

MANUEL CP/M3



Copyright version originale © 1985

Toute reproduction, même partielle, par quelque procédé que ce soit, est interdite sans autorisation écrite préalable. Une copie ou reproduction par xérographie, photographie, film, bande magnétique ou autre constitue une contrefaçon passible des peines prévues par la loi sur la protection des droits d'auteur.

Printed in France. World rights reserved.

PREAMBULE

Ce manuel d'introduction au système CP/M 3 ne se veut en aucun cas un outil d'aide à la programmation avancée en langage assembleur.

Son but est de cerner toutes les fonctions de base du système CP/M par une description conséquente de la syntaxe, une explication de ses possibilités et des exemples d'utilisation dans un cadre d'une utilisation de base.

Toute critique constructive concernant ce manuel sera la bienvenue et répercutée au niveau du club Guépard.

C. DECHAMPS

1 - INTRODUCTION AU CP/M 3

I/ LANCEMENT DU CP/M

Mettre la disquette système CP/M dans le lecteur de droite (drive A) et appuyer sur le bouton reset du guépard.

Le chargement du CP/M en mémoire est effectué lorsque le message d'initialisation se présente à l'écran sous la forme :

```
CP/M V 3.0 (c) 1982 DIGITAL Research
BIOS V 3.0 (c) 1984 HBN COMPUTEUR & RINCENT Gérard
```

```
NUMERO DE SERIE : 1279 - xxxxx
```

Après l'initialisation un message à deux caractères apparaît sur l'écran :

```
A>
```

C'est le prompt du CP/M qui vous indique que le système est prêt à recevoir vos commandes.

Le A indiquant le lecteur de disquettes pris par défaut par le système. C'est le lecteur à partir duquel le système effectuera les commandes.

LA LIGNE DE COMMANDE :

Une ligne de commande s'écrit sous la forme d'un mot de commande, d'une option suivant le type de la commande et d'un carriage return pour lancer l'exécution.

Le carriage return est obtenu en appuyant sur la touche ENTER de votre clavier.

En cas d'erreur de frappe lors d'une composition de ligne, vous pouvez modifier le ou les caractères incorrects en appuyant sur la touche CLEAR ou sur CTRL-H.

CTRL étant l'abréviation de la touche contrôle.

Ces deux commandes permettent l'effacement du dernier caractère de la ligne et déplacent le curseur d'une position vers la gauche.

L'effacement de la ligne complète peut se faire en appuyant sur CLEAR et en attendant l'auto-repeat ou en appuyant sur SHIFT-CLEAR ce qui a pour effet de ramener le curseur sur le prompt CP/M et d'effacer la ligne de commande.

Le système CP/M reçoit deux types de commandes :

- les commandes directes
- les commandes indirectes

Les commandes directes sont exécutables à partir de la mémoire, les commandes indirectes étant exécutées après chargement de l'utilitaire concerné depuis le disque. Vous pouvez reconnaître ces commandes car elles sont suivies de l'option .COM

Ex. : exécuter la commande DIR qui permet l'affichage du contenu de la disquette.

```
A> DIR (cr)
```

```
A : GET          COM      A : LIB          COM ...
A : ED           COM      A : INITDIR        COM ...
```

Une commande mal formulée ou non reconnue par le système sera reprise et affichée suivie d'un point d'interrogation.

```
Ex. : A> ABC      commande passée
      ABC ?       réponse du système
      A>          retour au prompt
```

II/ REPRODUCTION DE DISQUETTES

1 - FORMATAGE :

Lorsque le prompt du CP/M est affiché, lancer la commande **FORMAT**

A> **FORMAT** (cr)

Le système charge alors en mémoire l'utilitaire de formatage et affiche le message suivant :

Sur quel drive désirez vous formater (a, b, c ou d) ?

Appuyer sur <Enter> pour retour au CP/M

Dès lors si vous appuyez sur la touche enter, vous retournerez dans la configuration de départ.

Les drives sous CP/M sont désignés par des lettres. Sur un guépard version de base (deux lecteurs de disquettes) vous n'utiliserez que les lettres A et B correspondants respectivement aux lecteurs de droite et de gauche. Le formatage d'une nouvelle disquette se fera donc sur le drive B en y plaçant une disquette vierge et en appuyant sur la lettre B.

Un autre message s'inscrit :

Nombre de pistes :

(1) 40 pistes

(2) 80 pistes

Le formatage est réalisable en 40 pistes ou en 80 pistes suivant le type de guépard dont vous disposez. Le plus courant étant le 80 pistes. Appuyez donc sur 2.

Un autre message s'inscrit :

Face ?

(s) pour simple

(d) pour double

Suivant l'utilisation que vous désirez faire de votre disquette, vous pouvez travailler en simple ou double face. La différence entre les deux résidant au niveau de la place utilisée sur la disquette. (334 k ou 690 k)

Si vous désirez recopier votre disquette système, vous devrez impérativement sélectionner la double face en appuyant sur la lettre D.

Attention indiquez sur vos disquettes le format que vous avez choisi lors de la création.

Ex. : disquette CP/M S.F. pour simple face

Il ne reste plus qu'à appuyer sur <Enter> si votre disquette est en place dans l'appareil.

Après le formatage, le retour au CP/M se fait par appui sur <Enter> lorsque le premier message réapparaît sur l'écran.

Vous venez de formater votre disquette pour une utilisation sous CP/M.

Cette disquette peut vous servir telle qu'elle pour une recopie de fichiers mais ne peut encore booter sur votre appareil.

Pour ce faire vous devez passer à la copie du système.

2 - COPIE DU SYSTEME : COPYSYS

Pour rendre votre disquette système, celle-ci doit posséder les pistes systèmes du CP/M ainsi que le fichier CCP.COM.

Lancer l'utilitaire COPYSYS

GENERATION DE DISQUETTES SYSTEMES V 1 . 0

Format de la disquette destination :

(1) 80 pistes simple face

(2) 80 pistes double face

Suivant le format adopté précédemment, indiquez l'option choisie :

Ex. : pour une simple face tapez (1)

Le message suivant s'inscrit :

Disquette source (a, b, c ou d) ? Appuyez sur A

Disquette destination (a, b, c ou d) ? Appuyez sur B

L'utilitaire recopie alors les pistes systèmes de votre disquette source sur la disquette destination puis le fichier CCP.COM et affiche :

GENERATION TERMINEE

Vous êtes alors en possession d'une disquette système CP/M capable de booter et ne possédant que le fichier CCP.COM.

3 - CONFIGURATION DES LECTEURS :

L'écriture ou la lecture en double ou simple face doit être précédée d'une configuration des lecteurs de disquettes.

Vous disposez de deux programmes pour les 80 pistes et deux pour les 40.

GUEP80SF et GUEP80DF permettent de configurer le système respectivement en simple et double face (80 pistes).

Ex. : — si vous travaillez en double face sur la disquette système et que vous désirez travailler en double face sur la disquette fichier pas de manipulations nécessaires.

— si vous travaillez en simple face sur la disquette système et que vous désirez travailler en simple face sur la disquette fichier pas de manipulations nécessaires.

— si vous travaillez en double face sur la disquette système et que vous désirez travailler en simple face sur la disquette fichier vous devez au préalable et après chaque appui sur RESET lancer l'utilitaire GUEP80SF en sélectionnant le drive B.

— si vous travaillez en simple face sur la disquette système et que vous désirez travailler en double face sur la disquette fichier vous devez au préalable et après chaque appui sur RESET lancer l'utilitaire GUEP80DF en sélectionnant le drive B.

Les mêmes manipulations doivent être réalisées si vous possédez un 40 pistes. De la même manière si vous désirez lire des disquettes au format IBMPC vous avez à votre disposition deux utilitaires qui permettent la configuration double ou simple face IBM. Ce sont : IBMPCDF.COM & IBMPC.COM

4 - COPIE DE FICHIERS :

La copie de fichiers est réalisée par l'utilitaire PIP.COM après CONFIGURATION de vos lecteurs.
(pour une utilisation plus avancée de l'utilitaire PIP.COM se reporter au chapitre le concernant).

Exécuter la commande suivante :

```
A> PIP (cr)
CP/M 3 Version 3.0
*          l'étoile correspond au prompt de l'utilitaire.
```

Faites alors :

```
*B:=A:*. * <V>      (cr)
```

Cette ligne de commande indique au système que la copie doit se faire de la disquette A sur la disquette B pour une copie complète avec vérification.

La vérification permet d'inhiber les erreurs disques lors de l'écriture.

La copie se fait séquentiellement, fichier par fichier, et se termine par un retour au prompt de PIP.

Pour revenir au CP/M appuyez sur <Enter>

Après toutes ces manipulations, vous possédez une disquette système, reproduction exacte de votre propre disquette qu'il est conseillé de classer à l'abri de la poussière et des champs magnétiques pour une utilisation en cas de destruction de la copie.

III/ ACCES AUX FICHIERS

L'accès aux programmes sur la disquette maître se fait en indiquant dans la ligne de commande le nom du programme à lancer.

```
Ex. : A> SET.COM (cr)
```

Pour appeler un programme sur le drive B faites :

```
A> B : SET.COM (cr)
```

Pour certaines commandes une inversion au niveau de la syntaxe s'impose :

```
A> DIR B:
```

Vous pouvez passer le drive B comme drive pris par défaut par le système

```
Tapez : A> B:      commande
        B>          réponse
```

Le système lors de l'exécution de cette commande va chercher les paramètres du drive B et les stocke en mémoire.

Une fois la commande exécutée, tous les accès aux fichiers se feront sur le drive B.

Parallèlement à ce changement de drive vous pouvez définir à l'intérieur d'un directory de disquette des sous directory appelés USER.

Ces USER sont numérotés de 0 à 15 et permettent donc 16 sous directory différents sur une même disquette.

Pour passer sur un USER faites :

```
A> 2A: (cr)      USER 2 drive A ou taper A> USER x (cr)
2A>              réponse du système

1A> 3B:          passage du user 1 drive A au user 3 drive B
3B>
```

1 - ACCESSION A PLUSIEURS FICHIERS :

Deux caractères spéciaux permettent d'obtenir des jeux de combinaisons au niveau de l'utilisateur. Ce sont les caractères : ? et *

Ex. : un nom de fichier est écrit sous la forme : 8 caractères. 3 caractères (INITDIR.COM)

L'option ? indique un caractère indéfini dans un fichier

Ex. : A > DIR ? .COM listera tous les fichiers dont le nom est codé sur un seul caractère et portant l'option .COM.

