

Remarque : Si les modes d'entrée et de sortie ne sont pas définis, le moniteur force l'entrée en hexadécimal et la sortie en mode mnémotechnique.

```

ENTREE  #N1      (nombre:1 octet)
ENTREE  #DH      (sortie numerique hexadecimale)
ENTREE  #BC00/ 34      -
ENTREE  #A       (sortie en code ASCII)
ENTREE  #BC00 / 4      - ( le code ASCII de 4 =34)
ENTREE  #N2      (nombre octets:2)
ENTREE  #BC00/ 347E    -
ENTREE  1
ENTREE  1
ENTREE  1
ENTREE  1 1 1
ENTREE  START/ 347E      1836
ENTREE  #START/ 1836      347E
ENTREE  #BC00 / 347E
ENTREE  #= /34 (le moniteur pointe sur l'adresse BC00
ENTREE  #PC/ 0000      BC00 (PC est charge a BC00
ENTREE  #PC/  BC00

```

MODE NUMERIQUE

Commande: Le moniteur en état « attente de commande » « # »

```

Tapez : N1  ENTREE
ou      : N2  ENTREE
ou      : N   ENTREE

```

Effet : La commande N[n] (MODE NUMERIQUE) force la sortie du moniteur en mode numérique.

Le suffixe [n] {1 ou 2} représente le nombre d'octets à examiner. Si [n] est omis dans la commande, le moniteur force n = 1.

Exemple :

```

ENTREE  #IH
ENTREE  #DH
ENTREE  #N1
ENTREE  #BC00/ 10
ENTREE  #N2
ENTREE  #BC00/ 10BE
ENTREE  #N
ENTREE  #BC00/ 10
ENTREE

```

OUTPUT : sortie

Commande : Le moniteur en état « attente de commande » « # »

Tapez : OD **ENTREE**

ou OH **ENTREE**

Effet : La commande O {H ou D} (OUTPUT) spécifie le mode numérique de sortie du moniteur

Les codes admis sont :

 DÉCIMAL : D

 HEXADÉCIMAL : H

Si la commande n'a pas été précisée, le moniteur forcera le code **hexadécimal** par défaut.

Sans suffixe, la valeur affichée est exprimée en hexadécimal.

Le code décimal est spécifié par le suffixe T.

Exemple :

	#N2	
ENTREE		
	#BC00/	347E
ENTREE		
	#0D	
ENTREE		
	#BC00/	13438T
ENTREE		
	#0H	
ENTREE		
	#BC00/	347E
ENTREE		

MODE ASCII

Commande : Le moniteur en état « attente de commande » « # »

Tapez : A **ENTREE**

Effet : La commande A (MODE ASCII) force la sortie du moniteur en mode caractère ASCII.

Le moniteur affichera le caractère représenté, en code ASCII, par le contenu de l'octet spécifié.

Si la valeur de l'octet n'est pas comprise entre 32 et 127 en décimal ou 20 et 7F en hexadécimal, représentant les caractères dits « imprimables », le moniteur affichera un point.

Exemple :

	#A		
ENTREE			
	#BC00/	A	(34)
ENTREE			
	#BC01/	-	(7E)
ENTREE			
	#BC02/	.	(BD)
ENTREE			
	#BC03/	s	(E8)
ENTREE			

MODE MNEMONIQUE

Commande : Le moniteur en état « attente de commande » « # »

Tapez : M **ENTREE**

Effet : La commande M (MODE MNEMONIQUE) permet de reconstituer à la sortie du moniteur les instructions en langage symbolique du programme source. Cette opération est aussi appelée « désassemblage ».

Pour cela, le moniteur examine la mémoire par paquets de un à cinq octets suivant l'instruction.

L'adresse de la commande doit pointer sur le premier octet d'une instruction, sinon le désassemblage du programme objet n'aurait aucun sens.

Si le moniteur ne peut interpréter le contenu d'un octet, il affichera un point d'interrogation « ? »

Si aucun mode de sortie du moniteur n'est demandé, le système forcera le mode mnémonique.

Exemple :

```
#M
ENTREE
#BC00/ PSMS 0,Y,X,DP,B,A
1
#BC02/ JSR >INIT
1
#BC03/ LDA >PORTC
1
#BC05/ ORA #1
1
#BC0A/ STA >PORTC
```

INDIRECTION

Commande : Tapez : **■** après une valeur d'adresse dont vous souhaitez connaître le contenu.

