16.317: Microprocessor-Based Systems I

Spring 2012

Exam 3 Solution

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. Which of the following statements about microcontrollers are true?

A. A microcontroller is simply another name for a microprocessor.

B. Microcontrollers are typically low-power devices, making them ideal for designing battery-powered embedded systems.

C. All microcontrollers possess the same capabilities and peripherals.

D. Typical limitations of microcontrollers include storage space and processing power.

i. A and C

ii. B and C

iii. A and D

iv. B and D

b. What is the major benefit of using a Harvard memory architecture over a von Neumann memory architecture?

i. Harvard architectures store data and instructions in the same memory module, therefore making memory accesses simpler.

ii. Harvard architectures separate data and instruction memory, allowing a processor to simultaneously access both and therefore improve performance.

iii. The Harvard name alone inflates the importance of the Harvard memory architecture, just as professors at that institution inflate the grades of their students.

iv. None of the above

1 (cont.)

c. Which of the following operations can be performed using the PIC 16F684 system stack?

A. Storing a return address for a function call.
B. Storing general purpose registers prior to calling a function.
C. Passing variables to a function.
D. Storing the return address for an interrupt routine.

i. Only A

ii. B and C

iii. <u>A and D</u>

iv. A, B, and C

v. A, B, C, and D

d. You are given the PIC function shown below:

```
F:    movf   PORTA, W
      andlw  0x02
      addwf  PCL, F
      retlw  0x03
      retlw  0x02
      retlw  0x10
      retlw  0xFE
```

What is the return value of this function if the current value of PORTA is 0x17?

i. 0x03

ii. 0x02

iii. <u>0x10</u>

iv. 0xFE

2. (38 points) ***Reading PIC assembly language***

Show the result of each PIC 16F684 instruction in the sequences below.

a. (22 points)

```
cblock 0x20
    var1, var2
endc
```

movlw	0x15	***W = 0x15***
movwf	var1	***var1 = W = 0x15***
comf	var1, F	***var1 = ~var1 = ~0x15 = ~0001 0101$_2$ = 1110 1010$_2$ = 0xEA***
incf	var1, F	***var1 = var1 + 1 = 0xEA + 1 = 0xEB***
addwf	var1, W	***W = var1 + W = 0xEB + 0x15 = 0x00***
btfss	STATUS, Z	***Z = 1 because previous result is 0; program skips next instruction***
goto	L1	
swapf	var1, W	***W = value in var1 with nibbles swapped = 0xBE***
goto	L2	***Unconditional jump to L2***
L1: addlw	0x11	
L2: movwf	var2	***var2 = W = 0xBE***

2 (cont.)

b. (16 points)

```
    cblock 0x20
        x
    endc
```

movlw	0xC3	***W = 0xC3***
movwf	x	***x = W = 0xC3***
xorlw	0x06	***W = W XOR 0x06 = 0xC3 XOR 0x06 = 0xC5***
iorlw	0x18	***W = W OR 0x18 = 0xC5 OR 0x18 = 0xDD***
andwf	x, F	***x = x AND W = 0xC3 AND 0xDD = 0xC1***
bsf	STATUS, C	***Carry bit = 1***
rlf	x, W	***W = x rotated 1 bit to the left, through carry***
		(Carry, x) = 1 1100 0001_2 before rotation
		After rotation, (Carry, W) = 1 1000 0011_2
		→ Carry = 1, W = 0x83

3. (42 points, 14 per part) ***Writing PIC assembly code***

For each of the following 80386 instructions, write a sequence of PIC 16F684 instructions that performs an equivalent operation. The operation is described in italics.

Assume that variables with the same names are defined for all 8-bit 80386 registers (for example, "AL" and "BL"). If an operation uses a 16-bit register (e.g., AX), you can address each byte within that register (e.g. AH and AL). Also assume "TEMP" has been defined for cases where you may need an extra variable.

Finally, note that shift or rotate operations should not be done by simply writing copies of the PIC rotate instructions. Use the shift amount provided as a literal value that will help determine the number of times you shift or rotate.

a. XCHG AL, BL *(Swap contents of AL and BL)*

Solution:

```
        movf    AL, W
        movwf   TEMP        ; TEMP = AL
        movf    BL, W
        movwf   AL          ; AL = BL
        movf    TEMP, W
        movwf   BL          ; BL = TEMP
```

b. MOVSX AX, BL *(Move byte in BL to 16-bit register AX and sign-extend it to fill upper 8 bits)*

