

# Chapitre 2

## ARCHITECTURES DES ORDINATEURS II

Annexe :

**un peu de pratique sur les interruptions !**

EQUIPE PEDAGOGIQUE :

Chargé de cours : Mr M.S AYACHE  
Chargée de TP : Mme S. BOUCHENE  
Chargée de TD : Melle I. SETITRA

# PLAN

## 1. VERIFICATION DU COMPORTEMENT DU PROCESSEUR LORS D'UNE INTERRUPTION EN UTILISANT LE DEBUG

- 1.1 Sauvegarde du contexte du  $\mu P$  avant de passer à la routine d'IT
- 1.2 Préparation du  $\mu P$  pour passer à la routine d'IT

## 2. OUTILS PRATIQUES DE PROGRAMMATION DES INTERRUPTIONS

# VERIFICATION DU COMPORTEMENT DU PROCESSEUR LORS D'UNE INTERRUPTION

DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
D:\>debug
-a
073F:0100 int 48
073F:0102
-r
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100 NV UP EI PL NZ NA PO NC
073F:0100 CD48 INT 48
-t
AX=0000 BX=0000 CX=0000 DX=0000 SP=00F7 BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=F000 IP=1060 NV UP DI PL NZ NA PO NC
F000:1060 FE38 ??? [BX+SI] DS:0000=CD
-d 073f:00f7
073F:00F0 02-01 3F 07 02 F2 00 00 57 ..?....W
073F:0100 CD 48 00 00 00 00 00 00 00-00 00 00 00 00 AE FE .H.....
073F:0110 00 F0 93 74 00 00 B9 F4-47 0B 74 00 34 00 2E 07 ...t....G.t.4...
073F:0120 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0130 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0140 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0150 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0160 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0170 00 00 00 00 00 00 00 00 .....
-
```

Sauvegarde du contexte du  $\mu P$   
avant de passer à la routine d'IT

```
DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

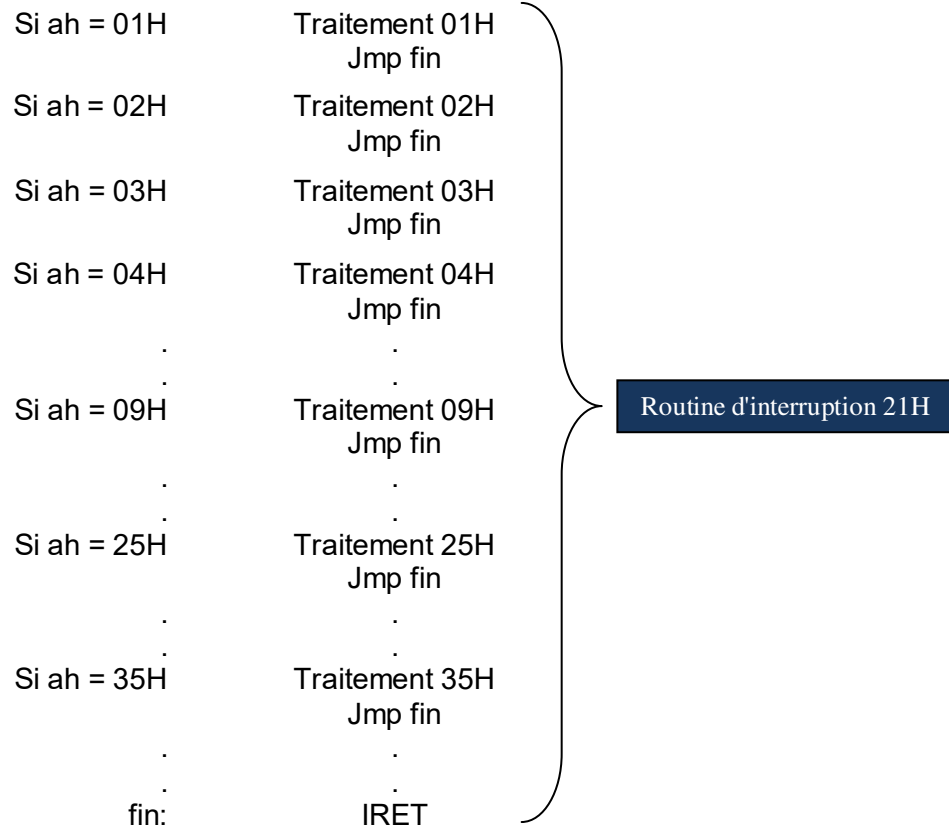
D:\>debug
-a
073F:0100 int 48
073F:0102
-r
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100  NU UP EI PL NZ NA PO NC
073F:0100 CD48          INT     48
-t
AX=0000 BX=0000 CX=0000 DX=0000 SP=00F7 BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=F000 IP=1060  NU UP DI PL NZ NA PO NC
F000:1060 FE38          ???    [BX+SI]          DS:0000=CD
-d 0000:0120
0000:0120 60 10 00 F0 60 10 00 F0 60 10 00 F0 00 16 00 F0  \. . . . .
0000:0130 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00 F0  \. . . . .
0000:0140 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00 F0  \. . . . .
0000:0150 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00 F0  \. . . . .
0000:0160 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00 F0  \. . . . .
0000:0170 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00 F0  \. . . . .
0000:0180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  \. . . . .
0000:0190 00 00 00 00 00 00 00 00 00 60 10 00 F0 04 00 41 C8  \. . . . .A.
```

# PRESENTATION DE QUELQUES INTERRUPTIONS UTILES AUX TD & TP

## CAS DE L'INTERRUPTION 21H :

c'est une très longue routine d'interruption parfois appelée INTERRUPTION SYSTEME , elle offre plusieurs services. Chaque service est une fonction dont le numéro doit être au préalable envoyé dans **le registre AH** avant l'appel à cette interruption.

## structure de la routine d'interruption 21H appelée avec **INT 21H**



