

2022 C

01.09.2021

Ankara University  
Computer Engineering Department  
COM3076 Computer Architecture Final Exam

Name-Surname: Mohammad Shabib

Number:

Signature:

Note: 1. The questions will be answered by only using the techniques discussed in the classes.

2. Write your answers right under the questions or in the given tables.

Good luck!

### QUESTIONS

1. (10 points) Answer the following questions considering modern computers.

a. (5 points) List the four main **structural components of a computer**.

1.	Central processing unit (CPU)
2.	Main memory
3.	I/O
4.	System interconnection (System Bus)

b. (5 points) List the four major **structural components of a processor**.

1.	Control unit
2.	Arithmetic and logic unit (ALU)
3.	Registers
4.	CPU interconnection

2. (10 points) You have a disk with an average seek time of 3ms. Its rotation speed is 20,000rpm. The disk is organized as 512-byte sectors with 500 sectors per track. Assume that the disk uses sequential organization.

a. (5 points) How long would it take to read a file consisting of 3000 sectors?

$$\text{Seek time} = 3 \text{ ms}$$

Answer: 30 ms

$$\text{Rotation speed} \rightarrow \frac{20000}{60000} = \frac{2}{6} \text{ rpm} \rightarrow 1 \text{ round per } 3 \text{ ms} \rightarrow \text{Avg} = 1.5 \text{ ms}$$

$$\text{to read the first track} = 3 \text{ ms} + 1.5 \text{ ms} + 3 \text{ ms} = 7.5 \text{ ms}$$

$$\text{to read 3000 sector} = 7.5 \text{ ms} + (5 * (1.5 + 3)) = 30 \text{ ms}$$

b. (5 points) What is the total size of the file read in **MB**?

Answer: 1.468 MB

$$3000 \times 512 \text{ byte} = \frac{1536000}{2^{20}} \text{ Byte} = 1.4648 \text{ MB}$$

3. (5 points) What is the advantage of using **data striping** in RAID (Redundant Arrays of Independent Disks) systems for all levels? Explain in detail.

The advantage of this layout is that if a single I/O request consists of multiple logically contiguous strips, then up to n strips for that request can be handled in parallel, greatly reducing the I/O transfer time.

\* of physical disks

4. (5 points) List the three approaches for **storing the return address** in instructions.

1.	Register
2.	Start of called procedure
3.	Top of stack

5. (5 points) Assume that the 8bit input word initially stored to a location in memory is 01010101. When a word was read from the same memory location it is found that the check bits for the retrieved word was 1101. What was the word read from the memory? Use the table below for your calculations.

Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit					C8				C4		C2	C1

$$C_1 = D_1 + D_2 + D_4 + D_5 + D_7 = 1 \oplus 0 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$C_2 = D_1 + D_3 + D_4 + D_6 + D_7 = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 1$$

$$C_4 = D_2 + D_3 + D_4 + D_8 = 0 + 1 + 0 + 0 = 1$$

$$C_8 = D_5 + D_6 + D_7 + D_8 = 1 + 0 + 1 + 0 = 0$$

check 1101

G111  
1010

10 → error in the 10<sup>th</sup> bit = D6

Retrieved word

01110101

6. (10 points) Assume that you are asked to implement a 64Mbyte memory using eight 8Mbyte memory chips. The word size for the chips is given as 8bits.

a. (5 points) What should be the length of a memory address?

64Mbyte =  $2^6 * 2^{20}$  so 26 bits is the length

b. (5 points) How would you perform the memory-mapping?

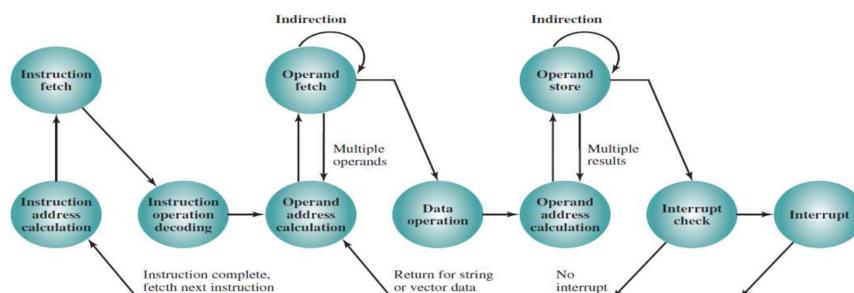
We have 8M byte chips so we need 3 bits to decode  $2^3=8$

7. (5 points) How can we extract the middle four bits of a byte using a logical operator. Explain by giving an example. logical

First Right Shift x2 times  
second And with 00011111(15)

Ex 00110101  
SLR → 00 001101  
And → 00011111  
1101

8. (10 points) Draw the complete instruction cycle diagram with **10 states** (including indirect and interrupt states). Note that you need to draw the diagram with all labels correctly depicted to get a full mark or you will receive none.



