extra variables.

Finally, please note that you are not required to write comments describing each instruction. You are certainly welcome to do so if you feel it will make your solution clearer to the instructor.

- a. You are given three 16-bit values X, Y, and Z. You can access individual bytes within each value—“X” contains bytes XH and XL (XL is the least-significant byte), “Y” contains bytes YH and YL, and “Z” contains bytes ZH and ZL. Perform the following 16-bit addition:

$$X = Y + Z \quad (\text{Add } Y \text{ and } Z \text{ and store the result in } X, \text{ without changing } Y \text{ or } Z)$$

4 (continued)

b. Given two 8-bit variables, X and B, copy the value of bit position B within variable X into the carry flag. For example:

- If $X = 0x03$ and $B = 0x00$, set C to the value of bit 0 within X.
 - Since $X = 0x03 = 0000\ 0011_2$, $C = 1$
- If $X = 0xC2$ and $B = 0x04$, set C to the value of bit 4 within X.
 - Since $X = 0xC2 = 1100\ 0011_2$, $C = 0$

Note that:

- This operation is very similar to the bit test (BT) instruction in the x86 architecture.
- Since B is not a constant, you cannot use the value of B directly in any of the PIC bit test instructions (for example, `btfsc X, B` is not a valid instruction).
- Your code should not modify either X or B.

4 (continued)

c. Given an 8-bit variable, Y, perform the multiplication:

$$Y = Y * 10$$

Hint: Note that multiplication by a constant amount can be broken into a series of shift and add operations. For example:

- $X * 2$ can be implemented by shifting X to the left by 1 ($X \ll 1$)
- $X * 5$ can be implemented as $(X * 4) + X = (X \ll 2) + X$