```
; programme appelant  
.  
.  
.  
mov ah, #fonction  
int 21h  
.  
.  
.
```

## QUELQUES FONCTIONS DE L'INTERRUPTION 21H

---

AH=01H	LECTURE D'UN CARACTERE AU CLAVIER AVEC ECHO
AH=02H	ECRITURE D'UN CARACTERE A L'ECRAN
AH=05H	ENVOI D'UN CARACTERE A L'IMPRIMANTE
AH=08H	LECTURE D'UN CARACTERE SANS ECHO AU CLAVIER
AH=09H	AFFICHAGE D'UNE CHAINE DE CARACTERS A L'ECRAN
AH=0AH	LECTURE D'UNE CHAINE DE CARACTERS AU CLAVIER
AH=25H	INSTALLATION D'UN VECTEUR D'INTERRUPTION
AH=2AH	LECTURE DE LA DATE SYSTEME
AH=35H	LECTURE D'UN VECTEUR D'INTERRUPTION

---

### 1. AH=01H

#### LECTURE D'UN CARACTERE AU CLAVIER AVEC ECHO

ENTREE : AH=01H

RETOUR : AL=CARACTERE LU

### 2. AH=02H

#### ECRITURE D'UN CARACTERE A L'ECRAN

ENTREE : AH=02H , DL=CARACTERE A ECRIRE

RETOUR : RIEN

```
; programme
.
.
.
mov ah , 1
int 21h
.
.
.
```

### 3. AH=08H

#### LECTURE D'UN CARACTERE SANS ECHO AU CLAVIER

ENTREE : AH=08H  
RETOUR : AL=CARACTERE LU

### 4. AH=09H

#### AFFICHAGE D'UNE CHAINE DE CARACTERES A L'ECRAN

ENTREE : AH=09H  
DS:DX ADRESSE DU 1<sup>er</sup> CARACTERE DE LA CHAINE A AFFICHER  
(la chaîne de caractères doit se terminer par "\$")  
RETOUR : RIEN

### 5. AH=0AH

#### LECTURE D'UNE CHAINE DE CARACTERS AU CLAVIER

ENTREE : AH=0AH ,  
DS:DX ADRESSE DU PREMIER OCTET DE LA CHAINE  
(la case mémoire DS:DX doit être initialisée à la taille max du buffer de réception)  
DS :DX+1 reçoit à la fin de la lecture le nombre de caractères réellement reçu,  
A partir de DS :DX+2 la chaîne reçue  
RETOUR : RIEN

### 6. AH=2AH

#### LECTURE DE LA DATE SYSTEME

ENTREE : AH=2AH  
RETOUR : CX=ANNEE ( 0=1980..119=2099)  
DH=MOIS (1=JANVIER,2=FEVRIER..)  
DL=JOUR (1...31)