Effet : La commande **■** (INDIRECTION) permet d'introduire un niveau d'indirection dans l'examen du contenu des mémoires.

La commande INDIRECTION utilise le contenu de la sortie courante pour adresser la nouvelle mémoire.

Si la sortie est en mode numérique sur un octet ou en mode ASCII, le moniteur complètera l'adresse avec l'octet suivant.

En sortie sur deux octets, la valeur représente l'adresse à examiner.

En mode mnémonique, le moniteur pointera l'adresse indiquée dans la ligne courante. Si aucune adresse n'est présente ou si l'instruction ne touche pas au PC, la commande INDIRECTION est sans effet et répète la ligne courante.

Cette commande est très utile pour examiner la destination des branchements, par exemple : JSR, JMP, BRA, etc.

Remarque : si l'instruction touche au PC, la commande **■** visualise le contenu de l'adresse du PC après modification par cette instruction.

Exemples :

```
#N2 (mode numérique 2 octets)
ENTREE
#BC00/ 347E
1
347E / 2016
1
2016 / 9E90
ENTREE
```

```
#N1 (mode numérique 1 octet)
ENTREE
#BC00/ 34
1
34 / 20
1
20 / 9E
ENTREE
```

#M (mode mnémonique)

ENTREE

#BC00/ PSHS U,Y,X,DP,B,A

■

0F7D5/ PSHS U,Y,X,DP,B,A

(l'indirection est sans effet la ligne courante ne contient pas d'adresse)

■

0BC02/ JSR INIT

■

INIT / JMP >0F7D5

■

0F7D5/ PSHS DP,B,A

ENTREE

#X/ BC00 (chargement de X)

ENTREE

(utilisation du contenu de X comme adresse d'indirection)

#X/ 0F7D5

■

0F7D5/ PSHS DP,B,A

ENTREE

MODE CALCULATEUR

Commande : Le moniteur en état « attente de commande » « # »

Tapez : <expression> =

A droite du signe « = », le résultat du calcul de l'expression sera affiché.

Effet : Le **MODE CALCULATEUR** du **MONITEUR** permet d'évaluer une expression et d'afficher le résultat.

Toutes les expressions admises par l'assembleur sont évaluables par le moniteur.

Les étiquettes présentes dans l'expression à évaluer doivent être également présentes dans la table des symboles issue du dernier assemblage.

L'évaluation d'une expression peut être utilisée pour examiner la mémoire correspondante.

Les résultats négatifs sont exprimés en valeur complémentaire plus 1. Cette manière de représenter les nombres est appelée « complément à 2 »

Par exemple :

1	0001
0	0000
-1	FFFF

#0H (sortie en hexadécimal)

ENTREE

#IH (entrée en hexadécimal)

ENTREE

#BC00+18= 0BC18

ENTREE

#ENDMEM-START= 100

ENTREE

#0D (sortie en décimal)

ENTREE

#ID (entrée en décimal)

ENTREE

#1345+6640= 7985

ENTREE

BC00H+03/ 0BC03

GO : lancer

Commande : Le moniteur en état « attente de commande » « # »

Tapez : G <expression> **ENTREE**

Effet : La commande G <expression> (GO) permet de lancer l'exécution d'un programme à partir de l'adresse définie par l'expression de la commande.

La commande moniteur GO est équivalente à un saut à une adresse (instruction JMP).

Les étiquettes utilisées dans l'évaluation de l'expression doivent être présentes dans la table des symboles issue du dernier assemblage.

Si en cours d'exécution du programme le système rencontre une interruption logicielle (SWI), il affichera le message :

« 8 BRK 0 (adresse) » et arrêtera l'exécution du programme.

Ce message (indice 8) indique qu'un point d'arrêt a été positionné par l'utilisateur dans son programme. Sous contrôle du moniteur, on ne peut placer que huit points d'arrêt numérotés de 0 à 7. SWI peut être utilisé pour retourner à la fin de l'exécution du programme utilisateur, sous contrôle du moniteur.

Exemple :

#GSTART (exécution du programme « MOIRAGE »)

ENTREE



BREAK : arrêt

Commande : Le moniteur en état « attente de commande » « # »

Tapez : K <expression> **ENTREE**

Effet : La commande K <expression> (BREAK) positionne un point d'arrêt sur l'adresse spécifiée par l'expression de la commande. L'utilisateur a la possibilité de positionner huit points d'arrêt numérotés de 0 à 7 dans l'ordre de la programmation.