Le sigle * remplace tous les ? d'une commande.

Ex. : A > DIR ??????.COM = A > DIR *.COM liste des fichiers en .COM

Suivez bien les exemples suivants :

Votre disquette contient les fichiers :

A.COM AA.COM AAA.COM B.COM A.LIB B.LIB

- DIR *.* liste tous les noms de fichiers (= DIR)
- DIR *.COM liste tous les fichiers portant l'extension .COM
- DIR ??????.COM idem précédemment
- DIR ?.* listera les fichiers A.COM, B.COM, A.LIB, B.LIB
- DIR ?.??? listera les fichiers A.COM, B.COM, A.LIB, B.LIB
- DIR A ?.COM listera les fichiers A.COM et AA.COM
- DIR A *.COM sera traité comme A ??????.COM et donnera A.COM, AA.COM, AAA.COM

2 - PROTECTION DES FICHIERS :

La protection des fichiers s'effectue par :

- numéro de USER
- attributs
- heure et date
- mot de passe

Attributs :

Deux attributs sont admis par le CP/M lors de la création des fichiers. Ils peuvent être chargés par la commande SET.

Lors de la création le premier attribut est DIR ce qui permet de lister tous les noms de fichier par la commande DIR.

Vous pouvez remplacer cette option par SYS et seule une commande DIRSYS donnera accès au fichier lors d'un affichage de directory.

Si vous donnez l'attribut SYS sur un fichier du USER 0, ce fichier pourra être utilisé par tous les autres USER ce qui permet de faire un directory commun.

Le second attribut peut être R/W ou R/O pour écriture/lecture ou lecture seule.

Date :

Se rapporter à la commande SET.COM

Mots de passe :

Se rapporter à la commande SET.COM

2 - CARACTERES DE CONTROLE

2.1 - IMPRIMANTE

Recopie sur imprimante : CTRL-P

La fonction de recopie permet de sortir sur imprimante les caractères envoyés à la vidéo.

Ex. : CTRL-P	sélection de l'imprimante
A > DIR	affichage du directory
CTRL-P	désélection de l'imprimante

Listing d'une partie de fichier : CTRL-S CTRL-P CTRL-Q

Pour ne lister qu'une partie de fichier, utilisez la commande type pour débiter un listing sur l'écran.

Appuyer sur CTRL-S (le listing s'arrête)
CTRL-P (sélection de l'imprimante)
CTRL-Q (reprise de l'affichage + édition)

Pour l'inhibition, pratiquez de la même manière :

Appuyer sur CTRL-S (le listing s'arrête)
CTRL-P (désélection de l'imprimante)
CTRL-Q (reprise de l'affichage sans édition)

Ex. : A > TYPE HELP.HLP
faites CTRL-S, CTRL-P, CTRL-Q en ayant pris soin de connecter une imprimante.

2.2 - VIDEO :

CTRL-S : Arrêt de l'affichage

CTRL-Q : Reprise de l'affichage

CTRL-H : Retour arrière avec effacement (back-space)

CTRL-E : Retour ligne suivante sans lancement de la commande en cours

CTRL-I : tabulation

CTRL-J : Identique au retour chariot

CTRL-M : Identique au retour chariot

CTRL-R : Place un # à l'endroit du curseur et répète la ligne sur la ligne suivante

CTRL-U : Place un # à l'endroit du curseur et se place sur la ligne suivante

CTRL-X : Effacement de la ligne en cours et retour début de ligne

CTRL-C : Arrêt d'exécution d'une procédure en cours (programme en cours)

CLEAR : Idem CTRL-H

SHIFT-CLEAR : idem CTRL-X

TAB : Idem CTRL-I

3 - COMMANDES DU CP/M3

REPERTOIRE DES COMMANDES DU SYSTEME SOUS CP/M3 :

DATE	: MC-7	INITDIR	: MC-13	SAVE	: MC-23
DEVICE	: MC-8	LIB	: MC-14	SET	: MC-23
DIR	: MC-9	LINK	: MC-14	SETDEF	: MC-27
DUMP	: MC-10	MAC	: MC-15	SHOW	: MC-29
ED	: MC-10	PATCH	: MC-15	SID	: MC-30
ERASE	: MC-10	PIP	: MC-15	SUBMIT	: MC-32
GENCOM	: MC-11	PUT	: MC-21	TYPE	: MC-34
GET	: MC-12	RENAME	: MC-22	USER	: MC-34
HELP	: MC-13	RMAC	: MC-22	XREF	: MC-34
HEXCOM	: MC-13				

DATE :

Syntaxe : DATE (cr)
DATE HH:MM:SS
DATE JJ/MM/AA
DATE C
DATE SET

Explication : La commande DATE permet l'affichage de la date et de l'heure et la mise à jour de ces paramètres.

Exemples :

A> DATE affiche la date et l'heure du jour

A> DATE 10/10/85 mise à jour de la date

A> DATE 09:25:85 mise à jour de l'heure

A> DATE C affichage de la date et de l'heure avec continuité
 jusqu'à l'appui sur la touche <Enter>

A> DATE SET mise à jour des paramètres mais avec menu d'aide

DEVICE :

Syntaxe : `DEVICE < NOMS:VALEURS:PHYSICAL DEVICE:
LOGICAL DEVICE>
DEVICE LOGICAL DEVICE = PHYSICAL DEVICE
<OPTIONS>
DEVICE LOGICAL DEVICE = NULL
DEVICE PHYSICAL DEVICE <OPTIONS>
DEVICE CONSOLE < PAGE : COLONNES : LIGNES>`

Explications :

La commande `DEVICE` permet l'affectation des unités logiques aux unités périphériques de votre micro-ordinateur.

Elle permet aussi l'installation de la sortie série et la modification de la taille de votre écran.

CP/M supporte cinq périphériques logiques :

— CONIN :	entrées	(clavier)	(CLAV)
— CONOUT :	sortie	(écran)	(VID)
— AUXIN :	auxiliaire en entrée	(rs232)	(RS)
— AUXOUT :	auxiliaire en sortie	(rs232)	(RS)
— LST :	imprimante	(imprimante)	(IMP)

Exemples :

```
A > DEVICE LST = NULL
A > DEVICE RS <9600>
A > DEVICE CONOUT: = VID,IMP
```

Le premier exemple assigne au périphérique de sortie imprimante une valeur nulle. Elle dévalide ce périphérique.

Le deuxième exemple initialise la sortie série à une vitesse de 9600 bauds. (pas de protocole possible).

Le troisième exemple associe au périphérique de sortie la vidéo et l'imprimante ce qui permet de lister en même temps à l'écran et sur papier.

Vous pouvez tout aussi bien n'assigner comme périphérique de sortie que l'imprimante : `A > DEVICE CONOUT = IMP`

`A > DEVICE (cr)`

Permet l'affichage sous forme de menu de l'affectation des périphériques et la modification de ceux ci.

`A > DEVICE CON` liste les périphériques logiques sortie et entrée

`A > DEVICE AUX` liste les périphériques auxiliaires

`A > DEVICE CONSOLE PAGE`

Permet l'affichage des paramètres de votre écran (colonne et ligne)

DIR :

Syntaxe : DIR x:
DIR FICHIER
DIRSYS ou DIRS
DIR x : OPTIONS

Explications :

La commande DIR et DIRS permet l'affichage du directory de votre disquette dans la zone du user spécifié.

Elle accepte les caractères de contrôle ? et * .

Exemples :

- | | |
|--------------------|---|
| A> DIR | affiche tous les noms de fichier du user 0
drive A |
| A> DIR B: | affiche tous les noms de fichier du user 0
drive B |
| 2A> DIR C: SET COM | affiche à l'écran le nom du fichier SET.COM
si le fichier se trouve sur la disquette drive C |
| 4B> DIR *.COM | affiche tous les fichiers portant l'extension
.COM du drive B USER 4 |
| 4B> DIR X*.??M | affiche les noms de fichiers sur le drive A
USER 0 débutant par X et dont l'option se
termine par M |
| A> DIRS | affiche tous les fichiers du drive A USER 0
portant l'attribut SYS (c.f fonction SET.COM) |

Options

- | | |
|--|--|
| A> DIR<FULL> | affiche les caractéristiques de tous les fichiers
du drive A USER 0 |
| A> DIR DATE | affiche l'ensemble des fichiers du drive A
A USER 0 avec leur date si celle ci à été
activée |
| A> DIR<RW,SYS> | affiche l'ensemble des fichiers portant l'ex-
tension écriture lecture et l'attribut SYS |
| A> DIR<USER-ALL> | affiche les caractéristiques de tous les fichiers
de tous les USER de l'unité active |
| A> DIR<DRIVE-ALL> | affiche les caractéristiques de tous les fichiers
de toutes les unités actives du système |
| A> DIR<EXCLUDE>*.COM | liste tous les fichiers du drive A avec leurs
caractéristiques ne portant pas l'extension
.COM |
| A> DIR<SIZE> | affiche les longueurs des fichiers du drive
pris par défaut |
| A> DIR<SIZE>*.COM | idem mais sur les fichiers ne comportant
que l'extension .COM |
| A> DIR<MESSAGE, DRIVE=ALL, USER=ALL, NOPAGE> | |

Liste les caractéristiques de tous les fichiers de toutes les unités actives de tous les user avec affichage de message sur chaque user. L'option NOPAGE indiquant au système de ne pas attendre de Return lors d'un scrolling.

DUMP :

Syntaxe : DUMP nom de fichier

Explications :

La fonction DUMP permet l'affichage du contenu d'un fichier sous forme de table hexadécimale et sous forme ascii

Exemple :

A > DUMP SET.COM

CP/M 3 Dump - Version 3.0

```
0000:  41  42  43  0D  0A  44  45  46  0D  0A  47  48  49  0D  0A  1A  ABC...DEF..GHI...
0010:  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  1A  .....
```

.
.
.

A > DUMP SET.COM <NOPAGE> idem précédemment mais sans attente de Enter entre chaque scrolling de page.

Rappel : CTRL-C arrête le scrolling en cours et retourne au CP/M

ED :

Syntaxe : ED fichier en entrée <x:> fichier en sortie

Explications :

L'utilitaire ED est un éditeur de fichier qui vous permet de créer vos fichiers ou de modifier d'anciens fichiers

Se rapporter au chapitre concernant l'éditeur pour de plus amples explications.

ERASE :

Syntaxe : ERASE nom de fichier <confirm>
ERA nom de fichier <confirm>

Explications :

La commande ERASE ou ERA (abréviation) permet de détruire un ou plusieurs fichiers d'une unité active ou d'un USER particulier.