```
; programme
data      segment
msg       db 'Bonjour Archi2$'
data      ends

code      segment
assume cs: code , ds, data
start:    mov ax, data
          mov ds, ax.
          .
          .
          mov ax, seg msg
          mov ds, ax
          mov dx , offset msg
          mov ah , 9
          int  21h
          .
          .
code      ends
end start
```

## 7. AH=25H

### INSTALLATION D'UN VECTEUR D'INTERRUPTION

ENTREE : AH=25H  
AL= NUMERO D'INTERRUPTION  
DS:DX ADRESSE DE LA PREMIERE INSTRUCTION DE LA ROUTINE D'INTERRUPTION  
RETOUR : RIEN

## 8. AH=35H

### LECTURE D'UN VECTEUR D'INTERRUPTION

ENTREE : AH=35H  
AL= NUMERO D'INTERRUPTION  
RETOUR : ES:BX ADRESSE DE DEBUT DE LA ROUTINE D'INTERRUPTION

## Autres interruptions utiles pour les TPs

### POSITIONNER LE CURSEUR :

```
mov ah,2 ;numéro de fonction
mov bh,0
mov dh,lig ;ligne de l'écran
mov dl,col ;colonne de l'écran
int 10h
```

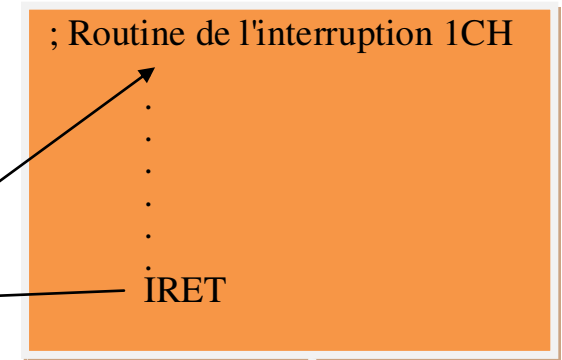
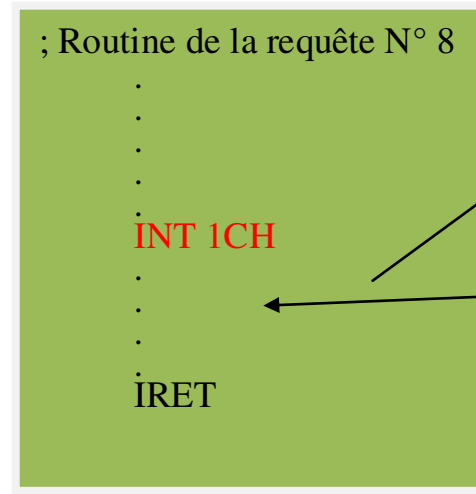
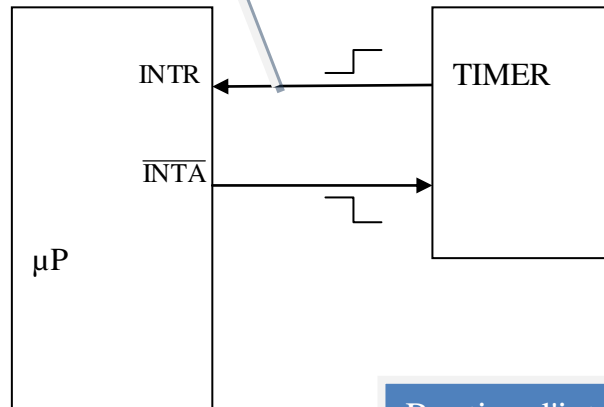
### EFFACER ECRAN :

