

Solutions des Exercices de la série de Travaux Dirigés N° 5

Exercice N° 1 :

On suppose les données suivantes : (IP) = 2B1Ah, (BX) = 637Eh, (SI) = 2A9Bh et Déplacement = 0F7h.

Mode d'adressage	Calcul de l'adresse	Adresse Effective
Adressage basé par BX	[BX]	637E h
Adressage basé par BX avec déplacement	[BX+Dep.]	637E h - 9 h = 6375 h (dep. 8bits) 637E h + 00F7 h = 6475h (dep. 16 bits)
Adressage immédiat	Aucune Adresse	Aucune Adresse
Adressage basé indexé avec déplacement	[BX+SI+Dep.]	637E h + 2A9B h - 9 h = 8E10 h (dep. 8bits) 637E h + 2A9B h + 00F7 h = 8F10 h (dep. 16bits)
Adressage relatif	[IP + Dep.]	2B1A h + 0F7 h = 2B1A h - 9 h = 2B11 h

Exercice N° 2 :

Instructions	Type d'adressage	Instructions	Type d'adressage
MOV CX, 10	Immédiat	JZ Fin	Relatif
MOV DI, 0	Immédiat	ADD AX, [DI+0200h]	Indexé avec déplacement
MOV AX, 0	Immédiat	ADD AX, [BX+0100h]	Basé avec déplacement
DEC CX	Implicite	Fin : MOV [0300h], AX	Direct

Exercice N° 3 :

Instructions	Type d'adressage	Instructions	Type d'adressage
MOV AL, 0DFh	Immédiat	MOV AH, 0CCh	Immédiat
AND AL, 11011111b	Immédiat	OR AH, 20h	Immédiat
INC AL	Implicite	DEC AH	Implicite

Après l'exécution de cette séquence : AL = E0 h = 11100000 b ; AH = EB h = 11101011 b

Exercice N° 4 :

Instruction	Forme	Champs : bin. – hex. (dec.)	Code
MOV AX, 0F2h	1011 w reg data data if w = 1	1011 1 000 (F2 00)h	B8 F2 00
ADD BX, 200	100000 s w mod 0 0 0 r/m data data if s: w = 01	100000 0 1 11 000 011 (C8 00)h	81 C3 C8 00
DEC DX	01001 reg	01001 010	4A
Etiq: SUB CX, [DI-2]	001010 d w mod reg r/m	001010 1 1 01 001 101 (FE)h	2B 4D FE
JMP Etiq	11101011 disp	11101011 (FB)h	EB FB
MOV AX, 0B800h	1011 w reg data data if w = 1	1011 1 000 (00 B8)h	B8 00 B8
MOV DS, AX	10001110 mod 0 reg r/m	10001110 11 0 11 000	8E D8
AND DX, [BX+DI+0100h]	001000 d w mod reg r/m	001000 1 1 10 010 001 (00 01)h	23 91 00 01

Exercice N° 5 :

- If (ax==1) bx = 10; else { bx = 0; cx = 10; }

CMP AX, 1

JE egal

MOV BX, 0

MOV CX, 0

JMP fin

egal: MOV BX, 10

fin: HLT

- `For (cx=0; cx<=10; cx++) { bx = bx + cx; ax=ax+5; }`

1. Sans l'utilisation de l'instruction LOOP

```
MOV CX, 0
Boucle: ADD BX, CX
ADD AX, 5
INC CX
CMP CX, 11
JNE Boucle
HLT
```

2. Avec l'instruction LOOP

```
MOV CX, 11
MOV DX, 0
Boucle: ADD BX, DX
ADD AX, 5
INC DX
LOOP Boucle
HLT
```

- `while (bx > 0) { bx = bx - 1; }`

```
Boucle: CMP BX, 0
JLE Fin
DEC BX
JMP Boucle
Fin: HLT
```

- `do { cx=cx+1 ; bx=bx*2 } while (cx<=15)`

```
Boucle: INC CX
MOV AL, 2
MUL BL
MOV BX, AX
CMP CX, 16
JNE Boucle
HLT
```

- `if ((ax>bx) && (cx <= dx)) { ax=ax-bx ; cx=cx+dx ; } -- Pour des nombres signés.`

```
CMP AX, BX
JLE Fin
CMP CX, DX
JG Fin
SUB AX, BX
ADD CX, DX
Fin: HLT
```

