

Solution du Contrôle (conception des systèmes à microprocesseur)

Questions de cours (5pts)

Pour un microprocesseur Intel 8086.

1-L'unité d'exécution (EU) exécute les instructions contenues dans la file d'attente. (1 point)

2-L'unité d'interface de bus (BIU) recherche les instructions en mémoire et les range dans une file d'attente. (1 point)

3-Le registre IP est le Pointeur d'Instruction : il contient l'adresse de la prochaine instruction à exécuter. (1 point)

4-C'est une mémoire gérée en FIFO, d'une taille de 6 octets qui sert à stocker les instructions à exécuter (1 point)

5- Une constante SIZE = 16. (1 point)

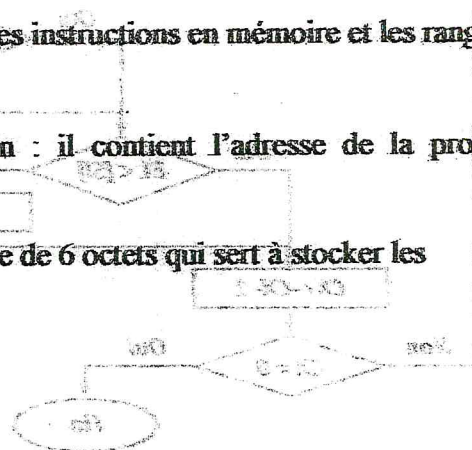
SIZE EQU 16

Une variable MIN.

MIN DB 0

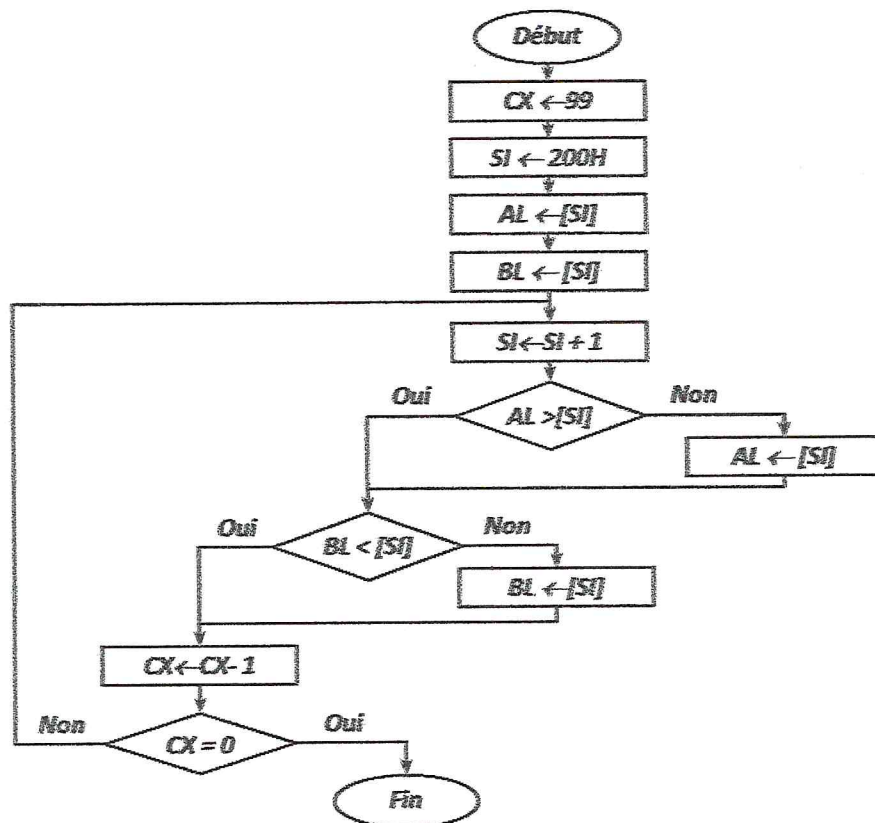
Exercice 1 :(5pts)

	MOV SI, 4000h	(0.5 point)
	MOV DI, 6000h	(0.5 point)
	MOV CX, 30	(0.5 point)
	REPE CMPSB	(1 point)
	JNE Non	(0.5 point)
	MOV AL, 0	(0.5 point)
	JMP Fin	(0.5 point)
Non:	MOV AL, 1	(0.25 point)
Fin:	HLT	(0.25 point)



(1 point)

EXERCICE 2 :



(3.5 point)

2-la taille du tableau traité par ce programme 100 éléments (0.5 point)

3- les valeurs que vont contenir les registres *AL* et *BL* représentent le max et le min respectivement (1 point)

Exercice 3(5pts)

portA equ 3FDAH ; adresses des registres du 8255 (1 point)

portB equ 3FDCH

portC equ 3FDEH

controle equ 3FE0H

mov dx,controle ; initialisation du port B en entrée (0.25)

mov al,82H ; et du port A en sortie (mode 0) : (0.5 point)

out dx,al ; contrôle = 10000010B = 82H (0.25 point)

```

boucle :  mov  dx,portB      ; lecture du port B (0.5 point)
          in   al,dx         (0.25 point)
          and  al,11000000B  ; masquage PB7 et PB6 (0.5 point)
          cmp  al,10000000B  ; test PB6 = 0 et PB7 = 1 (0.5 point)
          jne  faux          ; non -> aller au label "faux" ...
          mov  al,11001000B  ; oui -> mettre PA7, PA6 et PA3 a` 1 (0.5 point)
          jmp  suite          ; et continuer au label "suite"

faux :    mov  al,00000000B  ; ... mettre PA7, PA6 et PA3 a 0 (0.25 point)

suite :   mov  dx,portA      ; écriture du port A(0.5 point)
          out  dx,al
          jmp  boucle         ; retourner lire le port B

```