Le point numéro 8 est affiché par le moniteur quand il rencontre l'interruption logicielle (SWI) dans le programme utilisateur.

L'adresse de positionnement des points d'arrêt peut être définie par une constante, une étiquette ou une expression.

Les étiquettes utilisées doivent être présentes dans la table des symboles issue du dernier assemblage.

Un retour sous contrôle de l'éditeur ou MENU efface les points d'arrêt programmés précédemment.

Un point d'arrêt qui n'est pas positionné sur le premier octet d'une instruction va détruire un octet de cette instruction et provoquer des effets imprévisibles à l'exécution !!

IL NE FAUT JAMAIS FAIRE CELA !!!

La technique des points d'arrêt consiste à insérer une interruption logicielle SWI sur chaque adresse pointée par un point d'arrêt.

On ne peut donc programmer un point d'arrêt dans le SYSTEME, qui est en mémoire ROM.

Exemple :

#KBC05

ENTREE

#GSTART

ENTREE

0 BRK 0 0BC05

#

CONTINUE : continuer

Commande : Le moniteur en état « attente de commande » « # » après affichage d'un point d'arrêt.

Tapez : C [n] **ENTREE**

Effet : La commande C [n] (CONTINUE) permet de continuer l'exécution d'un programme suspendue par un point d'arrêt.

[n] indique le nombre de passages par le point d'arrêt qu'il faut faire pour que le programme y soit suspendu. A chaque point d'arrêt est associé un "nombre de passage" qui est initialisé à 1 par l'instruction K.

Si le paramètre n est omis, 1 est pris par défaut et le programme s'exécutera jusqu'au prochain point d'arrêt défini par l'utilisateur ou le moniteur.

Si le dernier point d'arrêt a été positionné sur un branchement vers une zone de programme en ROM, le MONITEUR affichera le message « Can't continue » (ne peut pas continuer) (voir commande BREAK)

Remarque : la commande G (GO) ne remet pas à 1 les nombres de passages associés aux BREAKPOINTS

Exemple :

```
#KM2
ENTREE
#GSTART
ENTREE
0 BRK 0 M2
#C
ENTREE
0 BRK 0 M2
#
```

BREAKPOINT : point d'arrêt

Commande : Le moniteur en état « attente de commande » « # »

Tapez : B **ENTREE**

Effet : La commande B (BREAKPOINT) affiche les points d'arrêt programmés ainsi que leur adresse.

L'affichage est effectué dans l'ordre croissant des numéros.

Exemple :

```
#KBC02
ENTREE
#KBC08
ENTREE
#KBC0D
ENTREE
#KBC0A
ENTREE
#B
0 BRK 0 0BC02
1 BRK 0 0BC08
2 BRK 0 0BC0D
3 BRK 0 0BC0A
#
```

REGISTRE : registre

Commande : Le moniteur en état « attente de commande » « # »

Tapez : R **ENTREE**

Effet : La commande R(REGISTRE) permet d'afficher le contenu des registres internes du microprocesseur 6809.

La ligne 0 de l'écran, sous contrôle du moniteur, spécifie le nom des registres.

Le moniteur force l'affichage des registres en hexadécimal quel que soit le mode de sortie.

Exemple :

```
PC  A  B  DP  CC  X  Y  U  S
#
#KBC05
ENTREE
#GSTART
ENTREE
0 BRK 0 0BC05
#R
ENTREE
BC05 00 00 00 B4 0000 0000 0000 62FA
#
```

YANK : supprimer

Commande : Le moniteur en état « attente de commande » « # »

Tapez : Y[n] **ENTREE**

ou : Y **ENTREE**

Effet : La commande Y [n] (YANK) supprime le point d'arrêt numéroté n.

Si [n] est omis dans la commande, tous les points d'arrêt programmés précédemment seront supprimés.

Exemple :

```
#KM2
ENTREE
#KM1
ENTREE
#KM0
ENTREE
#KBC21
ENTREE
#B
ENTREE
0 BRK 0 M2
1 BRK 0 M1
2 BRK 0 M0
3 BRK 0 0BC21

#Y2 (supprime le point d'arrêt n° 2)
ENTREE
#B
ENTREE
0 BRK 0 M2
1 BRK 0 M1
3 BRK 0 0BC21

#Y (supprime tous les points d'arrêt)
ENTREE
#B
ENTREE
# (plus de points d'arrêt)
```

```
BC00 34 7E BD E8 00 B6 E7 C3 4 .....
BC03 8A 01 B7 E7 C3 2E 40 00 .....@.
BC18 81 31 3F 26 FA 10 8E 00 .1?&...
BC20 14 CC AA AA ED 81 31 3F .....1?
#
```


TRACE

Commande : Le moniteur en état « attente de commande » « # » après affichage d'un point d'arrêt.