Exemples :

A > ERA B:TEST.ASM effacement du fichier TEST.ASM sur l'unité B

A > ERA TEST.COM <CONFIRM> effacement du fichier TEST.COM sur l'unité A avec confirmation par oui ou non avant effacement.

A > ERA *.* effacement total de tous les fichiers de l'unité active cet effacement est obligatoirement confirmé.

GENCOM :

Syntaxe : GENCOM <FICHIER.COM> <FICHIER RSX> <OPTIONS>

Explications :

La commande GENCOM permet la création de fichier .COM sur lesquels peuvent être attachés des fichiers RSX.

Les extensions RSX sont des extensions résidentes au système.

GENCOM place un entête spéciale en début de fichier indiquant au système que le chargement d'un RSX est requis.

Il peut aussi remettre des fichiers .COM dans le format initial d'utilisation sous CP/M c'est à dire sans nécessité de charger une extension RSX.

L'option LOADER permet de maintenir activé, en positionnant un flag, le programme loader.

L'option NULL indique que dans la commande seuls les fichiers .RSX seront nommés.

Exemples :

A > GENCOM MONPROG PROG1 PROG2

Génération d'un fichier de commande nommé MONPROG.COM avec extensions PROG1.RSX et PROG2.RSX

A > GENCOM PROG1 PROG2 < NULL >

Génération d'un fichier fictif PROG1.COM avec extensions PROG1.RSX et PROG2.RSX

A > GENCOM MONPROG

Reprise du fichier MONPROG.COM avec suppression de l'entête et des fichiers portant l'option .RSX qui lui sont associés.

GET :

Syntaxe : GET < CONSOLE INPUT FROM > nom du fichier
GET < CONSOLE INPUT FROM > CONSOLE

Explications :

La commande get permet de définir l'entrée console comme un fichier pour la prochaine commande entrée au clavier.

L'entrée console sera prise comme un fichier jusqu'à ce que la commande en cours se termine.

Si le fichier est vide à la fin de la première commande, la commande GET s'appliquera à la commande suivante entrée par l'utilisateur.

En accolant l'option SYSTEM à la commande, le système va directement au fichier comme entrée console.

Pour faire revenir la console comme console d'entrée, taper la seconde ligne de commande.

Exemple :

```
A > GET FILE XINPUT  
A > MYPROG
```

La séquence de commande indique au système d'activer la commande GET.

L'option SYSTEM n'étant pas spécifiée, le système lit la seconde ligne de commande et exécute le programme MYPROG. Si le programme fait appel à la console, la prise des informations se fera à partir du fichier XINPUT. La commande exécutée, le système réassignera la console comme console d'entrée.

```
A > GET FILE XIN2 <SYSTEM>
```

La commande permet avec l'option system d'indiquer que tous les appels à la console se feront en provenance du fichier XIN2.

La réassignation de la console en console d'entrée devra se faire en entrant la commande sous la forme :

```
A > GET CONSOLE
```

HELP :

Syntaxe : **HELP SUJET SOUS-SUJET** <NOPAGE, LIST>
 HELP <EXTRACT>
 HELP <CREATE>

Explications :

La commande permet une aide à la programmation en indiquant les résumés des commandes du système CP/M.

Sans argument la commande liste tous les sujets disponibles.

Sans sous sujet la commande liste après listing du sujet, les sous-sujets disponibles

Exemples :

A > **HELP** appel du menu d'aide

A > **HELP DATE** appel du menu d'aide sur le sujet date

A > **HELP ED COMMANDES** appel du menu du sous-sujet des commandes de l'éditeur

A > **HELP** <EXTRACT> permet la création d'un fichier **HELP.DAT** permettant le listing continu des données contenues dans le fichier pour sortie sur imprimante.

A > **HELP** <CREATE> permet la création d'un fichier **HELP.HLP** à partir du fichier **HELP.DAT**

HEXCOM :

Syntaxe : **HEXCOM** nom de fichier

Explications :

La commande **HEXCOM** permet la transformation d'un fichier portant l'extension **.HEX** en fichier **.COM**

Rappel : le fichier **.HEX** étant généré par **MAC.COM**

Exemple :

A > **HEXCOM B:PROGRAMM.HEX**

Transformation du fichier **PROGRAMM.HEX** en fichier **PROGRAMM.COM**

INITDIR :

Syntaxe : **INITDIR** d :

Explications :

La commande **INITDIR** permet l'initialisation du directory pour permettre l'impression de la date et de l'heure sur les fichiers de ce disque. Si votre disquette est vierge, **INITDIR** initialisera le directory. Par contre si votre disquette contient déjà des fichiers, la commande vérifie la place disponible pour l'initialisation de la date et de l'heure. Après avoir initialisé votre directory, vous devez utiliser la commande **SET.COM** pour spécifier les options.

Exemples :

A / **INITDIR B :**

Si votre disque a déjà été initialisé, le programme vous demande si vous désirez reformatier les données contenues ou les conserver telles qu'elles.

LIB :

Syntaxe : LIB nom du fichier < I, M, P, D >
LIB nom de fichier < I, M, P > = nom du fichier à modifier

Explications :

Une bibliothèque est un fichier contenant des informations. Afin de créer ou modifier une librairie, utiliser la commande LIB.COM

Ces fichiers portent les extensions .REL et permettent leur utilisation par le macro assembleur de DIGITAL RESEARCH.

Options :

I : permet la création d'un fichier bibliothèque indexé de type .IRL
Cette création de bibliothèque indexée permet l'accélération du processus de recherche du linker.

M : affiche les noms de module

P : affiche les noms de module et les variables déclarées publiques

D : affiche un dump des fichiers modules sous forme ascii

Modificateurs :

Les modificateurs permettent de rechercher à l'intérieur d'une librairie un module donné et de le modifier ou le détruire.

Options :

delete : < module =>

replace : < module=nom de fichier .REL >

select : < 1er module, 2ème module,, nième module >

Exemples :

A > LIB TEST4 < P >

Affiche tous les modules et publics du fichier TEST4

A > LIB TEST5 < P > =FILE1, FILE2

Crée le fichier TEST5.REL à partir des fichiers FILE1 et FILE2 et liste tous les modules ainsi obtenus.

A > LIB TEST=TEST1 (MOD1, MOD4) , TEST2 (C1-C4, C6)

Création d'un fichier librairie de nom TEST des modules des fichiers de TEST1 (modules 1 et 4) ainsi que les modules C1 à C4 et C6.

A > LIB FICH2=FICH3 < MOD=>

Création d'un fichier librairie de nom FICH2 à partir du fichier FICH3 et en omettant le module MOD du fichier FICH3

A > LIB FILE6=FILE5 < MODA=FILEB.REL >

Création du fichier librairie de nom FILE6 à partir du fichier FILE5 et en remplaçant le module MODA par le fichier FILEB.REL

A > LIB FILE6=FILE5 < THISNAME >

Création du fichier librairie de nom FILE6 à partir du fichier FILE5 et dans lequel le module THISNAME est remplacé par le fichier de même nom en .REL

A > LIB FILE1 < I > = B : FILE2 < PLOTS, FIND, SEARCH=DISPLAY >

Création du fichier en indexé (FILE1.IRL) sur le drive A en reprenant les modules du fichier FILE2 plots, find et des modules search à display sur le drive B.

LINK :

Consulter l'annexe se rapportant à la commande LINK

MAC :

Consulter l'annexe se rapportant à la commande MAC

PATCH :

Syntaxe : PATCH nom de fichier <typ> <n>

Explications :

L'utilitaire PATCH affiche le numéro de patch au système CP/M 3 ou aux fichiers de commande. Ce numéro doit être compris entre 1 et 32.

Exemple :

A > PATCH SHOW 2

Rapièce le système de fichier SHOW.COM avec le numéro de patch 2

PIP :

Explications :

L'utilitaire PIP.COM permet de copier des fichiers d'une unité vers une autre, d'un user vers un autre, de renommer un fichier après une copie, de copier des caractères d'un fichier vers un périphérique de sortie (ex : imprimante), de transférer des données d'un périphérique d'entrée vers un périphérique de sortie

Tous les attributs et les mots de passe validés d'un fichier traité par l'utilitaire, sont recopiés sur le support de destination.

Si un fichier est protégé, vous pouvez pour le copier indiquer dans votre ligne de commande le mot de passe de ce fichier.

Syntaxe : PIP d : <Gn> = fichier source <options>

PIP fichier destination <Gn> = d : <options>

PIP fichier destination = fichier source <o>

Explications :

La première syntaxe indique à l'utilitaire de copier le fichier source du drive pris par défaut vers le drive spécifié par le paramètre d

L'option <Gn> permet la recopie de votre fichier sur le user spécifié par le paramètre n.

Exemple : A > PIP A : <G2> = HELP.COM

Recopie du fichier HELP.COM du drive A USER 0 sur le drive A USER 2

La seconde formulation est une variante de la première. PIP lit le fichier source sur le drive spécifié par d, le copie sur le drive pris par défaut ou spécifié par un paramètre optionnel et le recopie sous le nom adopté pour le fichier destination.

La troisième formulation vérifie la possibilité de renommer le fichier après la copie. Vous pouvez le copier sur le même drive, même USER ou sur un autre drive, autre USER.

Avant de lancer une copie par PIP vérifiez la place disponible sur votre disquette destination en utilisant la fonction DIR et SHOW et en utilisant s'il n'y a pas assez de place, la fonction ERASE.

La copie s'effectue sur un fichier portant l'extension \$\$\$ qui ne sera mise à jour que si la copie est correctement réalisée.

Si vous recopiez un fichier sur un fichier du même nom mais portant l'extension RO (lecture seule) l'utilitaire vous demande si vous désirez le supprimer.

Exemples :

A > PIP B : A : TEST.DAT

A > PIP B : TEST.DAT = A :

A > PIP B : TIST.DAT = A : TEST.DAT

Le premier exemple recopie le fichier TEST.DAT du drive A sur le B.
Le second est une formulation différente de la première ligne de commande.
Le troisième exemple recopie le fichier TEST.DAT du drive A sur le drive B et le renomme avec le nom TIST.DAT

Copies multiples :

Syntaxe : PIP d : <Gn> = <d:> options

Explications :

Si vous spécifiez un ou plusieurs caractères spéciaux dans la ligne de commande, vous pouvez faire une copie multiple qui s'effectuera fichier par fichier.
L'utilitaire affiche à chaque copie de fichier le nom du fichier en cours de copie.