- Switch (ax) {case 1: bx=3 ; break ; case 2: bx=5 ; break ; default : bx=1 ; }
 CMP AX, 1
 JE Case1
 CMP AX, 2
 JE Case2
 Default: MOV BX, 1 ; Correspond à Default
 JMP Break
 Case1: MOV BX, 3
 JMP Break
 Case2: MOV BX, 5
 Break: HLT
- if ((ax>bx) || (cx <= dx)) { bx=bx+cx ; cx=cx*dl } -- Pour des nombres non-signés.
 CMP AX, BX
 JA Valide ; Correspond à une condition vérifiée
 CMP CX, DX
 JBE Valide ; Correspond à la deuxième condition vérifiée
 JMP Fin ; Correspond à aucune condition vérifiée
 Valide: ADD BX, CX
 MOV AL, DI
 MUL CI
 MOV CX, AX
 Fin : HLT

Exercice N° 6 :

1- Déterminer la valeur du bit N° 3 du registre AX

En utilisant l'opération de masquage par une opération AND

AND AX, 0008 h ; (0000 0000 0000 1000 b) Masquage de tous les bits sauf le bit N° 3
 CMP AX, 0008 h ; (0000 0000 0000 1000 b) Test si le bit N° 3 est égal à 1
 JE Etq

L'opération de masquage peut être également effectuée par une opération OU

OR AX, 0FFF7h ; (1111 1111 1111 0111 b) Masquage de tous les bits sauf le bit N° 3
 CMP AX, 0FFF7h ; (1111 1111 1111 0111 b) Test si le bit N° 3 est égal à 0
 JNE Etq

En utilisant 4 décalages successifs à droite :

SHR AX,4
 JC Etq

2- Mettre le bit N°1 du registre DH à 1, le bit N° 3 à 0 sans changer les autres bits du registre.

OR DH, 02 h ; (0000 0010 b) Forcer le bit N°1 à 1
 AND DH, F7 h ; (1111 0111 b) Forcer le bit N°3 à 0

3- Vérifier si les bits du registre AX sont : A0=1, A3=0, A5=0, A8=1 et A14=1.

AND AX, 4129h ; (0100 0001 0010 1001 b) Masquer tous les bits sauf les bits 0, 3, 5, 8 et 14.
 CMP AX, 4101h ; (0100 0001 0000 0001 b) Comparer par rapport à la bonne combinaison

Ou bien en utilisant un masquage par une opération OR :

OR AX, 0BED6h ; (1011 1110 1101 0110 b) Masquer tous les bits sauf les bits 0, 3, 5, 8 et 14.
 CMP AX, 0FFD7h ; (1111 1111 1101 0111 b) Comparer par rapport à la bonne combinaison

Exercice N° 7 :

MOV AL, [1100h]
ADD AL, [1101h]
JS Negatif
JZ Nul
MOV ES:[1102 h], AL
JMP fin

Negatif : MOV ES:[1103 h], AL
JMP Fin

Nul : MOV ES:[1104 h], AL

Fin : HLT

Exercice N° 8 :

$\left(\begin{array}{l} 0\ X\ X\ X\ X\ X\ X\ X\ X\ X\ X \\ X\ 0\ X\ X\ X\ X\ X\ X\ X\ X\ X \\ X\ X\ 0\ X\ X\ X\ X\ X\ X\ X\ X \\ X\ X\ X\ 0\ X\ X\ X\ X\ X\ X\ X \\ X\ X\ X\ X\ 0\ X\ X\ X\ X\ X\ X \\ X\ X\ X\ X\ X\ 0\ X\ X\ X\ X\ X \\ X\ X\ X\ X\ X\ X\ 0\ X\ X\ X\ X \\ X\ X\ X\ X\ X\ X\ X\ 0\ X\ X\ X \\ X\ X\ X\ X\ X\ X\ X\ X\ 0\ X \\ X\ X\ X\ X\ X\ X\ X\ X\ X\ 0 \end{array} \right)$

MOV AX, 0200h

MOV DS, AX

MOV CL, 10

MOV SI, 0

MOV BX, 0

Etiquette: MOV [BX+SI+1000h],0

ADD BX, 10

INC SI

LOOP Etiquette

HLT