Tapez : T[n] **ENTREE**

ou : T **ENTREE**

Effet : A partir d'un point d'arrêt, la commande T [n] (TRACE) permet de visualiser [n] instructions du programme en cours d'exécution ainsi que le contenu des registres internes du 6809.

Si [n] est omis dans la commande TRACE, le moniteur forcera n = 1. Les instructions du programme ne sont visualisables qu'en mnémoniques.

La commande TRACE utilisant la technique des points d'arrêt, elle ne peut pas être utilisée sur la mémoire ROM (voir BREAK).

(On ne peut pas « tracer » les routines du moniteur TO7)

Exemple :

```
PC  A  B  DP  CC  X  Y  U  S
#
#KM2
ENTREE
#GSTART
ENTREE
#BRK @ M2
#T
ENTREE
#BC14 LDD #5555
#T3
ENTREE
```

WRITE : écrire

Commande : Le moniteur en état « attente de commande » « # » après affichage d'un point d'arrêt.

Tapez : W[n] **ENTREE**

ou : W **ENTREE**

Effet : La commande W[n] (WRITE) est une variante de la commande TRACE. La fonction est presque semblable, mais WRITE s'exécute sans afficher les sous-programmes qu'ils soient en mémoire RAM ou ROM.

WRITE est un TRACE du programme principal.

Il permet par ailleurs de tracer un programme dont les sous-programmes sont en ROM.

Les instructions des sous-programmes ne sont pas comptées dans la valeur [n].

Exemple :

```
PC  A  B  DP  CC  X  Y  U  S
#
#KBC00
ENTREE
#GSTART
ENTREE
@ BRK @ START
#U
BC02 00 00 00 80 0000 0000 0000 62FA
ENTREE
0BC02 JSR >INIT
#U
BC05 00 00 00 84 0000 0000 0000 62FA
ENTREE
0BC05 LDA >PORTC
#U2
ENTREE
BC08 01 00 00 80 0000 0000 0000 62FA
0BC08 ORA #1
BC0A 01 00 00 80 0000 0000 0000 62FA
0BC0A STA >PORTC
#
```

PRINT : imprimer

Commande : Le moniteur en état « attente de commande » « # »

Tapez : **P** **ENTREE**

Effet : La commande **P** (**PRINT**) permet de commuter l'affichage du MONITEUR sur l'imprimante.

Après avoir reçu la commande **PRINT**, le moniteur demande la confirmation de la commande en envoyant le message « Printer Ready Y/N? »

Si la réponse est positive, la commande **PRINT** est exécutée. Si la réponse est négative, la commande **PRINT** est annulée et le moniteur attend une nouvelle commande.

La commutation de l'affichage sur l'imprimante préserve le contenu de l'écran.

Pendant la période d'affichage sur l'imprimante, les échos des commandes à partir du clavier ne sont pas générés. Les lignes sont affichées, en bloc, après validation par la touche **ENTREE**.

Une seconde commande **PRINT** génère un saut jusqu'au sommet de la prochaine page sur l'imprimante et redonne l'affichage sur l'écran en effaçant le contenu précédent.

Exemple :

#P

ENTREE

Printer Ready Y/N?

Y

(l'affichage est commuté sur l'imprimante)

P **ENTREE**

(l'imprimante effectue un saut de page, l'écran est effacé et l'affichage est redonné à l'écran)

SAVE : Sauvegarde d'un programme objet

Commande : Le moniteur en état « attente de commande » « # »

Tapez : **S**<descripteur> <adr.1> <adr.2> <adr.3> **ENTREE**

Effet : La commande **S** <descripteur de fichier> <adresse1> <adresse2> <adresse3> (**SAVE**) permet de sauvegarder un programme objet ou des données sur le périphérique assigné dans la commande.

Si le descripteur de fichier ne contient pas de suffixe, le moniteur forcera le suffixe **.BIN** par défaut.