Exemples :

A > PIP B : = A : *.COM

Recopie du drive A vers le drive B de tous les fichiers du drive A portant l'extension .COM

A > PIP B : = A : *.*

recopie complète de toute la disquette A sur la disquette B

A > PIP B : = A : PROG????.*

Recopie de tous les programmes commençant par PROG et portant une extension quelconque.

Combinaison de fichiers :

Syntaxe : PIP fichier destination <Gn> = fichier source <o> , fichier source <o> ..

Explications :

Cette formulation de la commande permet de spécifier plusieurs noms de fichiers dans la ligne de commande.

PIP copie les fichiers indiqués de la gauche vers la droite et les combine pour ne former qu'un seul fichier destination.

On appelle cette procédure concaténation.

Il est possible de spécifier pour les fichiers sources une ou plusieurs options.

Certaines options de commande obligent l'utilitaire à exécuter la copie caractère par caractère , PIP recherche alors un CTRL-Z pour déterminer la fin de fichier.

Toutes les options de PIP forcent le transfert caractère par caractère sauf les suivantes.

A, C, Gn, K, O, R, V et W

Une copie de données d'un ou vers un périphérique logique force la transmission caractère par caractère.

Si vous désirez faire une concaténation de fichier en langage machine, vous devez utiliser l'option O dans la commande des fichiers sources. Celle-ci inhibe la recherche du code CTRL-Z dans le dernier fichier.

Exemple :

A > PIP NEWLIFE=FILE1, FILE2, FILE3

Concaténation des fichiers FILE 1, 2 et 3 pour ne former qu'un seul et même fichier appelé NEWFILE.

A > PIP B : X.BAS=Y.BAS, B:Z.BAS

Le fichier Y.BAS est concaténé au fichier Z.BAS du drive B puis est renommé X.BAS avant achèvement de la commande.

Copie de fichier vers ou en provenance des auxiliaires logiques :

Syntaxe : PIP fichier destination < Gn > = fichier source < o >

AUX :	AUX : < 0 >
CON :	CON : < 0 >
PRN :	NUL :
LST :	EOF :

Exemples :

Cette ligne de commande permet la copie d'un fichier vers un auxiliaire, d'un auxiliaire vers un fichier ou d'un auxiliaire vers un auxiliaire.

Les auxiliaires définis doivent obligatoirement être en rapport avec leur utilisation (pour une lecture —> périphérique de sortie).

La syntaxe avec les deux points doit être respectée pour un auxiliaire afin qu'il n'y ait pas de confusion avec un fichier.

CTRL-C permet l'arrêt de la copie vers un auxiliaire.

Les périphériques logiques sont nommés :

- CON : console en entrée ou sortie. Pour une utilisation normale utilisez la vidéo en sortie et le clavier en entrée
- AUX : périphérique auxiliaire
- LST : généralement imprimante
- NUL : périphérique source produit 64 zéros
- EOF : périphérique source produisant un CTRL-Z en fin
- PRN : périphérique d'impression

Exemple :

A > PIP PRN : = CON:, MYDATA.DAT

Les caractères en provenance de la console d'entrée (clavier) sont directement envoyés à l'imprimante. Pour arrêter l'exécution de la commande tapez CTRL-Z.

Note : lorsque le périphérique logique est CON, les seuls caractères de contrôle vidéo sont Enter et ligne feed.

A > PIP B : FUNFILE.SUE = CON :

Création du fichier FUNFILE.SUE sur le drive B avec pour information tout ce que vous entrer au clavier.

Sortie par CTRL-Z

A > PIP LST : = CON. :

Envoi des caractères entrés au clavier vers l'imprimante ; sortie par CTRL-Z

A > PIP LST : = B : TEST.TXT < T8 >

Impression du fichier TEXT.TXT sur l'imprimante ou chaque caractère de tabulation est considéré comme huit espaces.

Modé de commande multiple :

Syntaxe : PIP

Explications :

Cette forme de commande permet de passer sous le prompt de l'utilitaire PIP et d'entrer plusieurs lignes de commandes.

La sortie du mode se fait en appuyant sur <Enter> lorsque le prompt est affiché.

Exemple :

```
A > PIP
CP/M 3 PIP VERSION 3.0
*B := A : *.* copie complète de A vers B
```

Les options de PIP :

Explications :

L'utilisation d'option(s) avec l'utilitaire permet de traiter les tabulations, de transférer des caractères (échange poids fort - poids faible), d'extraire des parties de texte, de vérifier si une copie est correcte...

Les options suivantes sont représentées par n qui indique un nombre et s qui représente une séquence de caractères terminée par CTRL-Z. Toute option doit être entourée par des accolades.

Pour les options utilisant des numériques, aucun blanc ne doit être inscrit entre l'option et le numérique.

Options	Fonctions
A	Copie seulement les fichiers qui ont été modifiés depuis la dernière copie. Cette option est appelée l'option archive.
C	Permet la copie fichier par fichier avec demande si copie du fichier avant le lancement.
Dn	Supprime tous les caractères supérieurs à la colonne n. Cette option permet de reconfigurer un texte pour une utilisation sur une machine ne possédant pas le même nombre de colonnes.
E	Autorise le listing du fichier à l'écran avant la copie.
F	Retire tous les form-feeds d'un fichier
Gn	Permet la copie de fichier vers un user spécifié par le paramètre n. Ce paramètre doit être compris entre 0 et 15.
H	Contrôle le format .HEX. Lors d'un transfert de fichier PIP vérifie si toutes les données du fichier sont au format INTEL.
I	Ignore les enregistrements nuls. Cette option force l'option précédente.
L	Transformation des majuscules du fichier source en minuscules sur le fichier destination.
N	Numérote les lignes copiées en sortie. La numérotation se fait en commençant par 1 et par incrémentation de 1. Le signe : suit chaque numéro de ligne
O	Force le transfert des fichiers objet. Ignore les CTRL-Z de fin de fichier normal.

- Pn** Sélection de format de page. Si ce paramètre modifie le transfert du fichier source, pip inclue un form feed au début du fichier destination et à chaque n lignes. Si $n = 1$ ou n'est pas spécifié, pip insère un form feed à chaque soixantième ligne.
- Qs** Permet de quitter la copie dès que la chaîne de caractères spécifiés par le paramètre s est atteinte. La chaîne doit obligatoirement être suivie d'un CTRL-Z
- R** Lecture et copie des fichiers portant l'attribut SYS. Si cette option n'est pas spécifiée, pip ne recopie pas les fichiers portant l'attribut sys. Si elle l'est la copie des fichiers est exécutée avec leurs attributs.
- Ss** Débute la copie du fichier source dès que la chaîne s est atteinte. La chaîne de caractères doit être terminée par un CTRL-Z. Si cette option est associée avec l'option Qs, pip peut extraire une partie de fichier à n'importe quel endroit du fichier source.
- Tn** Remplace chaque tabulation du fichier destination avec autant d'espaces nécessaires pour arriver à une colonne divisible par le paramètre n. Le code de tabulation est CTRL-I
- U** Transformation des minuscules du fichier source en majuscules sur le fichier destination.
- V** Vérifie que les données qui viennent d'être copiées l'ont été sans erreurs par comparaison entre le fichier source et le fichier destination.
- W** Permet la copie sans question des fichiers portant l'attribut lecture seule (R/O). Généralement pip à la rencontre d'un de ces fichiers pose une question afin de savoir si l'utilisateur désire réellement copier ce fichier. Cette option permet l'inhibition de la question et la copie obligatoire.
- Z** Positionne à zéro le bit de parité d'un fichier source ascii.

Exemples :

A > PIP NEWPROG.BAS=CODE.BAS < L > , DATA.BAS < U >

Création du fichier NEWPROG.BAS sur le drive A par concaténation des fichiers CODE.BAS et DATA.BAS du même drive. Pendant la copie les majuscules du fichier CODE.BAS sont transformées en minuscules alors que le contraire est appliqué au fichier DATA.BAS

A > PIP CON:=WIDELIFE.BAS < D80 >

Cette commande écrit le fichier WIDELIFE.BAS du drive A vers le périphérique console et élimine tous les caractères se positionnant à une colonne supérieure à 80.

A > PIP B:=LETTER.TXT < E >

Copie du fichier du drive A vers le B avec affichage sur l'écran durant la copie.

A > PIP LST:=B:LONGPAGE.TXT < FP65 >

Recopie du fichier du drive B vers l'imprimante avec insertion de form-feed au début du fichier et à chaque 65ème ligne.

A > PIP PORTION.TXT=LETTER.TXT < SCher Monsieur
CTRL-Z QAmicalement CTRL-Z >

Cette commande permet de prendre du fichier LETTER.TXT la portion de texte comprise entre les chaînes Cher Monsieur et Amicalement.

B > PIP B:=A:*.COM < VWR >

Copie de tous les fichiers .COM du drive A vers le drive B avec vérification copie de tous les fichiers en lecture seule et de tous les fichiers portant l'attribut SYS.

PUT :

Syntaxe : PUT CONSOLE <OUTPUT TO> nom de fichier <o>
PUT PRINTER <OUTPUT TO> nom de fichier <o>
PUT CONSOLE <OUTPUT TO> CONSOLE
PUT PRINTER <OUTPUT TO> PRINTER

Explications :

Cette commande permet de stocker dans un fichier nommé dans la commande, toutes les données prises en sortie sur un des périphériques désigné.

En y associant l'option système, elle permet de créer un fichier portant l'attribut .SYS jusqu'à rencontre de la commande PUT CONSOLE ou PUT PRINTER.

Options :

- ECHO : indique l'écho console du fichier en train d'être créé.
- NO ECHO : aucun écho sur la console
- FILTER : transformation des codes de contrôle en caractères imprimables
- NO FILTER: pas de transformation des caractères de contrôle
- SYSTEM : La sortie de la console vers le fichier nommé désigné se fera jusqu'à ce que l'utilisateur entre la commande PUT CONSOLE qui renverra la console à la sortie console.

Exemples :

A > PUT CONSOLE OUTPUT TO FILE XOUT < ECHO >

Sortie de la console vers un fichier XOUT avec mode écho

A > PUT CONSOLE OUTPUT TO FILE DIR.PRN < ECHO >
Putting console output to file : DIR.PRN
A > DIR

Sortie de la console vers le fichier DIR.PRN ce qui à pour effet si vous entrez la ligne de commande DIR de créer un fichier DIR.PRN réplique exacte de votre directory.
Pour vérifier faites : A > TYPE DIR.PRN (cr)

A > PUT PRINTER OUTPUT TO FILE XOUT
A > MYPROG

Dérivation de la sortie imprimante vers le fichier XOUT. le lancement du programme MYPROG permettra le stockage des informations destinées à l'imprimante dans le fichier.