Solution:

```
        movf    BL, W
        movwf   AL          ; AL = BL
        clrw                ; W = 0—will be assigned to AH if B is positive
        btfsc   BL, 7       ; Check sign to determine how to extend
        movlw   0xFF        ; B is negative—set W = 0xFF (all 1s)
        movwf   AH          ; Copy contents of W to upper byte of AX
```

3 (cont.)

c. ROR AX, 10 *(Rotate 16-bit value AX right by 10 bits)*

Solution:

```
        movlw   10          ; Initialize loop counter to 10
Loop:   rrf     AH, F       ; Rotate upper byte right
                            ; LSB moves into carry
        rrf     AL, F       ; Rotate lower byte right
                            ; LSB from upper byte shifted into MSB of lower byte
        btfss   STATUS, C   ; Check carry (which equals LSB of lower byte)
        goto    CZero
        bsf     AH, 7       ; Carry = 1 → set MSB of upper byte to 1
        goto    LEnd
CZero:  bcf     AH, 7       ; Carry = 0 → set MSB of upper byte to 0
LEnd:   addlw   -1          ; Decrement loop counter
        btfss   STATUS, Z   ; Skip next instruction if counter = 0
        goto    Loop        ; Return to start of loop
```

The following pages contain references for use during the exam: tables containing the PIC 16F684 instruction set and memory map, as well as a block diagram of the microcontroller. You may detach these sheets from the exam and do not need to submit them when you finish.

Remember that, in the table below:

- f = a register file address
- W = the working register
- d = destination select: “F” for a file register, “W” for the working register
- b = bit position within an 8-bit file register
- k = literal field, constant data or label
- PC = the program counter
- C = the carry bit in the status register
- Z = the zero bit in the status register

TABLE 13-2: PIC16F684 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	1, 2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1, 2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	–	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1, 2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1, 2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2, 3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1, 2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2, 3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1, 2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1, 2
MOVWF	f	Move W to f	1	00	0000	lfff	ffff		
NOP	–	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1, 2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1, 2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	1, 2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1, 2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1, 2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1, 2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1, 2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call Subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	–	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	–	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	–	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	–	Go into Standby mode	1	00	0000	0110	0011	$\overline{TO}$, $\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