La désignation du périphérique n'est pas obligatoire dans la commande. Le lecteur de disquettes **Ø** ou le **LEP** seront utilisés par défaut.

La zone mémoire à sauvegarder est comprise entre <adresse1> et <adresse2>, <adresse1> spécifiant le début et <adresse2> la fin de la zone mémoire à sauvegarder.

<adresse3> est l'adresse qui sera utilisée pour lancer l'exécution du programme.

<adresses 1, 2 et 3> peuvent être spécifiées par une constante, une étiquette ou une expression, sous réserve que les étiquettes utilisées soient présentes dans la table des symboles issue du dernier assemblage.

Si vous utilisez un **LEP**, le **MONITEUR** posera la question : « Cassette Ready Y/N? ». Si la réponse est positive le programme sera sauvegardé.

Exemple :

#S0:MOIRAGE1.OBJ BC00 BC2F BC00

ENTREE

(MOIRAGE1.OBJ est sauvegardé.)

La même commande peut aussi s'écrire :

#SMOIRAGE1 START START+2F START

ENTREE

(MOIRAGE 1.BIN est sauvegardé.)

LOAD : Chargement d'un programme objet

Commande : Le moniteur en état « attente de commande » « # »

Tapez : L<descripteur> [<offset>] **ENTREE**

Effet : La commande L <descripteur de fichier> <offset> (LOAD) charge en mémoire un programme objet qui a été sauvegardé sur le périphérique assigné dans la commande.

<offset> est optionnel dans la commande LOAD et permet une translation du programme objet.

Le moniteur charge le compteur de programmes (PC) avec l'adresse d'exécution qui a été spécifiée à la sauvegarde du programme.

Si (offset) est présent dans la commande, le compteur de programmes (PC) sera chargé à la valeur obtenue en ajoutant l'offset à l'adresse d'exécution.

Note : L'adresse d'exécution du programme peut être définie, dans le moniteur, par <adr3> dans la commande SAVE, ou bien dans l'éditeur, quand le programme objet est créé par l'assembleur ; l'adresse d'exécution est alors définie par la présence de l'étiquette, dans le champ opérande de la directive END, qui adresse la première instruction à exécuter dans le programme.

Si aucune étiquette ni symbole n'est spécifié avec la directive END, l'assembleur charge 0000 comme adresse d'exécution.

Exemple :

```
#L0:MOIRAGE1.OBJ
```

ENTREE

MOIRAGE1.OBJ est chargé à l'adresse spécifiée lors de la sauvegarde. (BC00)

```
#L0:MOIRAGE1 0040
```

ENTREE

MOIRAGE1.BIN est chargé à l'adresse spécifiée lors de la sauvegarde plus 0040.(BC40)

VERIFY : vérification

Commande : Le moniteur en état « attente de commande » « # »

Tapez : V<descripteur> <adr.1> <adr.2> <adr.3> **ENTREE**

Effet : La commande V <descripteur de fichier> <adresse1> <adresse2> <adresse3> (VERIFY) compare le contenu du fichier défini par le descripteur avec la zone mémoire comprise entre <adresse1> et <adresse2>.

<adresse1> début de zone

<adresse2> fin de zone

<adresse3> adresse d'exécution

Si les contenus sont les mêmes, le curseur se repositionne et attend une nouvelle commande moniteur.

Si la zone mémoire spécifiée et le fichier désigné sont différents, un message d'erreur « Verification error » est affiché sur la ligne commentaires du moniteur.

ATTENTION : adr1, adr2, adr3 doivent être **EXACTEMENT** les mêmes que ceux utilisés dans SAVE ou que le début, la fin et l'adresse d'exécution d'un fichier créé par l'assembleur. En outre, le programme assemblé vers un fichier peut ne pas correspondre à l'image mémoire s'il ne contient pas un ensemble de code CONTINU. Par exemple, des ORG successifs peuvent créer un fichier segmenté ; de même, des RMB incrémentent le compteur de programme sans créer de code. Dans ce cas, l'image mémoire n'est pas comparable au fichier du programme objet.

