



Notre Dame University

ECCE Department

Exam #1

Microcontroller Design EEN326

Name: _____

Student ID: _____

Question1	10
Question2	5
Question3	15
Question4	20
Total	/50

- ✓ **Closed Book**- The Instruction Set is at page 7 and 8
- ✓ Simple Calculators are allowed; Scientific Calculators are NOT!!
- ✓ Cellular should be switched OFF
- ✓ Answers on the exam sheet (use back pages for scratch)
- ✓ No borrowing under any circumstances!

Question 1. (10 points)

Choose the correct answer or answer True / False for each of the following statements (1 point each)

1. The downloaded file into the PIC is saved in PIC's memory a) RAM b) FLASH c) EEPROM d) REGISTERS		
2. The Interrupt subroutine uses 's PIC memory. a) RAM b) FLASH c) EEPROM d) REGISTERS		
3. The PIC16F84 has 2 ports RA and RB. The pin number of RA and RB respectively are a) 4, 8 b) 8, 4 c) 5, 8 d) 8, 5 e) 4, 7 f) 7, 4		
4. Four Banks can be used in PIC 16F84	T	F
5. The RB0 port in PIC16Fxx can be assigned as input and output in same program.	T	F
6. The CBLOCK directive is used to allocate variables to memory a) RAM b) FLASH c) EEPROM d) REGISTERS		
1. The instruction code of the PIC16F84 is a) 16 bits b) 14 bits c) 12 bits d) 10 bits e) 8 bits		
8. A 4 MHz crystal can run a PIC at a) 8 MHz b) 4 MHz c) 2 MHz d) 1 MHz		
9. The PIC16F84 is a CISC architecture.	T	F
10. All Arithmetic and Logic operations use the work (W) register.	T	F
11. An interrupt is an asynchronous control break.	T	F

Question 2. (5 points)

Draw the block diagram of a microcontroller.

Question 3. (15 points)

Execute the following program, and show the content of registers REG1, REG2, REG1_TEMP, COUNTER when there is (are) any change(s).

```
#INCLUDE "P16F84A.INC"

CBLOCK 0X020
REG1, REG2,
REG1_TEMP, COUNTER
ENDC

ORG      0X000
GOTO    MAIN

MAIN
MOVLW   B'11010001'
MOVWF   REG1
MOVWF   REG1_TEMP
BCF     STATUS, C

MOVLW   D'8'
MOVWF   COUNTER

RLF     REG1_TEMP, F
RRF     REG2, F
DECFSZ  COUNTER, F
GOTO    $-3

GOTO    $

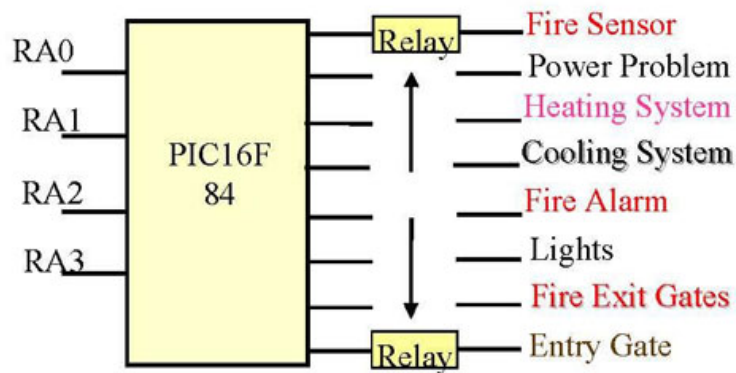
END
```

Count	REG1	REG1_Temp	REG2
8	10001011		

Question 4. (20 points)

Realize a system using a PIC16F84 that can do the following:

- 1- The Entry Gate is OPEN and the lights are ON if RA0 receives the correct code. (0 error, 1 correct). Otherwise the Entry Gate is CLOSED and the lights are OFF.
- 2- The Heating System is ON AND the Cooling System is OFF if RA1 = 0, otherwise (RA1=1) the Heating System is OFF AND the Cooling System is ON.
Note: The Heating System and the Cooling System work normally at all times.
- 3- RA3 is flashing (0 and 1) for a specific DELAY; it switches to ON when it receives an **Interruption** from Fire Sensor or Power Problem.
- 4- When a Fire Sensor is active (1), the Fire Alarm will switch to ON AND Fires Exit Gates will be OPENED AND the Heating System and the Cooling System will switch to OFF.
- 5- If there is a Power problem the Heating System and the Cooling System will switch to OFF.



```
#INCLUDE "P16F84A.INC"
      CBLOCK 0X00C

      ENDC
      ORG    0X000
      GOTO   MAIN
MAIN
```

