EXAM [Duration: 1h and 30 min]
MICROPROCESSOR SYSTEMS

Exercise 1 : « 3 pts »

I. Answer the following statements by "right" or "wrong" :

Statements	Right	Wrong
1)) The indirect data transfer between I/O device and memory is called DMA.		
2)) The Von Neumann architecture is the basis of computer architectures.		
3)) ROM is volatile memory.		
4)) The data bus is a bidirectional bus.		
5)) Accumulator is the register in which Arithmetic and Logic calculations are done		
6)) Twelve (12) address lines are required to connect the microprocessor with a 8KB RAM		

Exercise 2: « 6 pts »

Consider a Von Neumann architecture equipped with a 20-bit µP (address) and a data bus of 2 Bytes size running at frequency of 2 GHz.

1. Draw this architecture specifying the location of the different buses and components of the central unit.
2. What is the capacity of memory in bytes and in megabytes?
3. Calculate the time for one cycle of this processor.

Reading a 4-byte block from memory to the processor takes 3 cycles, and processing the 4 bytes by the processor takes 5 cycles.

4. How much time is needed to process all the data contained in memory?

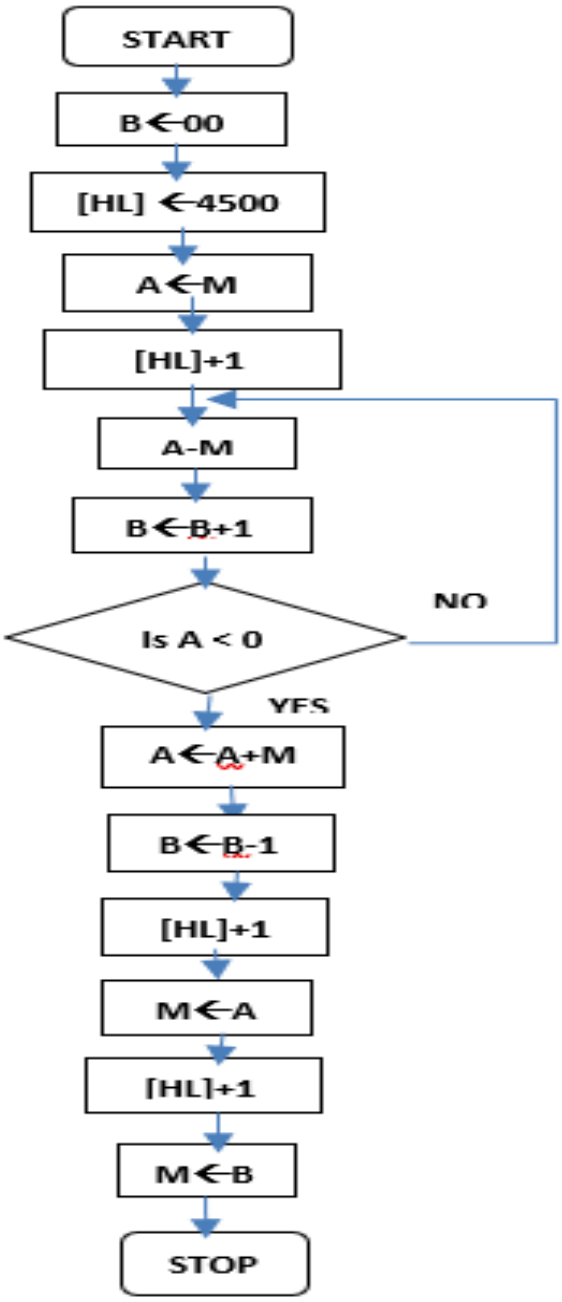
Exercise 3 : « 6 pts »

We have several RAMs (M1) having a capacity of 4M×8 bits, a CS input and a W/E input. We want to create a RAM (M2) with a capacity of 8M×8 bits.

1- Calculate the size of the address buses of the two types of RAM.
2- How many chips of (M1) are needed to create (M2)
3- Give the wiring diagram
4- Give the intervals of addresses of the used memories (M1) for creating the memory (M2).

Exercise 4: « 5 pts »

Consider the following Flowchart:



1. Write the correspondent 8085 INTEL microprocessor assembly program.
2. What does this program execute as an operation?

Exam Correction

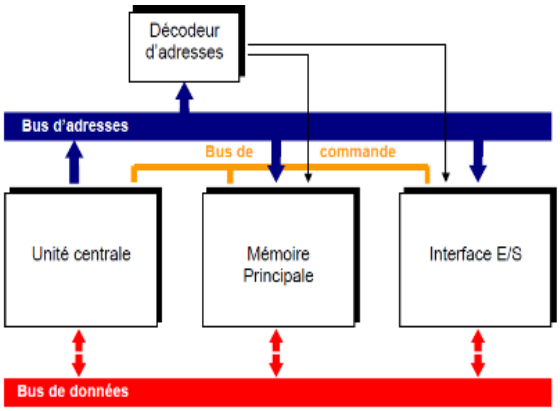
Exercise 1 : « 3 pts »