A > PUT PRINTER OUTPUT TO FILE XOUT2 < ECHO, SYSTEM >

Idem précédemment mais écho sur la console et arrêt de la dérivation par ligne de commande PUT PRINTER OUTPUT TO PRINTER

A > PUT CONSOLE OUTPUT TO CONSOLE

Redérivation de la console vers la console

RENAME :

Syntaxe : **RENAME** nouveau fichier = ancien fichier

Explications :

La commande **RENAME** ou **REN** en abrégé permet de changer le nom d'un fichier déjà existant.

Cet utilitaire ne change que le nom du fichier au niveau du directory et ne recopie pas le fichier.

Si le fichier est inexistant quand vous le renommez, le système vous renverra un message d'erreur : No file

Si un fichier du même nom est déjà existant vous pouvez le détruire avant de renommer votre autre fichier. Dans ce cas le système vous demande que faire, avant d'entamer une procédure irrémédiable.

Exemples :

A > **REN *.TX1=*.TEX**

Changement des options de tous les fichiers en .TEX en fichier portant l'option .TX1

A > **RENAME TEST.BAS = TIST.BAS**

Changement du fichier TIST.BAS en TEST.BAS

A > **RENAME**

Enter New Name : TEST.BAS

Enter Old Name : TIST.BAS

Idem précédemment mais avec aide du système

RMAC :

Syntaxe : **RMAC** nom de fichier < \$Rd, \$Sd, \$Pd >

Explications :

L'utilitaire **RMAC** est un macro assembleur permettant l'assemblage des fichiers de type .ASM en fichier .REL capable d'être liés pour donner des fichiers de commande .COM

Les options permettent de définir la destination des fichiers. L'option d détermine le drive sur lequel va s'inscrire le fichier concerné, X permet d'obtenir une sortie consoleé, P une sortie sur l'imprimante, Z rien en sortie

OPTIONS	d—OPTION DE SORTIE
R drive pour fichier REL	A—O, Z
S drive pour fichier SYM	A—O, X, P, Z
P drive pour fichier PRN	A—O, X, P, Z

Vous ne pouvez pas diriger le fichier .REL vers la console ou l'imprimante car le fichier .REL n'est pas en ascii.

Exemple :

A > **RMAC TEST \$PX SB RB**

RMAC assemble le fichier TEST.ASM du drive A dirige le fichier .PRN vers la console, dirige le fichier .SYM sur le drive B ainsi que le fichier relogeable TEST.REL.

SAVE :

Syntaxe : SAVE

Explications :

La commande SAVE permet la copie d'une zone mémoire vers un fichier. Pour l'utiliser, vous devez lancer l'activation de l'utilitaire, utiliser une commande de chargement de fichier en mémoire et exécuter votre programme. Avant le retour au système, l'utilitaire SAVE vous invite à définir les zones mémoire à sauvegarder.

Exemple :

```
A > SAVE                : activation de l'utilitaire SAVE
A > SID dump.com         : chargement du programme
#$O                      : exécution
```

SAVE Ver 3.0

File (or RETURN to exit) ? dump2.com

Delete dump2.com? Y

From?100 : copie du contenu des adresses mémoire comprises entre
: 100 hexa et 400 hexa

To? 400

```
A >                : retour au CP/M
```

SET :

Syntaxe : SET <options>
SET d : <options>
SET fichier <options>

Explications :

La commande permet l'initialisation des mots de passe et l'activation de contrôle de la date sur les fichiers. Elle positionne également les attributs tels que lecture/écriture, lecture seule, fichiers SYS.

Afin de permettre le marquage de la date et de l'heure sur vos fichiers, vous devez lancer la commande INITDIR.COM

Marquage par attribut :

SET fichier <attributs/options>

Permet la définition d'attributs aux fichiers.

OPTIONS	EXPLICATIONS
DIR	Positionne l'attribut fichier DIR
SYS	Positionne l'attribut fichier SYS (permet le listing du fichier avec la commande DIRS)
RO	Attribut fichier n'autorisant que la lecture
RW	Attribut fichier autorisant la lecture/écriture
ARCHIVE=OFF	Positionne l'attribut archive sur OFF. Cela permet au fichier de ne pas être copié si pip n'est pas utilisé avec l'option A (archivage)
ARCHIVE=ON	Positionne l'attribut archive sur ON. Cette option autorise la recopie du fichier.
F1=ON/OFF	Active ou désactive l'attribut F1 sur le user défini
F2=ON/OFF	Active ou désactive l'attribut F2 sur le user défini
F3=ON/OFF	Active ou désactive l'attribut F3 sur le user défini
F4=ON/OFF	Active ou désactive l'attribut F4 sur le user défini

Exemple :

A > SET MYLIFE.TEX <RO SYS>

Donne l'option lecture seule et système au fichier MYFILE.TEX

A > SET MYLIFE.TEX <RW DIR>

Donne l'option écriture lecture avec l'option DIR permettant l'affichage du nom de fichier lors d'une commande DIR.

Attribut de drive :

Syntaxe : SET <d:> <RO>
SET <d:> <RW>

Explications :

Spécifie l'attribut de drive en lecture seule ou en écriture lecture. Si un drive est spécifié en lecture seule l'utilitaire pip ne peut aller écrire dessus. Les commandes REN et ERA sont inefficaces. Si vous entrez un CTRL-C sous le système, tous les drives sont remis en attribut écriture/lecture.

Exemple :

A > SET B <RO>

Définit le drive b en lecture seule.

Assignation d'un label sur le disque :

Cette commande permet d'assigner un nom au drive pris par défaut ou au drive spécifié dans la commande.

Ce nom peut être défini sur 8 caractères et 3 caractères d'extension tel un fichier normal. Le nom défini par défaut est LABEL.

A > SET <NAME=DIK100>

Assignation d'un mot de passe au label :

La première forme d'assignation nomme un drive avec un label. Cette commande permet d'assigner un mot de passe au disque ce qui le rend inaccessible aux autres utilisateurs en théorie.

```
A> SET <PASSWORD=SECRET>
A> SET <PASSWORD=(cr)>
```

La première commande assigne le mot secret comme mot de passe, la deuxième annule le mot de passe existant.
Autorisation et annulation de protection d'un disque :

```
A> SET <PROTECT=ON>
A> SET <PROTECT=OFF>
```

La première forme de commande valide la protection par mot de passe. La seconde forme la dévalide. Une fois que vous avez assigné un mot de passe à votre disque et que la protection est validée vous êtes dans une configuration permettant d'assigner des mots de passe à vos fichiers. Vous vérifiez si un disque est protégé en utilisant la commande SHOW qui liste le label.

Assignation de mot de passe aux fichiers :

```
A> SET MYLIFE.TEX PASSWORD=MYFIL
```

La commande permet l'assignation du mot de passe MYFIL au fichier MYLIFE.TEX le mot de passe peut être d'une longueur maximale de huit caractères et un changement des minuscules en majuscules est effectué avant l'assignation. Notez le mot de passe assigné au fichier. En cas d'oubli vous ne pourriez accéder à votre fichier.

Assignation de mot de passe de protection sur fichiers :

```
A> SET MYLIFE.TEX <PROTECT=READ>
A> SET MYFILE.TEX <PROTECT=WRITE>
A> SET MYLIFE.TEX <PROTECT=DELETE>
A> SET MYFILE.TEX <PROTECT=NONE>
```

L'assignation du mot de passe READ oblige l'utilisateur à le nommer en lecture, écriture, destruction et changement de nom. WRITE prend les mêmes conditions mis à part pour la lecture où le mot de passe n'a pas besoin d'être spécifié. DELETE oblige le mot de passe pour une destruction de fichier ou un changement de nom. NONE indique qu'il n'y a pas de mot de passe.

Assignation d'un mot de passe par défaut :

```
A> SET *.TEX <PASSWORD=SECRET, PROTECT=WRITE>
```

Assignation du mot de passe SECRET à tous les fichiers portant l'extension .TEX. Le mot de passe est validé en écriture

Utilisation de la date et de l'heure :

```
Syntaxe : SET <CREATE=ON>
          SET <ACCESS=ON>
          SET <UPDATE=ON>
```

Explications :

- <CREATE = ON> Activation de la fonction de stockage de la date et de l'heure sur le drive pris par défaut. Pour enregistrer la date et l'heure d'un fichier, la fonction create doit avoir été validée.
- <ACCESS = ON> Activation de la fonction access sur le drive pris par défaut. Les options access et create sont exclusives ce qui veut dire que l'activation de access entrainera la désactivation de create.
- <UPDATE = ON> Cette activation entraine l'enregistrement de l'heure à laquelle le fichier a été modifié.

Pour inhiber ces fonctions, vous devez lancer la commande INITDIR.COM qui permet un reformatage du directory au niveau de l'heure et de la date de chaque fichier. Bien qu'il y ait 3 modes de stockage de date et d'heure, seulement deux peuvent être associés entre eux.

A> SET <ACCESS=ON>

La fonction DIR suivie de l'option FULL permet de visualiser les dates et heures de chaque fichier.

A> DIR <FULL>
Directory for drive A :

Name	Bytes	Recs	Attributs	Prot	Update	Access
ONE .TEX	9 k	71	Dir RW	None		08/03/85 10:56
THREE .TEX	12 k	95	Dir RW	None		08/05/85 15:45
TWO .TEX	10 k	76	Dir RW	None		08/10/85 09:13

A> SET <CREATE=ON, UPDATE=ON>

La fonction DIR suivie de l'option FULL permet de visualiser les dates et heures et les UPDATES de chaque fichier.

A> DIR <FULL>
Directory for drive A:

Name	Bytes	Recs	Attributes	Prot	Update	Acces
ONE .TEX	9 k	71	Dir RW	none	08/04/85 14:01	08/03/85 10:56
THREE .TEX	12 k	95	Dir RW	none	08/04/85 15:45	08/05/85 15:45
TWO .TEX	10 k	76	Dir RW	none	08/04/85 16:30	08/10/85 09:13

A> SETDEF <TEMPORARY=B:>

Désigne le drive B comme drive provisoire pour les fichiers.

Définition de l'ordre de recherche sur disque :

A> SETDEF C:,*

L'exemple indique au système que la recherche doit commencer sur le drive C et si le fichier n'est pas trouvé, la recherche doit continuer sur le drive pris par défaut.

