

Suggestion de Solution du Contrôle

Questions de cours (6pts)

1-

SAR : Décalage Arithmétique vers la droite ; Cette instruction décale l'opérande de n positions vers la droite (0,25)

SAL : Décalage Arithmétique vers la gauche ; Cette instruction décale l'opérande de n positions vers la gauche (0,25)

CMPS :

- L'opérande est un octet : CMPSB (Compare String Byte) (0,25)

CMPSB compare deux chaînes formées d'une suite d'octets.

- L'opérande est un mot de 16 bits : CMPSW (Compare String Word) (0,25)

CMPSW compare deux chaînes formées d'une suite de mots

CLI (Clear Interrupt Flag) mise à zéro du masque d'interruption IF (0,25)

STI (Set Interrupt Flag) mise à 1 du masque d'interruption IF. (0,25)

2-

Le dépilement engendre deux opérations :

- $destination \leftarrow [SP]$: le sommet de la pile est copié dans destination. (0,25)
- $SP \leftarrow SP + 2$: la donnée du sommet étant retirée, on fait avancer SP pour pointer sur la donnée suivante (0,25)

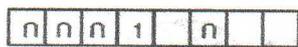
L'empilement s'effectue en deux étapes :

- $SP \leftarrow SP - 2$: on fait reculer SP de deux cases (donnée sur 16 bits), pour qu'il pointe sur le nouveau sommet susceptible de recevoir la donnée à empiler. (0,25)
- $[SP] \leftarrow donnée$: le sommet de la pile reçoit la donnée (0,25)

3- pour mettre à 0 ses 7 bits de poids faible on utilise l'instruction AND avec le masque (0,5)
 10000000

4- Il y a deux types de commandes que l'on peut adresser au PIC : les OCW (Operational Command Words) (0,5) et ICW (Initialization Command Words). (0,5)

ICW1 7 6 5 4 3 2 1 0 (1)

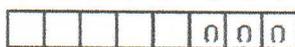


ICW4 est-il requis ? (0 = non, 1 = oui)

1 = un seul PIC existant
 0 = PIC en cascade

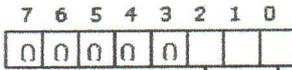
0 = mode attaque
 1 = mode palier

ICW2 7 6 5 4 3 2 1 0 (0,5)



Numéro de l'interruption à exécuter par IRQ0

ICW4



Numéro de la ligne d'interruption par laquelle l'esclave est relié au maître

Exercice 01(4ts)

```

MOV BX, 5           ; charger 5 dans un registre           ; 011
MOV AX, BX         ; copier le nombre dans un 2ème registre
MUL AX             ; x2 = x . x                             ; 011
MOV DL, 2          ; charger 2 dans un registre
MUL DL             ; multiplier x2 par 2                     ; 011
MOV CX, AX         ; sauvegarder 2x2 dans un registre      ; 011
MOV AX, BX         ; charger x dans AX
MOV DL, 3          ; charger 3
MUL DL             ; multiplier x par 3                     ; 011
ADD AX, CX         ; additionner 2x2 + 3x                    ; 011
SUB AX, 1          ; soustraire 1 de 2x2 + 3x                ; 011
MOV [0300H], AX   ; sauvegarder le résultat en mémoire    ; 011
HLT                ; fin du programme

```

Exercice 02 (5pts)

```

MOV CX, 100        ; 011
MOV BX, 0          ; 012
Boucle : MOV AX, CX ; 011
            MOV DL, 3 ; 012
            DIV DL    ; 011
            CMP AH, 0 ; 011
            JNE Suivant ; 011
            ADD BX, CX ; 011
Suivant : LOOP Boucle ; 011
HLT          ; 011

```

- 1- Ce programme calcule la somme des multiples de 3 inférieurs à 100 (11)
- 2- Bx va contenir cette somme (015)

Exercice 03(5pts)

```

MOV AL,8Bh        ; mot de commande (2)
MOV DX,106h       ; adresse du registre de contrôle 0.5 point (011)
OUT DX,AL         ; écriture du mot de commande dans RC (011)
MOV DX,102h       ; adresse du port B 0.5 point (011)
IN AL,DX          ; lecture du contenu du port B dans AL (011)
MOV DX,100h       ; adresse du port A (011)
OUT DX,AL         ; transfert du contenu de AL au port A (011)

```