```
mov ax , 3
int 10h
```



## INTERRUPTION 1CH :

la requête N° 8 envoyée  
toutes les 55 msec  
(18,2 requêtes/sec)



Routine d'interruption  
appelée toutes les  
55msec

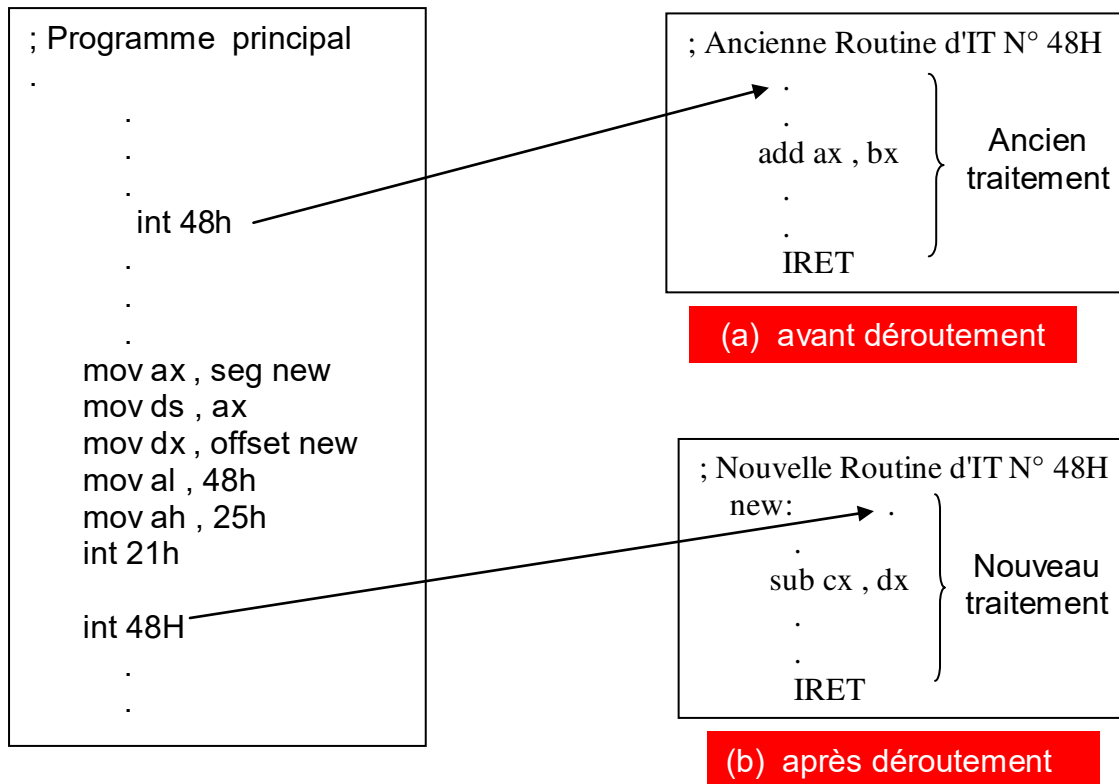
cette routine est  
également appelée toutes  
les 55 msec

**1/55 msec = 18,2 requêtes/sec**

Le code de la routine d'interruption 1CH est sans aucun intérêt et donc prévu pour un éventuel déroutement afin d'implémenter des quantums de temps indépendamment du processeur

## DEROUTEMENT D'UNE INTERRUPTION

Le déroutement de l'interruption numéro N consiste à conserver le numéro de l'interruption N et donc la conservation de l'adresse du vecteur N i.e  $4 \times N$  et le changement du contenu de ce vecteur i.e  $(4 \times N)$  et  $(4 \times N + 2)$ . En d'autres termes ceci consiste à conserver le numéro de l'interruption et changer son traitement.



**Rappel:**

**AH=25H , INT 21H**

**INSTALLATION D'UN VECTEUR D'INTERRUPTION**

**ENTREE :**

AH=25H

DS:DX ADRESSE DE LA ROUTINE D'INTERRUPTION

AL= NUMERO DE L'INTERRUPTION

**RETOUR : RIEN**