



CORRIGE DE L'EMD

Corrigé de l'exercice 01 : [04,5 points]

1. (1,0)
 - a) le 8086 est **d'architecture CISC.**
 - b) le 8086 est un processeur **16 bits.**
 - c) Le 8086 peut adresser une mémoire de **1M octets.**
 - d) le 8086 contient **40 broches.**
2. Les différentes étapes du cycle d'exécution d'une instruction sont : (1,0)
 - **Recherche de l'instruction / Décodage / Exécution / Sauvegarde du résultat.**
3. Si un périphérique d'adresse 10H demande une interruption au processeur 8086, l'adresse du registre IP sur la table des vecteurs d'interruptions est **40H** et celle du CS est **42H.** (0,5)
4. Si un PIC16F84 est cadencé par une horloge externe dotée d'un Quartz de 4MHz, le temps de cycle d'une instruction est **1µs.** (0,5)
5. Les circuits qui sont indispensables au fonctionnement du PIC16F84 sont : (0,5)
 - Le circuit de reset
 - L'horloge externe.
6. Les composants qu'on peut trouver dans un microcontrôleur sont : (1,0)

Le microprocesseur, les ports E/S, l'ADC, les temporisateurs, la mémoire programme et la mémoire de donnée.

Corrigé de l'exercice 02 : [02,5 points]

1. Faux. (0,5)
2. Vrai. (0,5)
3. Faux. (0,5)
4. Vrai. (0,5)
5. Vrai. (0,5)

Corrigé de l'exercice 03 : [3,0 points]

1. Le rôle de la file d'attente du microprocesseur 8086 est (0,5)
 - a) d'augmenter la capacité de la mémoire de la mémoire centrale.
 - b) **d'enregistrer les instructions provenant de la mémoire centrale avant d'être exécutés.**
 - c) d'enregistrer les données provenant de la mémoire centrale.
2. Le segment de pile (stack segment) géré par le 8086 (0,5)
 - a) **est organisé en 16 bits.**
 - b) **est une structure de données respectant l'ordre LIFO (Last In First Out).**
 - c) Contient les instructions du programme.
3. L'architecture de Van Neumann (0,5)
 - a) **a un bus commun pour le transfert des instructions et des données.**
 - b) est un type moderne d'architecture informatique.
 - c) permet au processeur de récupérer simultanément les données et les instructions.
4. Dès que l'instruction IRET (Interrupt RETurn) du 8086 est exécutée (0,5)
 - a) **le contenu du registre d'état, CS et IP sont restaurés à partir de la pile.**
 - b) le contenu des registres CS et IP sont restaurés à partir de la pile.

c) le contenu du registre IP est restauré à partir de la pile.

5. La table des vecteurs d'interruptions (0,5)

a) débute à l'adresse 0000H et contient 256 octets.

b) est vectorisée, où chaque vecteur est constitué de 4 octets représentant une adresse logique du type CS : IP.

c) débute à l'adresse 00000H et s'étend sur 1024 octets.

6. Le microcontrôleur PIC16F84 est (0,5)

a) d'architecture RISC, de mémoire de programme de $1k \times 14$ bits et de mémoire de données 68 octets.

b) doté d'une pile à 8 niveaux et de largeur 13 bits.

c) d'architecture RISC, de mémoire de programme de $1k \times 12$ bits et de mémoire de données 64 octets.

Corrigé de l'exercice 04 : [03,0 points]

1. L'instruction utilise l'adressage basé indexé avec déplacement. (0,5)

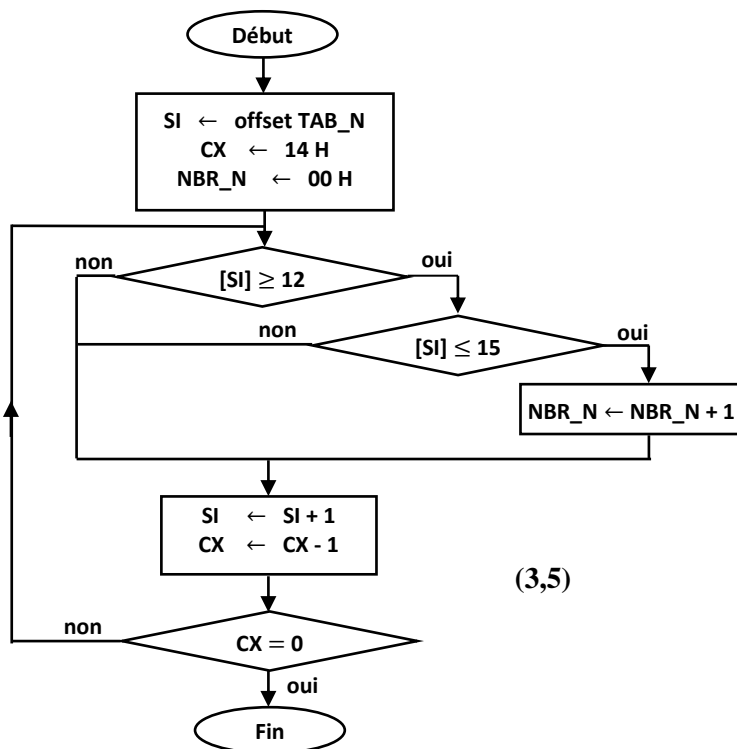
2. $IP = API - CS \times 10H = 20660H - 2000H \times 10H = 660H$ (1,0)

3. L'adresse physique du début de data segment est $DS \times 10H = 30000H$.

L'adresse physique de la fin du data segment est $DS \times 10H + FFFFH = 3FFFFH$. (1,0)

4. En utilisant l'adressage directe on peut écrire l'instruction sous la forme **MOV AL, [862H]** (0,5)

Corrigé de l'exercice 05 : [05,0 points]



```
Name "nul_tab"
ORG 100H
.DATA
TAB_N db 12, 0, 13, 14, 10, 15, 16, 14, 9, 11, 8, 7, 18, 6
NBR_N db 0
.CODE
MOV AX, @DATA
MOV DS, AX
;
LEA SI, TAB_N ; Chargement de l'adresse du début de TAB_N
MOV CX, 14 ; Chargement de la taille du tableau dans CX
encore:
CMP [SI], 12
JNAE mis_pnt
CMP [SI], 15
JNBE mis_pnt
INC NBR_N ; Incrémentation de NBR_N
mis_pnt:
INC SI ; Mise à jour du pointeur
DEC CX ; Mise à jour du compteur
JNZ encore
MOV AH, 4CH ; Fin de programme
INT 21H ; Retour au DOS
ENDP
```

(1,5)

Corrigé de l'exercice 06 : [02,5 points]

1. Les pines RB0..RB3 doivent être configurées en sortie, et Les pines RB4..RB7 en entrée. (0,5)

2. Le programme ci-dessous la configure le PORTB, puis allume les LEDs D4 et D3. (2,0)

```
BSF STATUS, RP0 ; Sélection de la bank1
MOVLW 11110000B ; RB0..RB3 en sortie et RB4..RB7 en entrée
MOVWF TRISB
BCF STATUS, RP0 ; Sélection de la bank0
MOVLW xxxx0011B
MOVWF PORTB
```