République Algérienne Démocratique et Populaire Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

USTHB Faculté d'Electronique et d'Informatique Département Informatique Archi 1 LMD2 ACAD

Série d'exercices n°2

<u>NB:</u> Les documents Jeu d'Instructions Assembleur 8086 et Code Machine sont indispensables pour traiter ces exercices

Exercice 1:

Soit le programme assembleur suivant:

```
Tableau DW
              852Fh, 9EA2h, ?
MOV
       BX,
MOV
      AX, Tableau[BX]
      BX, 1
ADD
ADD
      AX, Tableau[BX]
ADD
      BX, 1
      AX, 2FH
SUB
      Tableau[BX], AX
MOV
```

On suppose que le contenu du registre d'état avant l'exécution de la séquence assembleur est :

			OF	DF	IF	TF	SF	ZF		AF		PF		CF
0 (0	0	0	1	0	1	0	0	0	0	0	0	0	0

- 1. Donnez la valeur du registre PSW après l'exécution de chaque instruction de ce programme?
- 2. Indiquer les erreurs glissées dans la séquence.

Exercice 2:

Soit l'opération non signée suivante : W = X * Y.

Ecrire la séquence assembleur permettant d'évaluer l'opération ci-dessus dans le cas :

- 1. X et Y sont sur 16 bits.
- 2. X et Y sont sur 32 bits

Exercice 3:

Soit la séquence d'instructions suivante :

```
; Séquence Assembleur
STI
MOV
       BX,
           23H
MOV
       AL,
            [BX+71H]
MOV
       AL,
MOV
       DX,
           03FDH
MOV
       AX,
            [BX+0C05AH]
ADD
      AX,
            03E4H
SUB
       AX,
            DX
            0ABH
MOV
       AL,
```

- 1. Donner le temps d'exécution de la séquence, sachant que la fréquence du µP est 2,2GHz ?
- 2. Calculer la taille de la séquence assembleur ?
- 3. Donner le code machine de la séquence.
- 4. Si la première instruction de la séquence se trouve à l'adresse mémoire 01FCH:000AH, donner alors l'adresse mémoire de la dernière instruction de la séquence.

Exercice 4:

Soit la séquence d'assembleur suivante :

```
MOV BX, X
ADD Y, BX
SUB CX, Y
MOV BX, OFFSET X
MOV [BX+03A7H], CX
```

Avec : depl(X) = 0121H depl(Y) = 0375 H. (depl : Déplacement)

- 1) Calculer le temps d'exécution de la séquence. Fréquence du µP = 500 MHz
- 2) Donner la taille en octets de la séquence.

Exercice 5:

Si on prend:

Alors:

- 1) Calculer les adresses effectives de branchement pour les instructions suivantes :
 - a) JMP intrasegment direct;
 - b) JMP intrasegment indirect utilisant le mode d'adressage registre et le registre cité est BX ;
 - c) JMP intrasegment indirect utilisant le mode d'adressage registre relatif et le registre cité est BX ;
- 2) Donner le code machine pour chacun des cas.
- 3) Donner le temps d'exécution pour chaque instruction. Fréquence du µP = 500 MHz.

Exercice 6:

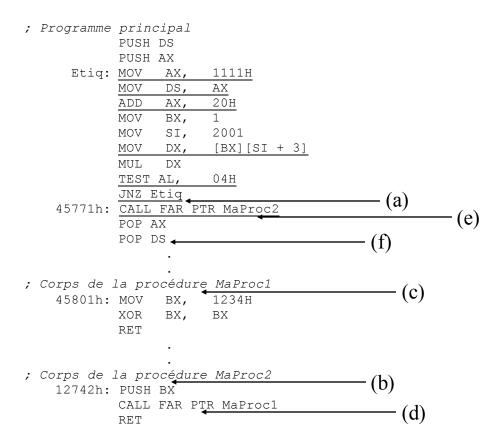
Soit la séquence d'instruction écrite en code machine et se trouvant à l'adresse mémoire 1C530H:

8A 06 00 12 8A 1E 20 1C 88 06 20 1C 88 1E 00 12

- 1) Donner le code assembleur correspondant ;
- 2) Dire que fait cette séquence ;
- 3) Calculer son temps d'exécution. Fréq μ P = 2.4 GHz
- 4) Proposer une écriture en assembleur plus simplifiée de cette séquence.

Exercice 7:

Soit la séquence d'instructions suivante :



Format en code machine de CALL : FF|Mod010R/M|Dep1B|Dep1H|SegB|SegH

Ouestions:

- 1) Donner le code machine (en hexadécimal) des instructions soulignées.
- 2) Calculer le temps d'exécution des instructions se trouvant entre Etiq et JNZ Etiq.
- 3) On suppose que juste avant la première instruction du programme, le contenu des registres suivants est :

$$PSW = 0200H$$
, $SS = 0200H$ et $SP = 0300H$.

Donner alors l'état de la pile aux points a, b, c, d, e et f?

Exercice 7:

Soit le programme suivant:

```
DATA SEGMENT
  X DW OCAOBH
  В
        DB 0C0H
   TAB
        DW 20
                 DUP(?)
DATA ENDS
CODE SEGMENT
  MOV CX, LENGTH TAB
  XOR
        SI, SI
  XOR
        BX, BX
        AX, WORD PTR [BX+1]
  MOV
Enc:MOV
        TAB[SI],
                 AX
         SI , 2
  ADD
   INC
         ΑX
   LOOP
         Enc
CODE ENDS
```

- 1) Calculer les adresses logiques de chaque variable.
- 2) Calculer le nombre d'accès-mémoire généré par l'exécution de ce programme.
- 3) Que fait ce programme ?