Définition de l'ordre de recherche d'un type de fichier:

A> SETDEF <ORDER(SUB,COM)>

La commande indique au système que la recherche d'un fichier doit commencer par les fichiers .SUB et se terminer s'il n'est pas trouvé par une recherche sur les fichiers en .COM

Validation/dévalidation de l'affichage :

A> SETDEF <DISPLAY>

Valide l'affichage, lorsqu'un fichier est appelé du drive sur lequel le fichier se trouve, du type de fichier et du numéro de user.

A> PIP

A:PIP COM affichage des paramètres du fichier
CP/M 3 PIP VERSION 3.0

*

4A> PIP

A:PIP COM (User 0) affichage des paramètres du fichier
CP/M 3 PIP VERSION 3.0 + numéro de user

*

Inhibition du mode : A> SETDEF <NO DISPLAY>

Validation/dévalidation du mode page :

A> SETDEF <PAGE>

La commande indique au système de passer en mode page.

SHOW :

Syntaxe : SHOW <d:><SPACE/LABEL/USERS/DIR/DRIVE>

Explications :

La commande SHOW permet de visualiser des informations sur les disques.

- la place restante sur le disque
- le label d'un disque
- le numéro de user courant
- le nombre de fichiers contenus dans chaque user
- le nombre d'entrées directory libres sur le disque
- les caractéristiques du disque

Affichage de la place disponible :

```
A> SHOW B:
B: RW, Space: 9,488k
A> SHOW
A: RW, Space: 4k
B: RW, Space: 9,488k
```

Le premier exemple visualise la place disponible sur le drive B ainsi que son mode d'accès en lecture/écriture.

Le deuxième exemple visualise les mêmes caractéristiques mais pour les deux drives.

Si vous lancez une commande show sans avoir pratiqué d'accès sur le drive B la commande ne vous donnera que les paramètres du drive A.

Dans ce cas faites SHOW B: ou DIR B: et SHOW

L'accès au drive B permet au système de connaître les caractéristiques du drive et le stockage en mémoire de ces paramètres.

Affichage du label du disque :

```
A> SHOW B: <LABEL>
```

La commande vous permet de visualiser le nom du drive B ainsi que ses caractéristiques au niveau des mots de passe et des dates et heures.

Affichage des informations concernant un user :

```
A> SHOW B: <USERS>
```

La commande vous permet de connaître le nombre de user utilisés sur le drive B ainsi que la place occupée par chacun d'eux.

Affichage du nombre d'entrées directory libres :

```
A> SHOW B: <DIR>
```

La commande vous permet de connaître le nombre d'entrées directory disponibles sur votre disque.

Affichage des caractéristiques de disque :

```
A> SHOW B: <DRIVE>
```

La commande vous renvoie les caractéristiques physiques et logiques de votre disquette.

SID :

Syntaxe : SID fichier programme, fichier sym

Explications :

L'utilitaire SID, Symbolic Instruction Debugger permet de tester vos programmes développés pour le processeur 8080.

L'utilitaire autorise les points d'arrêt, l'exécution, le désassemblage en symboles, l'assemblage, la visualisation d'une partie de la mémoire et des instructions propres à la gestion des fichiers.

En plus du chargement du fichier spécifié dans la ligne de commande, sid charge en mémoire un programme de test. Les commandes de contrôle sont CTRL-S pour arrêter le défilement vidéo, CTRL-Q pour repartir, CTRL-C pour terminer. Pour exprimer des nombres décimaux, faites précéder la valeur par un # ex : #65534=FFFEh

NOM	SYNTAXE	FONCTION
Assemble	As	Permet d'entrer des instructions en langage assembleur .s spécifie l'adresse de départ.
Call	Cs <d>>>	Call mémoire .s est l'adresse du call, b est la valeur du registre BC et d est la valeur du registre DE
Display	D <W> <s> <f>	Dump mémoire en hexa et en ascii. W indique un format 16 s est l'adresse de départ. f est l'adresse de fin.
Load	Epgm <sym-pgm>	Chargement d'un programme et d'une table de symboles pour exécution
Load	E* sym-pgm	Charge une table de symboles
Fill	Fs, f, d	Remplissage mémoire avec un octet sur 8 bits. s adresse de départ f adresse de fin d octet de remplissage
Go	G <p> <a> >>	Début l'exécution à l'adresse p a est un point d'arrêt b est un autre point d'arrêt GO exécute un warm boot
Hex	H Ha Ha, b	Affiche tous les symboles et adresses en hexa. a affiche sa valeur en hexa, décimal et ascii a, b affiche la somme et la différence des deux valeurs
Input	Icommand	Introduction de la ligne de commande
List	L <s> <f>	Liste les instructions sous forme de mnémoniques de l'adresse s à f

NOM	SYNTAXE	FONCTION
Move	Ms, h, d	Transfert mémoire de l'adresse s (pour début) à h (pour fin) à l'adresse mémoire d
Pass	P <p <, c>>	Pass point set.p est un point d'arrêt permanent et c le compteur de nombre de passe
Read	Rnom <, d>	Lecture d'un fichier disque
Set	S <W> s	Remplissage mémoire adresse s avec w = valeur de 16 bits
Trace	T <n <, c>>	Trace l'exécution du programme s est un pas d'exécution c est l'adresse de départ
Trace	T <W><n <, c>>	Trace mais n'affiche pas les sous routines (call)
Untrace	U <W><n <, c>>	Exécution sans trace sans suite des call ou n est le pas et c l'adresse d'entrée
Value	V	Affiche l'implantation mémoire après lecture disque
Write	Wnom <, s, f>	Ecriture disque d'une portion mémoire comprise entre s et f
Examine	X <f><r>	Examine les registres f peut être le flag C, E, I, M, Z r est le registre A, B, D, H, P, S,

Exemples :

A> SID

SID affiche son prompt lorsqu'il est prêt à recevoir des commandes

A> B:SID SAMPLE.HEX

Chargement du SID et du programme SAMPLE.HEX en mémoire sid affiche :

```
NEXT MSZE PC END
nnnn mmmm pppp eeee
```

nnnn indique l'adresse mémoire libre derrière le programme chargé.
 mmmm adresse mémoire après le plus grand programme (à l'init = nnnn)
 pppp valeur de PC (compteur de programme)
 eeee indique l'adresse de fin de TPA

#DFE00+#128+5

Visualisation de la mémoire de l'adresse FE80 (FE00 + 128) jusqu'à l'adresse FE85 (FE00 + 128 + 5)

Les utilitaires de SID :

HIST.UTL et TRACE.UTL sont des programmes spéciaux utilisés par SID pour faciliter la mise au point.

Pour les charger entrez : SID nom.UTL nom représente le nom de l'utilitaire des codes spécifiés dans le programme testé.

L'utilitaire TRACE.UTL obtient une trace inverse des instructions qui signalent un point d'arrêt. Vous pouvez obtenir jusqu'à 256 instructions entre deux points de passe en mode U et T.

SUBMIT :

Syntaxe : SUBMIT <nom du fichier> <argument> ...<argument>

Explications :

La commande permet d'exécuter un groupe de commandes contenues dans un fichier .SUB

Généralement vous entrez vos commandes ligne par ligne. Si vous désirez entrer une séquence de commandes (plusieurs lignes de commande) vous devez utiliser la commande SUBMIT.

Le fichier est spécifié par son nom et une extension .SUB. Lorsque vous lancez une commande submit en précisant le nom d'un fichier .SUB, le programme va lire ce fichier avant d'envoyer ligne par ligne les commandes au CP/M comme si vous les entriez une par une à la main.

La création des lignes de commandes d'un fichier .SUB se fait avec l'utilitaire ED.COM (se reporter à l'annexe)

Il est possible de passer des arguments aux lignes de commande exécutées pendant le lancement de SUBMIT. Le premier argument remplace chaque occurrence de \$1 dans le fichier, le second chaque \$2 ...etc... jusqu'à \$9.

Pour exemple si votre fichier START.SUB contient les données :

```
ERA $1.BAK
DIR $ 1
PIP $1=A:$2.COM
```

Entrez : A> SUBMIT START SAM TEX

L'argument SAM est substitué à chaque \$1 dans le fichier .SUB et TEX à chaque \$2. SUBMIT crée alors un fichier contenant ces modifications et exécute les commandes .

Le listing du fichier donne :

```
ERA SAM.BAK
DIR SAM
PIP SAM=A:TEXT.COM
```

Si vous entrez plus d'arguments qu'il n'y a de \$ dans le fichier .SUB, les arguments en trop ne seront pas pris en compte.

Si vous désirez inclure un \$ dans vos lignes de commande vous devez au moment de la création le répéter deux fois.

Ex : si votre fichier AA.SUB contient :

```
MAC $1 $$$2
```

et que vous entrez la commande :

A> SUBMIT AA ZZ SZ

Le fichier deviendra :

```
MAC ZZ $SZ
```


Programmation de lignes de commandes d'entrée :

Un fichier .SUB peut contenir des commandes d'entrée. Chaque ligne d'entrée est alors précédée par le signe <
Ex. : listing d'un fichier .SUB possédant des lignes d'entrée

```
PIP
<B:=*.ASM
<CON:=DUMP.ASM.
<
DIR
```

Les trois lignes suivant la commande pip sont des commandes d'entrée de la fonction PIP. La troisième ne comportant qu'un < indique un (cr) pour retour au CP/M en sortie de l'utilitaire PIP. Si le fichier contient plus de commandes d'entrée que nécessaire SUBMIT renvoie un message d'erreur du type : Warning : Programm Input ignored.

Si le fichier ne contient pas assez de commandes d'entrée, SUBMIT autorise l'entrée clavier.

Si vous désirez entrer des caractères de contrôle dans un fichier .SUB, vous devez utiliser les conventions qui sont de placer une flèche vers le haut avant la lettre correspondant au caractère de contrôle.

Le fichier sub accepte :

- toute commande correcte de CP/M
- toute commande avec paramètres de SUBMIT
- toute commande d'entrée
- toute ligne d'entrée avec paramètres \$

Fichier de commande débutant après le boot :

Chaque fois que vous effectuez un boot sur votre système, celui-ci regarde s'il existe un fichier nommé PROFILE.SUB à exécuter. Si ce fichier n'existe pas le système rend la main à l'utilisateur. S'il existe, le système exécute les unes après les autres les commandes contenues dans ce fichier avant de rendre la main à l'utilisateur.

