TP2 – Gestion et traitement des interruptions :

## Objectifs du TP :

* Comprendre les mécanismes d’interruption
* Visualiser la table des vecteurs d’interruptions
* Déclenchement d’une Interruption logicielle IT DOS : codage en assembleur

Exercice 1 : Interruptions de votre machine

Etant en quantité limitée, les IRQ ont longtemps été la cause de conflits matériels, car une IRQ ne peut être partagée que sous des conditions très strictes. Afin de vous faire une idée de l'utilisation des interruptions vous irez sur votre machine dans les **Propriétés systèmes,** puis en ouvrant le **Gestionnaire de périphériques**. Recherchez par exemple votre clavier dans la liste, puis choisissez **Propriétés** et regardez l'onglet **Ressources**.

Exercice 2 : Table des vecteurs d’interruptions de votre machine

On veut visualiser le contenu de la table des vecteurs d’IT.  
Prendre l’exemple de l’interruption 21h interruption du DOS.  
Visualisez la mémoire à l’adresse 0000:0084, la valeur du vecteur d’IT.   
(Exemple sur mon PC je vois : 00 02 00 F4 ce qui correspond à l’adresse F400:0200)

Allez à cette adresse (en mettant le registre CS à la valeur F400 et IP à la valeur 0200) et vérifiez bien que le code de la procédure de l’interruption logicielle 21h. Le seul moyen de vérifier est de constater qu’elle se termine bien par un IRET.

Exercice 3 : Déclenchement d’une interruption logicielle

Rappel **:**

Lorsqu’une IT est déclenchée le micro-processeur :

* empile les indicateurs du registre d’état
* désactive les IT matérielles
* empile CS
* empile IP

Il exécute ensuite la routine d’interruption. Cette routine se termine par un IRET qui a pour effet de :

• dépiler IP

• dépiler CS

• dépiler les indicateurs

Pour déclencher une interruption en assembleur il suffit d’invoquer le numéro de l’interruption par l’instruction INT.

Affichage de l’heure courante du PC :

Vous allez invoquer dans une boucle, l’interruption qui donne l’heure courante de votre PC et vous afficherez ce résultat avec le format suivant :

*L heure actuelle est : 12h01m36s11*

*L heure actuelle est : 12h01m36s43*

*L heure actuelle est : 12h01m37s15*

**Interruption** de l’horloge est INT 21h

**Entrée** : AH=2Ch

**Sortie** : CH = hour CL = minute DH = second DL = 1/100 seconds  
  
Ecrivez le code assembleur correspondant.   
Vous constaterez que les valeurs affichées ne sont pas qu format HH :MM :SS :CC.   
Vous écrirez donc une procédure qui permet la conversion des valeurs retournées par l’interruption en des valeurs affichables :  
Quelle est la durée moyenne d’exécution d’un passage dans la boucle ?

Exercice 3 : création d’une routine d’ Interruption logicielle

Vous allez invoquer votre propre routine d’interruption en utilisant un numéro d’interruption laissé libre pour l’utilisateur.   
Vous choisirez l’interruption n° 150.   
Le programme main met en place la routine d’IT dont le numéro sera 150 puis fait une boucle qui teste la valeur d’un compteur qui mémorise le nombre d’appel a l’IT avant de sortir.

Il faut rappeler les points suivants :

les fonctions du DOS qui permettent d’installer une procédure d’IT et de récupérer l’adresse d’un handler d’IT

**INT 21h**

**Fonction 25h** qui installe une procédure d’IT dans la table des vecteurs d’IT.

*entrée* :

AH <- numéro de la fonction

AL <- numéro de l’IT

DX <- offset de la procédure d’IT

DS <- segment de la procédure d’IT

*sortie :*

rien

**INT 21h**

**Fonction 35h** qui récupère l’adresse d’une procédure d’IT dans la table des vecteurs d’interruption

entrée :

AH <- numéro de la fonction

AL <- numéro de l’IT

sortie :

BX -> offset de la procédure d’IT

ES -> Segment de la procédure d’IT

Voici le pseudo-code des trois procédures à coder :

**Main**

Appel de la procédure Installation\_IT-150

Compteur <- 0

iterer

imprime M

invoque l’interruption numéro 150

teste un compteur

sort si compteur = 10

al <- compteur

affiche al

fin iterer

**Votre\_procedure\_dIT\_150**

compteur ++

imprime ‘I’

**Installation\_IT\_150**

AH<-25h

AL <- 150

DX <- offset de votre\_procédure\_dIT\_150

DS <- segment de votre\_procédure\_dIT\_150

Invoquer l’interruption 21h

Ne pas oublier qu’une procédure d’IT doit se terminer par IRET.  
Pour récupérer la valeur de l’offset d’une procédure en assembleur on utilise d’instruction OFFSET.  
Pour récupérer la valeur de segment d’une procédure en assembleur il suffit d’utiliser le nom du segment de la procédure (voir solution).

; TP4\_2.asm

PILE SEGMENT STACK

DB 256 DUP(?)

PILE ENDS

DONNEES SEGMENT

compteur db 0

DONNEES ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DONNEES

extrn prbyte : far

extrn crout : far

extrn cout : far

IT150 PROC FAR

PUSH DS

MOV AX,0

PUSH AX

MOV AX,DONNEES

MOV DS,AX

;

...

RET

IT150 ENDP

VOTRE\_PROCEDURE\_DIT\_150 PROC NEAR

...

iret

VOTRE\_PROCEDURE\_DIT\_150 ENDP

INSTALLATION\_IT\_150 PROC NEAR

...

ret

INSTALLATION\_IT\_150 ENDP

CODE ENDS

END IT150