جامعة بيرزيت

BIRZEIT UNIVERSITY

Electrical and Computer Engineering

ENCS238 - Computer Organization
First Exam

Summer Semester 2016 **Date:** Saturday 30/7/2016

Name: _____ Key _____ **ID:** _____.

Instructor: Dr. Abualsoud Hanani

Instructions:

- **You have 90 minutes (1.5 hours), budget your time carefully!**
- **Turn off your mobile.**
- **To make sure you receive credit, please write clearly and show your work.**

Question	Maximum	Mark	ABET SO
1	15		
2	10		C
3	10		E
4	10		A
5	12		A
6	13		A
Total	**70**		

Question 1 (15 marks)

1. Virtually all computer designs are based on the von Neumann architecture. A high level view of this architecture has the following three components:

   a. Buses, memory, input/output controllers
   b. Hard disks, floppy disks, and the CPU
   c. memory, the CPU, and printers
   d. memory, input/output modules, and the CPU

2. Variable MESSAGE is defined as follows:
   MESSAGE DW 000Ah, 000Bh, 000Ch
   How many bytes does the variable MESSAGE occupy in the memory?

   (A) 3 (B) 6 (C) 9 (D) 12

3. When perform the following instructions, how will the FLAGS bits be set.

```
MOV AL, 0C6h
MOV BL, 9Ah
```

```
ADD AL, BL
```

(A) CF=0, OF=0 (B) CF=1, OF=0

(C) CF=0, OF=1 (D) CF=1, OF=1

4. Given that the BL register contains 1111 0000, the effect of the following instruction **AND BL, 0000 0001** is to

a. Clear first bit of BL b. Set first bit of BL

c. clear all bits of BL except first bit d. Set all bits of BL except first bit

5. Accumulator machine that has only single register, uses instruction of

(A) Zero-addressing (B) One-addressing

(C) Two-addressing (D) Three-addressing

6. Which of the following is NOT a characteristic of RISC machines:

a. Large instruction set b. One instruction per clock cycle

c. Simple addressing modes d. Register to register operations

7. A computer's memory is composed of 16M bytes. How many bits are required for memory address if smallest addressable unit is one byte (i.e. word = 1 byte)?

(A) 23 (B) 24

(C) 16 (D) 14

8. Assembly language

a. Uses alphabetic codes in place of binary numbers used in machine language

b. Is the easiest language to write programs

c. Need not be translated into machine language

d. None of above

9. If a register containing data $(11001100)_2$ is subjected to arithmetic shift left operation, then the content of the register after shift shall be

(A) 11001100 (B) 11100110

(C) 10011001 (D) 10011000

10. An assembly language program is typically

a. non-portable b. shorter than an equivalent HLL program

c. harder to read than a machine code program d. slower to execute than a compiled HLL program

1	2	3	4	5	6	7	8	9	10
D	B	D	C	B	A	B	A	D	A

Question 2: (10 points)

Suppose you are designing an instruction set architecture which supports up to 18 different operations. The register file contains 15 registers. One of the instruction types we would like to support specifies an opcode, a destination register, and immediate source values with range -64 to +63. What is the minimum number of bits that are needed to specify each field?

Opcode	Register	Immediate number
5 bits	4 bits	7 bits

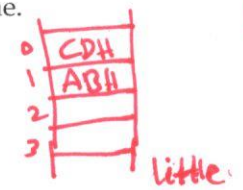
18 operations ⇒ opcode = 5 bits
15 Reg. ⇒ Reg. field = 4 bits
7 bits ⇒ signed range ⇒ $-2^6 \rightarrow +2^6-1$
$-64 \rightarrow +63$

Question 3(10 points)

a) Assume you have a machine that uses 32-bit integers and you are storing the hex value **ABCD** at address 0:

i) Show how this is stored in a big endian machine.

ii) Show how this is stored in a little endian machine.



iii) If you wanted to increase the hex value to **ABCDEF**, which byte assignment would be more efficient, big or little endian? Explain your answer. big-endian because we add EFH at address 2

b) The first two bytes of a 1MBytes memory (word = 1 byte) have the following hex values:

- Byte 0 is 0A
- Byte 1 is B8

If these bytes hold a 16-bit two's complement integer, what is its actual decimal value if:

i) Memory is big endian?

0AB8H = +2744

ii) Memory is little endian?

B80AH = -18422

Question 4(10 marks)

Write assembly program to evaluate the following expression using one-address machine and zero-address machine.

$$X = \frac{A*[B+C*(D+E)]}{F*(G+H)}$$

One-address machine code	Zero-address machine code
load G ADD H Mul F Store T load D ADD E Mul C ADD B Mul A Div T Store X	Push D Push E ADD Push C Mul Push B ADD Push A Mul Push G Push H ADD Push F Mul Div Pop X

Question 5[12 marks]

a) Which of the following instructions is wrong? State the reason and write the correction.

Instruction	Error?	Why?	Correction
MOV SS, 17h	error	Immediate mode not allowed with Segment Register	Mov AX, 17h Mov SS, AX
MOV AX, [CX]	error	CX cannot be used with Register Indirect mode	Mov AX, [BX]
MOV [BP+FFFFh], CX	No error	———	———
MOV DL, [BX + AX]	error	Based-Indexed mode allows only BX, BP as base and SI, DI as index	Mov DL, [BX + SI]
MOV AL, [BX+IP+5h]	error	IP cannot be used with base relative plus index addressing mode	Mov AL, [BX+DI+5h]

b) Assume that
DS = 1111h, SS = 2222h, SI = 3333h, DI = 4444h, BP = 5555h, AX = 6666h, BX = 7777h, CX = 8888h, DX = 9999h

What is the addressing mode, the size of data and the memory address (if any) for of each of the following instructions?

Instruction	Addressing Mode	Data Size	Physical (mem.) Address
MOV [BX], SS	Register Indirect	16 bits	DS : BX = 11110h + 7777h = 18887h
MOV BX, SS	Register	16 bit	Registers, No memory Access
MOV DL, [BP+DI+44]	Base-Indexed with displacement	8 bits	SS : (BP + DI + 44) 22220h 5555h 4444h + 002C = 2BBE5h

Question 6[13 marks]

a) Write assembly instructions to perform the following tasks.

Task	Instructions
Store the offset of variable x in BX	~~MOV DL, 5A~~ ~~MOV AH, 02~~ Lea BX, x ~~INT~~ OR MOV BX, offset x

Send 1234h to output device at port 567h	MOV AX, 1234H
	MOV DX, 5674H
	OUT DX, AX

Display the first letter of you name on the screen	MOV DL, 'A'
	MOV AH, 02
	INT 21H

Invert the three most significant bits of DX	XOR DX, 0E000H
Assume that AL=F0h. Write a single addition instruction that will result in an overflow	ADD AL, 80H

b) Fill in the table below, assuming that **AH=22h**, and **AL = 85h**

Instruction	Result	C flag	O flag
SUB AH, 4Ch	D6h	1	0
ADD AL, FFh	84h	1	0

c) What are the contents of AL and AH after executing the code below?

```
MOV AX , 123
MOV CL, 0Ah
IDIV CL
```

AL= 0Ch AH= 03h