Exemple :

```
#L MOIRAGE1.OBJ
```

ENTREE

```
#V0:MOIRAGE1.OBJ BC00 BC2F BC00
```

ENTREE

```
#
```

QUIT : retour sous contrôle de MENU

Commande : Le moniteur en état « attente de commande » « # »

Tapez : Q ENTREE

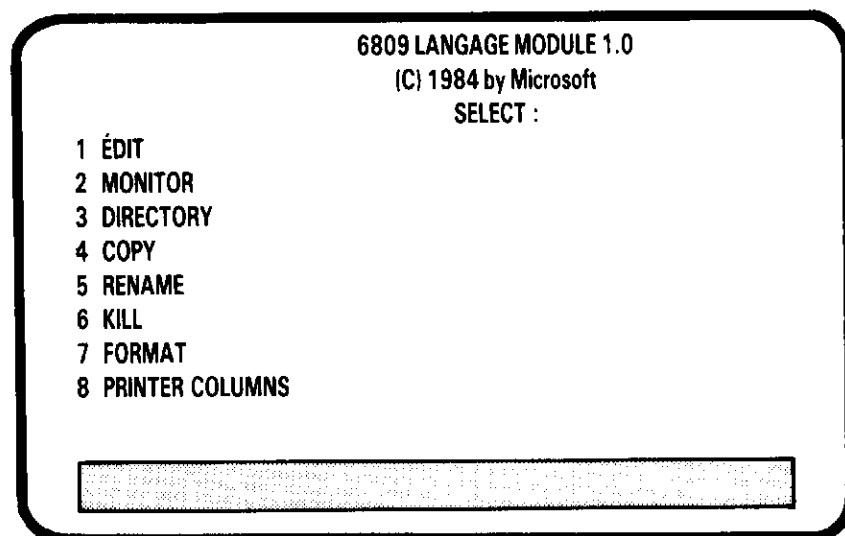
Effet : La commande Q (QUIT) redonne la main au MENU. À l'issue de la commande QUIT, le système génère l'écran de MENU et attend une commande.

Exemple :

#Q

ENTREE

MENU est affiché



EXIT : passage sous contrôle de l'ÉDITEUR

Commande : Le moniteur en état « attente de commande » « # »

Tapez : X ENTREE

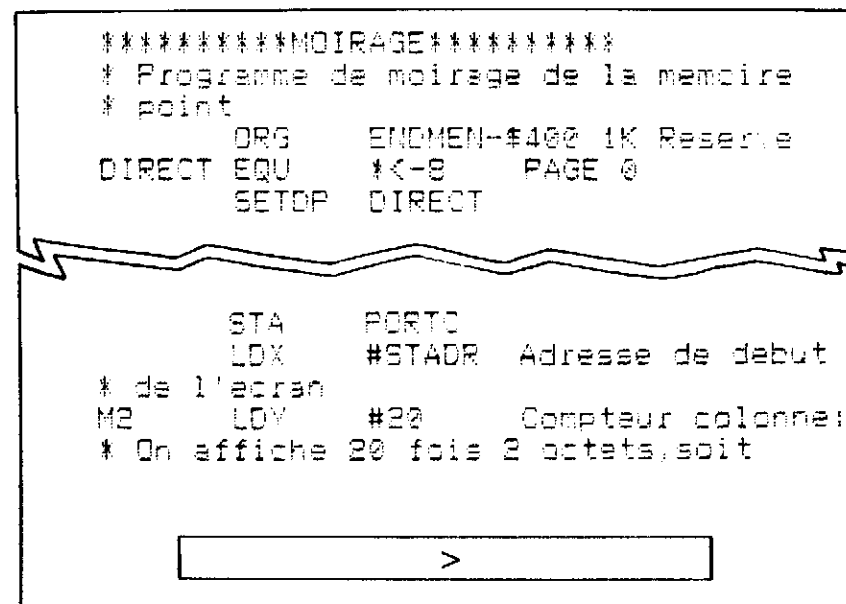
Effet : La commande X (EXIT) permet au système de quitter le moniteur et de passer sous contrôle de l'éditeur.

A l'issue de la commande EXIT, le système génère l'écran de l'éditeur et affiche les vingt-trois premières lignes du programme source résident.

Exemple :

#X

ENTREE



CHAPITRE 5

GESTION DE FICHIERS

La GESTION DE FICHIERS permet l'utilisation de fichiers sur disquettes et cassettes.

Il est constitué de six routines accessibles par le MENU qui permettent :

- d'établir le catalogue d'une disquette (DIRECTORY)
- de copier un fichier (COPY)
- de renommer un fichier (RENAME)
- d'effacer un fichier (KILL)
- de formater une disquette (FORMAT)

DIRECTORY : catalogue

La commande s'applique seulement aux disquettes.