Exemple :

Si vous désirez mettre à jour la date de votre guépard après chaque reset, vous pouvez créer un fichier PROFILE.SUB contenant la commande DATE SET qui vous permettra une remise à l'heure de votre appareil.

TYPE :

Syntaxe : TYPE <fichier <PAGE/NOPAGE>>>

Explications :

La commande TYPE permet l'affichage à l'écran du contenu ascii d'un fichier. L'option page indique au système que vous désirez un arrêt après chaque scrolling d'écran complet. L'option NOPAGE permet le listing du fichier sans arrêt entre scrolling.

Si vous omettez le nom du fichier le système se chargera de vous le demander en affichant un message de rappel.

Rappel :

Les commandes PAGE et NOPAGE peuvent être définies par défaut par la commande SETDEF.

Exemple :

A> TYPE MYPROG.TEX

Listing du fichier ascii MYPROG.TEX à l'écran.

L'arrêt momentané du listing peut être obtenu par appui sur CTRL-S (cf codes de contrôle). L'arrêt total est obtenu par appui sur CTRL-C.

USER :

Syntaxe : USER <number>

Explications :

La commande permet la mise à jour du USER sur lequel vous désirez travailler. A l'initialisation du système le USER pris par défaut est le 0. Vous pouvez sélectionner les USER(s) entre 16 disponibles. (0 à 15)

Rappel :

Si vous avez validé un fichier avec l'attribut SYS tous les USERS du système pourront y avoir accès.

Exemple :

A> USER	commande de sélection de USER
Enter User # : 5	sélection du USER 5
5A>	passage sur USER 5

XREF :

Syntaxe : XREF <d:> nom de fichier <\$P>

Explications :

La commande XREF permet la construction d'une table de référence sommaire de l'utilisation des variables dans un programme. Cette commande requiert les fichiers portant l'extension .PRN et .SYM des fichiers produits par MAC et RMAC.

Les deux fichiers doivent porter le même nom que dans la commande.

XREF crée un fichier de type XRF.

Exemple :

A> XREF B:MYPROG

Dans cet exemple la commande XREF se trouve sur le drive A et les fichiers .PRN et .SYM se trouvent eux sur le drive B. XREF produira un fichier MYPROG.XRF sur le drive B

A> XREF B:MYPROG \$P

L'option \$P permet la sortie du fichier sur l'imprimante.

ANNEXE A EDITEUR ED.COM

A1 - INTRODUCTION :

Afin de commander votre micro-ordinateur vous devez assez souvent lui entrer des commandes sous forme de données contenues dans un fichier.

L'utilitaire requis s'appelle un éditeur et permet la transformation de vos commandes en fichier.

L'éditeur ED.COM permet outre la création de nouveaux fichiers de données, la modification d'anciens fichiers déjà existants.

A2 - LANCEMENT DE ED.COM :

Syntaxe : fichier en entrée <d:/fichier en sortie>

Explications :

ED accepte une syntaxe ou deux paramètres peuvent être optionnels. Ce sont les numéros de drive sur lesquels résident les fichiers et le nom de fichier en sortie de l'éditeur.

Exemples :

A> ED TEST.TEX

La commande permet l'édition du fichier TEST.TEX ou la création de ce dernier sur le drive pris par défaut par votre système.

A> ED TEST.TEX B:

Edition du fichier à partir du drive B

A> ED TEST.TEX TOST.TEX

Si le fichier optionnel existe déjà l'éditeur renvoie un message d'erreur et retourne au CP/M. Dans cet exemple, le fichier TEST.TEX (déjà existant) est considéré comme fichier d'entrée et le fichier TOST.TEX comme fichier de sortie (celui qui contiendra les données après édition).

Si le fichier n'existe pas, l'éditeur vous renvoie un message avant d'afficher son prompt.

A> ED TEST.TEX
NEW FILE

.*

Note :

Avant de passer en édition utilisez la commande SHOW pour vérifier si la place disponible sur votre disquette est suffisante pour contenir les données que vous désirez entrer.

A3 - FONCTIONNEMENT :

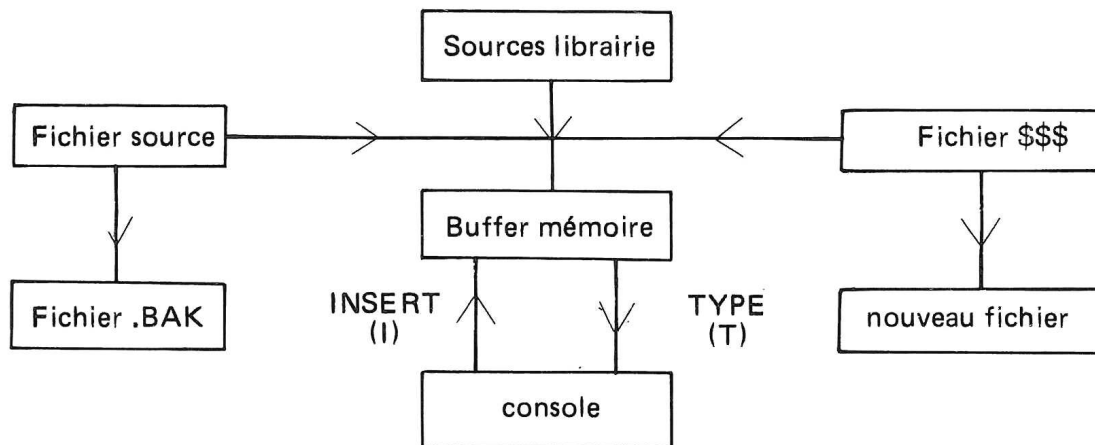
L'édition ou la création d'un fichier passe par un buffer de transit mémoire. Lorsque vous lancez l'éditeur, le buffer est vide même si vous éditez un fichier déjà existant. Vous devez donc charger des morceaux du fichier dans le buffer afin d'y apporter des modifications.

Pendant l'édition, et si le buffer est plein, l'éditeur écrit les données dans un fichier du même nom que celui assigné mais portant l'option \$\$\$.

Si une erreur et un retour impronptu au CP/M se produit en cours d'édition vous devrez supprimer ce fichier \$\$\$.

En fin d'édition, l'éditeur renomme le fichier d'entrée en prenant le même nom et en y accolant l'option .BAK (ce qui permet de conserver l'original) et renomme le fichier .\$\$\$ en utilisant le nom indiqué comme fichier de sortie. (ceci est valable si le nom du fichier d'entrée correspond au fichier sortie).

Si le nom du fichier d'entrée diffère du fichier en sortie, le système conservera le nom du fichier d'entrée pour sauvegarde de votre ancien fichier (pas de rename en .BAK).



L'illustration place la mémoire de transit entre la console et le fichier source et le fichier temporaire.

L'éditeur supporte des commandes permettant le transfert de lignes de texte entre la mémoire et le fichier temporaire et entre le fichier source et la mémoire.

Trois commandes permettent ces transferts et la cloture des fichiers pour devenir fichier .BAK et nouveau fichier.

Cde.	Fonction
nA	Fonction APPEND. permet de stocker des données en provenance du fichier source vers le fond de la mémoire n indique le nombre de lignes à transférer
nW	Fonction WRITE. transfère les n premières lignes de la mémoire dans le fichier temporaire.
E	Fin d'édition. copie de la mémoire sur le fichier temporaire et changement de nom des fichiers.

A3.1 - APPEND DE TEXTE EN MEMOIRE :

Lorsque vous lancez ED.COM, la mémoire de transit est vide. Vous pouvez utiliser la commande A pour transférer du texte en provenance du fichier source en mémoire.

Note :

Le début d'édition affiche : ce qui indique que la numérotation automatique des lignes est activée. Vous avez la possibilité de dévalider cette fonction.

V : valide la fonction de numérotation

— V : dévalide la fonction numérotation

OV : indique la place restante en mémoire sous la forme libre/total qui donne respectivement la place restante et la place disponible.

La commande APPEND :

Cette commande copie n lignes du fichier source vers la mémoire. Le signe n indiquant le nombre de lignes à charger en mémoire peut-être remplacé par un # qui permettra le chargement de tout le fichier en mémoire suivant la place disponible. (ce # est assumé comme 65535)

A3.2 - SORTIE DE L'EDITEUR :

La commande WRITE :

Cette commande permet l'écriture de lignes de texte contenues dans la mémoire de transit vers le fichier temporaire.

n indique le nombre de lignes à copier dans le fichier temporaire. Si n est supérieur à 0, ED transfère les n lignes correspondantes du buffer vers le fichier temporaire. Si n est nul, ED transfère la moitié du buffer sur le disque.

Vous pouvez aussi utiliser le # pour transférer tout le contenu de la mémoire sur le disque.

Note :

Après sauvegarde sur le disque par la commande W, vous devez utiliser la commande H pour rééditer les lignes venant d'être sauvegardées.

La commande EXIT :

Lorsque vous désirez quitter l'éditeur et que pour ce faire vous utilisez la commande E, ED transfère toutes les lignes de la mémoire et du fichier source non modifiées vers le fichier temporaire, détruit et renomme le fichier en .BAK si un fichier en .BAK existe déjà et renomme le fichier en .\$\$\$ avec l'option choisie dans la commande.

A4 - COMMANDES D'ÉDITION :

Nous venons de voir comment entrer et sortir de l'éditeur. Nous allons voir maintenant les commandes d'édition permettant de créer ou de modifier un fichier.

ED traite le fichier comme une suite de caractères groupés sous forme de lignes. ED édite ces lignes en fonction d'un pointeur imaginaire que nous appellerons (CP) (de l'anglais pointer character). Les commandes consistent en un argument numérique et une lettre de commande suivie d'un appui sur <Enter.>

- si l'argument numérique est oublié ED le prend égal à 1
- un argument négatif provoque une commande en sens inverse de la normale.
- si vous entrez un # comme argument, ED l'assume comme 65535
- certaines commandes acceptent l'argument 0

Cde	Fonction
B, -B	Déplace (CP) vers le début ou la fin du buffer mémoire.
nC, -nC	Déplace (CP) de n caractères vers l'avant ou en arrière de la mémoire.
nD, -nD	Retire n caractères après ou avant le pointeur mémoire.
I	Passage en insertion
lchaine CTRL-Z	Insertion d'une chaîne de caractères
nK, -nK	Efface n lignes après ou avant le pointeur (CP)
nL, -nL	Déplace le (CP) de n lignes vers l'avant ou vers l'arrière
nT, -nT	Imprime n lignes avant ou après le pointeur (CP)
n, -n	Déplace le (CP) de n lignes vers l'avant ou de n lignes vers l'arrière et affiche la ligne de destination.