1. Answer the following statements by "right" or "wrong" :

Statements	Right	Wrong
1)) The indirect data transfer between I/O device and memory is called DMA .**(0,5p)**		✓
2)) The Von Neumann architecture is the basis of computer architectures. **(0,5p)**	✓	
3)) ROM is volatile memory. **(0,5p)**		✓
4)) The data bus is a bidirectional bus. **(0,5p)**	✓	
5)) Accumulator is the register in which Arithmetic and Logic calculations are done**(0,5p)**	✓	
6)) Twelve (12) address lines are required to connect the microprocessor with a 8KB RAM**(0,5p)**		✓

Exercise 2: « 6 pts »

1.The architecture is as follows : **(1p)**



2. the capacity of memory in bytes : $2^{20} \times 2^4 = 2^{24}$bits $= 2^{21}$bytes**(1p)**=2megabytes**(1p)**
3. the time for one cycle of this processor is : T cycle = 1/f horloge = **0.5ns (1p)**
4. to calculate the time needed to process all the data contained in memory, we have to calculate first :
   - number of cycles per block: 8 cycles in total to process 1 block .**(0,5p)**

- number of all blocks = capacity in bytes / number of bytes for one block $=2^{21}$bytes/4=**2^{19}block .(0,5p)**
- number of all cycles = number of all block × number of cycles per block= **2^{19}**×8= **2^{22}cycles.(0,5p)**
- **time needed to process all the data contained in memory= number of all cycles× time for one cycle=2^{22}×0.5ns=2.09ms.(0,5p)**

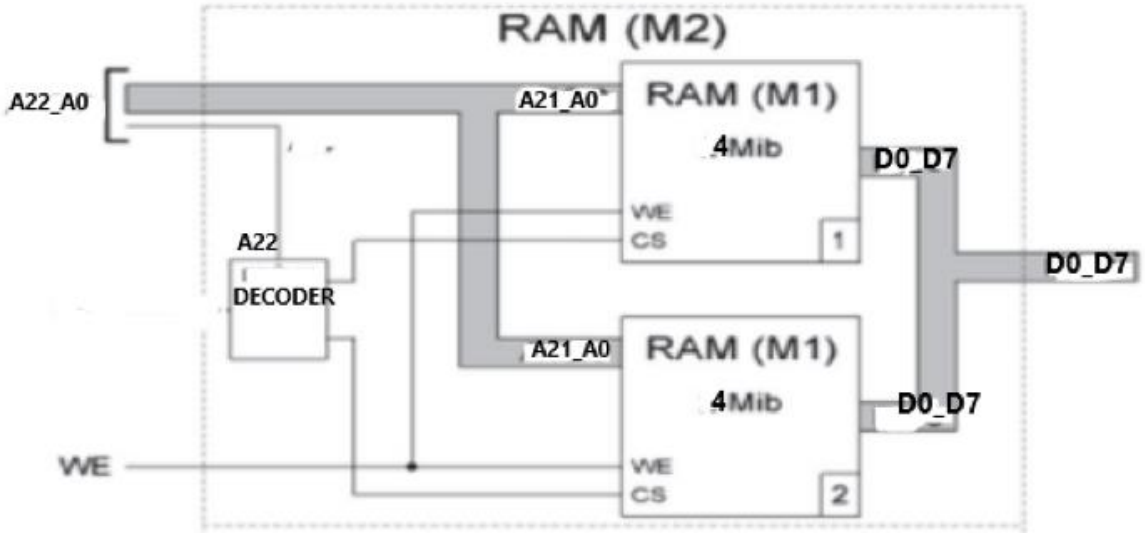
Exercise 3: « 6 pts »

1.the size of the address buses of (M1) is : 22 address lines.**(1p)**

For (M2) is : 23 address lines.**(1p)**

2.we need two chips of (M1) to create (M2). .**(2p)**

3. the wiring diagram. .**(1p)**



5. the addressing intervals: .**(1p)**

		Binary		**Hexadecimal**	
		beginning	**end**	**beginning**	**end**
M2	**$M1_1$**	**A_{22} A_{21}……A_0** **0**00000000000000000000	**A_{22} A_{21}……A_0** **0**11111111111111111111	**000000**	**3FFFFF**
	$M1_2$	**1**00000000000000000000	**1**11111111111111111111	**400000**	**7FFFFF**

Exercise 4: « 5 pts »

1, The program is : .**(4p)**

	MVI	B,00	Clear B reg for quotient
	LXI	H,4500	Initialize HL reg. to 4500H
	MOV	A,M	Transfer dividend to acc.
	INX	H	Increment HL reg. to point next mem. Location.
LOOP	SUB	M	Subtract divisor from dividend
	INR	B	Increment B reg
	JNC	LOOP	Jump to LOOP if result does not yield borrow
	ADD	M	Add divisor to acc.
	DCR	B	Decrement B reg
	INX	H	Increment HL reg. to point next mem. Location.
	MOV	M,A	Transfer the remainder from acc. to memory.
	INX	H	Increment HL reg. to point next mem. Location.
	MOV	M,B	Transfer the quotient from B reg. to memory.
	HLT		Stop the program

2.the programe executes : 8 BIT DIVISION. .(1p)