La commande de MENU **3** active la ligne DIRECTORY

3 DIRECTORY ☐

Le système attend un numéro de lecteur de disquettes compris entre 0 et 3 puis une validation par la touche **ENTREE**.

Si aucun numéro de lecteur n'est précisé avant la commande **ENTREE**, le système utilise le lecteur 0 par défaut.

Un numéro non compris entre 0 et 3 provoque l'affichage du message « Bad parameter » (mauvais paramètre) et le retour sous contrôle de MENU.

Un numéro correspondant à un lecteur absent provoque l'affichage du message « Disk Not Ready » (disque non prêt) et le retour sous contrôle de MENU.

DIRECTORY (catalogue) affiche la liste des programmes enregistrés dans la disquette positionnée dans le lecteur spécifié.

La ligne 25 (rouge) est destinée à recevoir les commentaires et messages d'erreur sous contrôle de la routine DIRECTORY.

Exemple :

Disk not Ready ou Press any Key

(disque non prêt ou appuyer une touche pour continuer)

EQUATES	.ASM	A A 1	MOIRAGE	.ASM	A A 1
MOIRAGE	.BIN	M B 1	PRU	.TRU	A A 1
T07	.ASM	A A 0	MICR6809	.T07	R B 4
1	2	3 4 5	1	2	3 4 5

1. Nom du fichier (huit caractères maximum)

2. Suffixe

(par défaut).ASM Programme source

.BIN Programme objet

Note : Pour le nom du fichier et le suffixe, l'utilisateur peut choisir d'autres noms. Ces noms peuvent contenir tous les caractères imprimables (non accentués) sauf « < > » « ? » « » et « : »

3. Type de fichier.

.A Programme assembleur (source)

.M Programme machine (objet)

4. Type de donnée.

.A ASCII

.B Binaire

5. Nombre de blocs utilisés par le programme sur la disquette.
(taille d'un bloc = 1 K octet)

Le retour sous contrôle de MENU est obtenu en pressant une touche quelconque.

Note : La consultation de DIRECTORY n'efface pas les programmes source et objet en mémoire. Vous pouvez donc utiliser cette fonction, à tout moment, pour contrôler les enregistrements et modifications de nom de fichier que vous venez d'effectuer.

DIRECTORY vous permet aussi de savoir si une disquette est initialisée. Avec une disquette qui n'est pas formatée, le système retourne le message : « Unreadable Disk » (disque illisible).

La commande DIRECTORY permet également de connaître la place encore disponible sur la disquette en faisant :

Place disponible en K octets = 80 - \sum blocs utilisés - 2

COPY : copie entre disquettes ou entre cassette et disquette

La commande de MENU **4** active la ligne COPY :

4 COPY TO

COPY attend le premier descripteur de fichier qui doit être obligatoirement le fichier à recopier.

La zone de saisie est limitée à quatorze caractères et validée par la commande **ENTREE**.

La ligne 25 (rouge) est destinée à recevoir les messages d'erreur.

4 COPY TO

La seconde zone, également limitée à quatorze caractères, est occupée par le fichier destination et validé par **ENTREE**.

Sa tâche effectuée, COPY retourne directement sous contrôle de MENU.

COPY permet la copie intégrale d'un fichier en changeant éventuellement son nom.

Avec deux lecteurs de disquettes, COPY permet de copier un fichier d'un lecteur sur l'autre en spécifiant les numéros des périphériques utilisés dans les descripteurs de fichier correspondants.

Avec un seul lecteur de disquette, COPY permet de copier un fichier sur la même disquette en changeant le nom du fichier. Si le périphérique n'est pas spécifié dans le descripteur source, le système utilisera le lecteur 0 par défaut. Les numéros de lecteurs autorisés sont : 1, 2 et 3.

Le suffixe est indispensable dans le descripteur de fichier source.

Le système complètera le descripteur destinataire avec les éléments du fichier source.

COPY permet également des copies de la cassette vers une disquette et vice-versa.

Exemple :

4 COPY TO

ENTREE

(COPIE DE 0 : MOIRAGE.RSM en 0 : MOIRAGE1.ASM)

4 COPY TO

ENTREE

(copie de 0 : MOIRAGE.ASM en C : MOIRAGE.ASM)

RENAME : changement de nom

La commande s'applique seulement aux disquettes.

La commande de MENU **5** active la ligne RENAME :

5 RENAME AS

RENAME attend le premier descripteur de fichier qui doit être obligatoirement le fichier à renommer.