9. (15 points) Assume that you have the instruction format described below:

OPCODE	MODE	I	OPE1	OPE2
4 bit	2 bit	2 bit	8 bit	8 bit

The OPCODE field specifies the instruction as below. Assume the instructions take 2 parameters. Example:

SAL RX, 5; performs left arithmetic shift 5 times on RX register.

0001 is the code for SAL (Shift Arithmetic Left) instruction.

0010 is the code for SAR (Shift Arithmetic Right) instruction.

0100 is the code for the ROL (Rotate Left) instruction.

Note that the second operand can be stored in a register or memory location.

The MODE field indicates whether either of the OPE1 and OPE2 are register or memory addresses:

00: OPE1 and OPE2 are memory addresses.

01: OPE1 is register, OPE2 is memory address.

10: OPE1 is memory address, OPE2 is register.

11: OPE1 and OPE2 are registers.

Registers		
Code	Register	Value
00	R0	EB
01	R1	B3
10	R2	00
11	R3	51

Memory	
Address	Value
...	...
A1	03
A2	A3
A3	04
A4	A1
A5	03
...	...

I field indicates whether indirection is used:

00: No indirection is used for both OPE1 and OPE2.

01: Indirection is used only for OPE1.

10: Indirection is used only for OPE2.

11: Indirection is used for both OPE1 and OPE2.

Show the **output result and used operands** for the following instructions for the given memory and register values by giving full explanation of your solution to get a mark. Use (X) notation to indicate indirection where X can be a register (denote as e.g. R1) or a memory location (denote as e.g. [A1]).

i. (5 points)

OPCODE	MODE	I	register OPE1	memory address OPE2
0100	01	00	00000011	<u>10100101</u>

ROL      direct      R3      A5      51

Operands Used	Result (in hexadecimal notation)
R3, [A5]	8A

ii. (5 points)

OPCODE	MODE	I	register OPE1	memory address OPE2 indirect
0001	01	10	00000001	<u>10100100</u>

SAL      R1      XA4      A1      03

Operands Used	Result (in hexadecimal notation)
R1, (A4)	98

iii. (5 points)

OPCODE	MODE	I	register OPE1	memory OPE2
0010	01	11	00000010	<u>10100100</u>

SAR      (R2)      (A2)      L A3 ]      04

(110 101) → 1111110 → F8

Operands Used	Result (in hexadecimal notation)
(R2), (A2) [ R0 ] { A3 ] 04	F8

3/4

10. (10 points) Answer the following questions about cache memories.

a. (5 points) List the four **cache replacement algorithms**.

1.	LRU (Least Recently Used)
2.	FIFO (First-In-First-Out)
3.	LFU (Least Frequently Used):
4.	Random

b. (5 points) Given a main memory of 32Mbytes and a cache memory with a capacity of 128Kbytes. Data can be transferred between the main memory and the cache as blocks of 4 bytes. What should be the length of the tag and line fields if direct mapping is used?

Tag	Line	Word
8 bits	15 bits	2 bits

32 MB:  $2^{32} \times 2^{20} = 2^{52}$  bits to address the main memory  
 1 byte:  $2^3$  bits to address one byte from the block  
 128 KB:  $\frac{128 \text{ KB}}{4} = 32 \text{ KB}$  lines

32 KB:  $2^5 \times 2^{10} = 2^{15}$  bits line size  
 $25 - 2 - 15 = 8$  bits for tag size

11. (15 points) Answer the following questions considering pipelining:

a. (5 points) Suppose you have a program with **100 lines of instructions**, not including any branching. There are a total of **six stages** for an instruction, each stage taking a single cycle. Compute the **speed-up** when pipelining is used rather than sequential execution.

$$\text{Speed-up} = \frac{\text{time before enhancement}}{\text{time after enhancement}} = \frac{600}{105} = 5.714 \quad \left\{ \begin{array}{l} \text{with pipelining} \\ 100 + 6 - 1 = 105 \\ \text{without} = 6 \times 100 = 600 \end{array} \right.$$

b. (10 points) You have a system with a single memory port, but a separate port is available dedicated to the stack. **So you need to be careful when accessing the memory**. Consider the following pipelining scenario to identify the types of the hazards and indicate the numbers of related instructions (which pair of instructions) along with the reason why they occurred. **Note that out instruction uses an I/O port isolated from the memory**.

#	Instruction/Time	1	2	3	4	5	6	7	8	9	10
1	add ax, [si]	FI	DI	FO	EI	WO					
2	push bx		FI	DI	FO	EI	WO				
3	out bx			FI	DI	FO	EI	WO			
4	add cx, dx				FI	DI	FO	EI	WO		
5	and bx, 0					FI	DI	FO	EI	WO	
6	add bx, cx						FI	DI	FO	EI	WO

Hazard #	Hazard Type	Instruction Numbers (instA, instB)	Reason
1.	Resource Hazard	(1,3)-(2,4)-(1,3,5)-(4,6)	trying to fetch operand and fetching instruction at same time
2.	Data Hazard	(2,3),(5,6),(4,6)	Read After write

Data