FIGURE 2-2: DATA MEMORY MAP OF THE PIC16F684

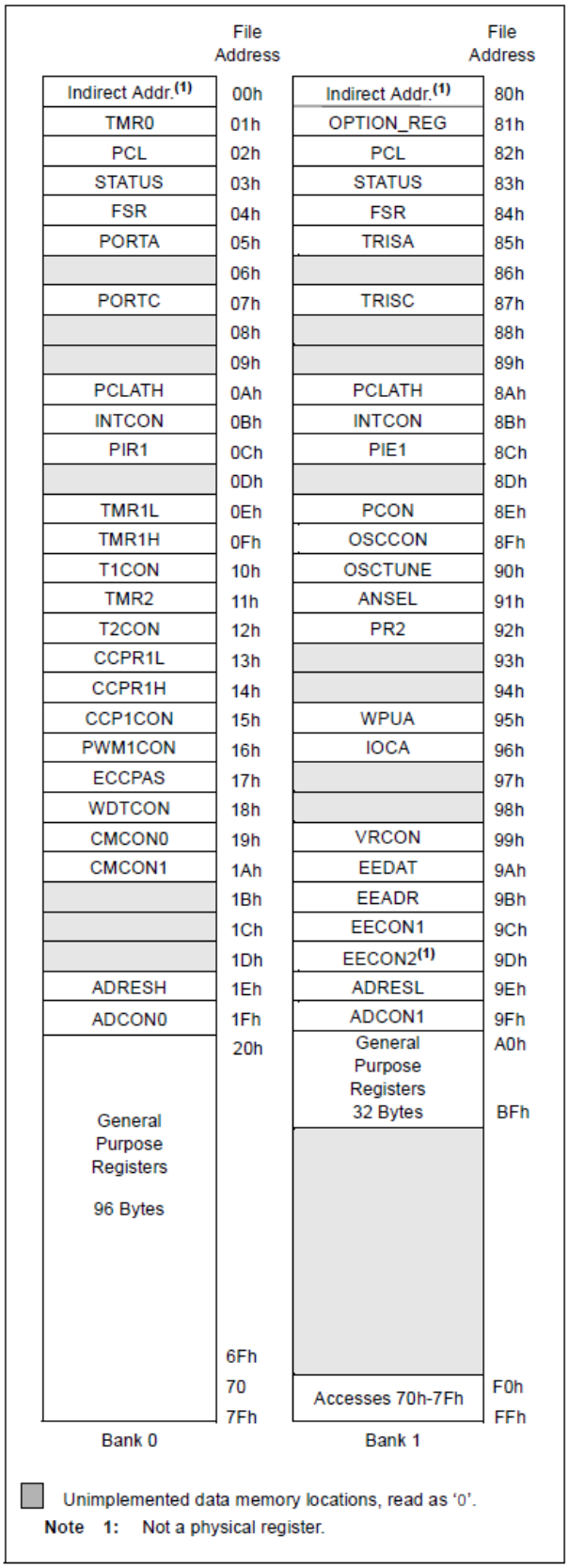
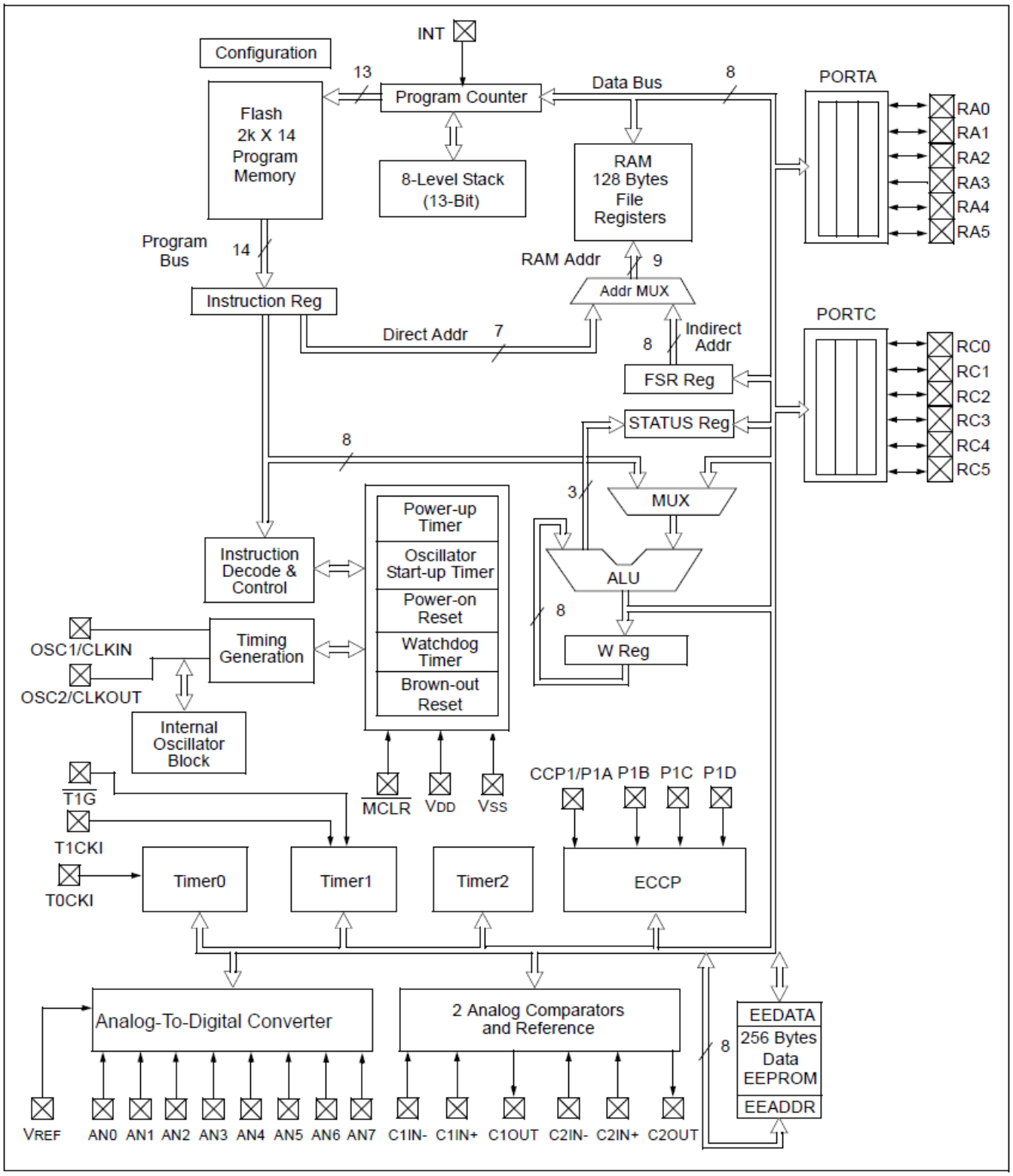


FIGURE 1-1: PIC16F684 BLOCK DIAGRAM



Source for all figures: *"PIC 16F684 Data Sheet", Microchip Technology, Inc. http://ww1.microchip.com/downloads/en/DeviceDoc/41202F-print.pdf*