La zone de saisie est limitée à quatorze caractères et validée par la commande **ENTREE**.

La ligne 25 (rouge) est destinée à recevoir les messages d'erreur.

5 RENAME AS

La seconde zone, également limitée à quatorze caractères, est occupée par le nouveau descripteur de fichier et validée par **ENTREE**.

Le changement de nom effectué, le système retourne sous contrôle de MENU.

La vérification du résultat peut être obtenu par la commande **DIRECTORY**.

RENAME permet le changement de nom d'un fichier sur une disquette seulement.

Le premier descripteur de fichier contient le nom du fichier à renommer, le second le nouveau nom du fichier.

Si le nouveau nom du fichier correspond à un fichier déjà existant dans la disquette, la message « File already exists » est affiché.

Si le périphérique n'est pas spécifié dans le descripteur à renommer, le système utilisera le lecteur 0 par défaut. Si le périphérique est omis dans le nouveau nom, le système utilisera le descripteur de fichier à renommer. Les numéros de lecteurs autorisés sont : 0, 1, 2 et 3.

Le suffixe est indispensable dans le descripteur de fichier à renommer.

Exemple :

5 RENAME AS

ENTREE

(renomme 0 : MOIRAGE.ASM en 0 : MOIRAGE1.ASM)

5 RENAME AS

ENTREE

(renomme 0 : MOIRAGE1.ASM en 0 : MOIRAGE1.ESS)

KILL : suppression d'un fichier

La commande s'applique seulement aux disquettes.
La commande de MENU **6** active la ligne KILL :

6 KILL

KILL attend l'entrée du descripteur de fichier à détruire.
La zone de saisie est limitée à quatorze caractères et validée par la commande **ENTREE**.
La ligne 25 (rouge) est destinée à recevoir les messages d'erreur ou les questions à destination de l'utilisateur. Après la validation de l'entrée, la commande KILL envoie, sur la ligne de commentaires, le message « Are you sure Y/N? » (êtes-vous sûr O/N ?).
Si la réponse est positive, le fichier est détruit, et le système retourne sous contrôle de MENU.
La commande KILL efface le fichier défini par le descripteur de fichier, supprime ses références dans le catalogue et libère la place qu'il occupe sur la disquette.
Si le périphérique n'est pas spécifié dans le descripteur de fichier à détruire, le système utilisera le lecteur 0 par défaut.
Le suffixe est obligatoire dans le descripteur de fichier à détruire.
La commande KILL ne fonctionne que sur un fichier résidant sur une disquette.

Exemple :

(créons le fichier « DETRUIT.FIC » afin de pouvoir le détruire dans la suite de notre exemple)

4 COPY TO
ENTREE

(0 : DETRUIT.FIC est créé sur la disquette du lecteur 0)

6 KILL
ENTREE

Le système répond

Y (0 : DETRUIT.FIC est détruit)

FORMAT : initialisation d'une disquette

La commande s'applique seulement aux disquettes.
La commande de MENU **7** active la ligne FORMAT.

7 FORMAT

FORMAT attend l'entrée d'un numéro de lecteur de disquettes compris entre 0 et 3.

Cette entrée est obligatoire : contrairement aux autres commandes, le lecteur 0 n'est pas pris par défaut.

ENTREE valide le lecteur spécifié.

La ligne 25 (rouge) est destinée à recevoir les messages d'erreur ou les questions à destination de l'utilisateur. Après la validation de l'entrée, la commande FORMAT envoie sur la ligne de commentaires le message « Are you sure Y/N? » (« êtes-vous sûr O/N ? »).
Si la réponse est positive la disquette placée dans le lecteur spécifié est initialisée (on dit aussi formatée) et le système retourne sous contrôle de MENU.

La disquette ne doit pas être protégée, c'est-à-dire que son encoche de protection doit être libre.

Après la phase d'initialisation, le catalogue de la disquette est vide. L'initialisation d'une disquette efface toutes les données qu'elle pouvait contenir auparavant.

Toute disquette neuve doit obligatoirement être initialisée avant d'être utilisée.

Exemple :

(nous vous proposons d'initialiser la disquette neuve que vous avez placée dans le lecteur 0)

7 FORMAT
ENTREE

Le système répond

Y (Le lecteur 0 tourne. Dans trois minutes environ votre disquette sera initialisée ou formatée.)