GOTO \$

```
SETUP
  CLRF   PORTA
  CLRF   PORTB
```

RETURN
END

```
ORG      0X004
MOVWF   W_TEMP
SWAPF   STATUS, W
MOVWF   STATUS_TEMP
```

```
RESTORE
SWAPF   STATUS_TEMP, W
MOVWF   STATUS
SWAPF   W_TEMP, F
SWAPF   W_TEMP, W
RETFIE
```

#	Mnemonic	Operands	Operation explanation	InstructionErreur ! Signet non défini. code				Flag
BYTE-oriented file register operations								
1	ADDWF	f, d	Add W and f	00	0111	d fff	ffff	C, DC, Z
2	ANDWF	f, d	AND W with f	00	0101	d fff	ffff	Z
3	CLRF	f	Clear f	00	0001	1 fff	ffff	Z
4	CLRWERreur ! Signet non défini.	-	Clear W	00	0001	0 fff	ffff	Z
5	COMFERreur ! Signet non défini.	f, d	Complement f	00	1001	d fff	ffff	Z
6	DECFERreur ! Signet non défini.	f, d	Decrement f	00	0011	d fff	ffff	Z
7	DECFSZERreur ! Signet non défini.	f, d	Decrement f, Skip if 0	00	1011	d fff	ffff	
8	INCFErreur ! Signet non défini.	f, d	Increment f	00	1010	d fff	ffff	Z
9	INCFSZ	f, d	Increment f, Skip if 0	00	1111	d fff	ffff	
10	IORWF	f, d	Inclusive OR W with f	00	0100	d fff	ffff	Z
11	MOVF	f, d	Move f	00	1000	d fff	ffff	Z
12	MOVWF	f	Move W to f	00	0000	d fff	ffff	
13	NOP	-	No Operation	00	0000	d fff	ffff	
14	RLF	f, d	Rotate Left f through Carry	00	1101	d fff	ffff	C
15	RRF	f, d	Rotate Right f through Carry	00	1100	d fff	ffff	C
16	SUBWF	f, d	Subtract W from f	00	0010	d fff	ffff	C, DC, Z
17	SWAPF	f, d	Swap NIBBLES in f	00	1110	d fff	ffff	
18	XORWF	f, d	Exclusive OR W with f	00	0110	d fff	ffff	Z
BIT-oriented file register operations								
19	BCF	f, b	BIT Clear f	01	00bb	b fff	ffff	
20	BSF	f, b	BIT Set f	01	01bb	b fff	ffff	
21	BTFSC	f, b	BIT Test f, Skip if Clear	01	10bb	b fff	ffff	
22	BTFSS	f, b	BIT Test f, Skip if Set	01	11bb	b fff	ffff	
Literal and control operations								
23	ADDLW	k	Add literal and W	11	111x	kkkk	kkkk	C, DC, Z

24	ANDLW	k	AND literal with W	11	1111	kkkk	kkkk	Z
25	CALL	k	Call subroutine	10	0kkk	kkkk	kkkk	
26	CLRWDI	-	Clear Watchdog Timer	00	0000	0110	0100	TO, PD
27	GOTO	k	Go to address	10	1kkk	kkkk	kkkk	
28	IORLW	k	Inclusive OR literal with W	11	1000	kkkk	kkkk	Z
29	MOVLW	k	Move literal to W	11	00xx	kkkk	kkkk	
30	RETFIE	-	Return from interrupt	00	0000	0000	1001	
31	RETLW	k	Return with literal in W	11	01xx	kkkk	kkkk	
32	RETURN	-	Return from Subroutine	00	0000	0000	1000	
33	SLEEP	-	Go into stanby mode	00	0000	0110	0011	TO, PD
34	SUBLW	k	Subtract W from literal	11	110x	kkkk	kkkk	C, DC, Z
35	XORLW	k	Exclusive OR literal with W	11	1010	kkkk	kkkk	Z

STATUS (bank0-bank1)

b7	IRP	RP1	RP0	TO	PD	Z	DC	C	b0
----	-----	-----	-----	----	----	---	----	---	----

OPTION REG (bank1)

b7	RBPV	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	b0
----	------	--------	------	------	-----	-----	-----	-----	----

INTCON (bank0 - bank1)

b7	GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	b0
----	-----	------	------	------	------	------	------	------	----