A4.1 - DEPLACEMENT DU POINTEUR DE CARACTERES :

Commande B :

La commande B permet le déplacement de (CP) vers le début ou la fin de la mémoire. B autorise un déplacement avant, —B un déplacement arrière.

Commande C :

La commande déplace le (CP) vers l'avant ou l'arrière de n caractères.

Un argument positif déplace le pointeur vers le fond de la mémoire, un argument négatif vers le début.

Vous devez inclure dans le compte des caractères un retour chariot et un retour de ligne qui se trouvent à la fin de chaque ligne mais n'apparaissent pas à l'utilisateur.

Commande L :

Déplace (CP) vers l'avant ou l'arrière d'un nombre de lignes dépendant de l'argument n. Vous pouvez utiliser l'argument 0 pour déplacer le (CP) vers le début de la ligne courante.

Commande N :

Déplace le (CP) de n lignes et affiche la ligne de destination. Cette commande peut être abrégée en entrant seulement n et en appuyant sur <Enter> pour faire avancer le (CP).

A4.2 - AFFICHAGE DU CONTENU DE LA MEMOIRE :

La commande T supportant l'argument n permet un listing écran des lignes de texte mémorisées.

Si n est un argument négatif, l'affichage se fera sur n lignes avant le (CP). Si n n'est pas spécifié, la commande permet l'affichage de (CP) à la fin de ligne. Pendant que la commande s'exécute, vous avez la possibilité d'arrêter momentanément l'affichage par appui sur CTRL—S. Le redémarrage se faisant par CTRL—Q (cf caractères de contrôle).

A4.3 - SUPPRESSION DE CARACTERES :

La commande D permet de détruire n caractères vers l'avant ou vers l'arrière à partir de (CP). Vous pouvez tout aussi bien détruire les caractères de fin CR et LF pour ne reformer qu'une ligne.

La commande K (kill) permet la destruction de n lignes en avant ou en arrière de (CP). Si l'argument est nul, la ligne en cours sera détruite.

Si le (CP) se trouve en milieu de ligne, une destruction avant provoquera une destruction de caractères jusqu'à la fin de ligne. Le début de ligne sera alors concaténé avec la ligne suivante.

Vous pouvez aussi utiliser le signe # pour détruire tout le texte.

A4.4 - INSERTION :

La commande I permet l'insertion de caractères. Si vous entrez votre commande en minuscules, ED convertit automatiquement en majuscules.

Si vous utilisez la commande I seule, ED stocke les lignes de texte une par une et attend l'appui sur <Enter> pour continuer.

Pour sortir du mode d'insertion, appuyer sur CTRL-Z ou sur la touche ESC.

Dès que le prompt sera réapparu, vous ne vous trouverez plus en mode d'insertion.

Cde	Fonction
CTRL-H	Efface le dernier caractère entré et déplace le curseur d'un caractère vers la gauche
CTRL-U	Se positionne sur la ligne suivante avec le même numéro de ligne sans effacer à la vidéo la ligne en cours
CTRL-X	Efface la ligne en cours et reste sur la même ligne à la vidéo

Si vous entrez en mode insertion avec I majuscule, toutes les données entrées au clavier seront transformées en majuscules. Si vous entrez en mode insertion avec un i minuscule, les données resteront en minuscule.

La seconde méthode consiste à entrer la commande U avant de passer en insertion ce qui aura le même effet que d'entrer avec un I.

Pour inhiber cette conversion, entrez la commande -U.

Commande d'insertion de chaîne :

Cette forme d'insertion de chaîne n'insère les caractères entrés, qu'en mémoire et retourne automatiquement au prompt de l'éditeur.

Pour insérer une chaîne de caractères, positionnez tout d'abord le (CP) sur la zone concernée. Pour exemple si vous désirez insérer une chaîne sur la première ligne du texte, utilisez la commande B pour positionner le (CP).

Entrez ensuite :

i <texte> CTRL-Z Ex. : iMonsieur, CTRL-Z

Cette commande a pour effet d'insérer le texte "Monsieur," au début du texte.

A4.5 - REMPLACEMENT DE CARACTERES :

La commande S (substitute) cherche en mémoire une chaîne de caractères spécifiée dans la commande et la remplace par l'autre chaîne spécifiée elle aussi dans la commande.

s <chaîne à chercher> CTRL-Z <chaîne de remplacement>

Ex : sMonsieur, CTRL-ZBonjour : ce qui aura pour effet de remplacer Monsieur par Bonjour

A5 - COMBINAISON DE COMMANDES :

Vous avez la possibilité d'entrer sur la même ligne de commande de l'éditeur plusieurs commandes. La chaîne de commandes n'étant exécutée qu'après un appui sur `<Enter>`

- la combinaison de commandes ne peut excéder 128 caractères
- si une combinaison comprend une chaîne de caractères, la ligne de commandes ne pourra excéder 100 caractères
- les commandes de fin d'édition ne peuvent apparaître dans une combinaison,
- les commandes I, J, R, S et X doivent être les dernières d'une combinaison ou se terminer par un `CTRL-Z`

Déplacement de (CP) :

Exemple : le texte est :

1 : Monsieur,
2 : suite à notre...

1 : *T affiche la ligne en cours
1 : Monsieur
1 : * 1-7 ct déplace le (CP) de 7 caractères vers
la gauche et affiche la ligne.

sieur

1 : *

Affichage de texte :

Vous pouvez mixer vos commandes d'édition, afin de faire des suppressions de parties de texte, de déplacer le (CP) en conséquence et d'éditer la ligne pour correction.

Ex : *15C5D0LT déplacement de (CP) de 15 caractères, suppression de 5 caractères, déplacement de (CP) vers le début de ligne et affichage de la ligne obtenue.

Ex : *2L2KB#T déplacement de (CP) de deux lignes en avant, suppression de deux lignes, repositionnement de (CP) en début de texte et affichage de tout le texte.

Ex : 20Ci monsieur, CTRL-Z 0LT

déplacement de (CP) de 20 caractères vers l'avant
insertion de la chaîne "Monsieur," , déplacement
de (CP) vers le début de ligne et affichage de la
ligne.

A-6 COMMANDES AVANCEES

A6-1 DEPLACEMENT DE CP ET AFFICHAGE :

Commande P :

Bien que vous puissiez afficher à l'écran tout ou partie de texte avec la commande T, il est parfois intéressant de pouvoir disposer d'un mode de listing par page. C'est le but de la fonction P qui supporte l'attribut n indiquant le nombre de pages à lister.

Syntaxe : nP, -nP

Si le paramètre n n'est pas spécifié, l'éditeur listera les 23 lignes suivant le (CP) et l'incrémentera en conséquence.
Pour afficher la page courante entrez la commande : OP

Commande N :

Lorsque l'éditeur affiche des lignes de texte à l'écran, il accepte une commande n lui indiquant la ligne de destination de (CP).

Syntaxe : n : exemple :4:

La commande déplacera le (CP) au début de la ligne spécifiée par le paramètre n.

Une autre forme de commande est :n inverse de la fonction précédente. Cette forme de commande permet l'exécution à partir d'une certaine ligne.

Ex : *2:: 4 t déplacement de (CP) vers la ligne 2 et affichage de 4 lignes

A6.2 - RECHERCHE ET REMPLACEMENT DE CHAINE :

Commande F :

Cette commande permet de rechercher dans un texte une chaîne de caractères spécifiée dans la commande.

Syntaxe : nF <chaîne>

n est une occurrence de recherche et <chaîne> la chaîne à rechercher

Ex : *2Fmonsieur

Si n n'est pas spécifié, l'éditeur recherchera la prochaine occurrence de la chaîne.

Le (CP) se déplace alors vers le début de la ligne où l'occurrence de la chaîne a été trouvée.

Si vous entrez votre commande en majuscule, la recherche se fera sur une chaîne en majuscules et réciproquement pour les minuscules. Si la chaîne n'est pas trouvée ED vous retournera le message :

BREAK "#" AT

Commande N :

La commande N permet la même recherche que précédemment mais en poussant la recherche au delà du buffer de transit jusqu'au fichier source. La commande supporte les mêmes attributs que la précédente.

Commande J :

La commande J (juxtaposition) insère une chaîne de caractères après la chaîne trouvée et supprime les lignes comprises entre la fin de la ligne insérée et la ligne entrée dans la commande.

Syntaxe : nJ <chaîne à rechercher> CTRL-Z <chaîne à insérer>
CTRL-Z chaîne à déléter

n indique toujours l'occurrence de recherche de la chaîne.

Si votre commande dépasse la longueur de l'écran, appuyez sur CTRL-E pour causer un retour ligne théorique et continuer votre ligne de commande (attention à ne pas dépasser 100 caractères)

Commande M :

La commande M (macro) permet de sauvegarder un groupe de commandes pour une utilisation ultérieure.

Syntaxe : nM <chaîne de commande>

n est le nombre de fois que la chaîne de commande doit être exécutée.

Un argument négatif n'est pas valide pour cette commande.

Si n est nul, le signe # sera son remplaçant par défaut et l'éditeur exécutera la macro jusqu'à rencontre de la fin du buffer de transit.

Ex : 2m10t exécutera 2 fois la commande d'affichage de 10 lignes

La terminaison pour une ligne de macro doit être la dernière commande d'une ligne. Cette ligne se termine par un <Enter.>

La fin d'exécution d'une macro affiche le message :

BREAK "#" AT..

Pour arrêter l'exécution d'une macro appuyez sur CTRL-Y

Commande Z :

Utilisez cette commande pour faire exécuter une pause entre l'exécution des commandes par l'éditeur. Cette pause vous permet de faire le point sur la tâche qui vient de se terminer.

Syntaxe : nZ

n indique le nombre de secondes d'attente pour la pause

Ex : *2m10t20z affichage de 10 lignes répétées deux fois avec attente de 20 secondes entre chaque exécution

A6-3 DEPLACEMENT DE BLOCS DE TEXTE

Commande X :

Syntaxe : nX
 nX <fichier> CTRL-Z

n représente le nombre de lignes à transférer dans un fichier. Si le nom de fichier n'est pas indiqué, l'éditeur prendra par défaut le nom X\$\$\$\$\$\$\$.LIB. Ce fichier est détruit lorsque vous quittez le mode d'édition.

Si vous spécifiez le nom du fichier, il ne sera pas détruit. L'option prise par défaut est .LIB.

Commande R :

Syntaxe : R
 R <fichier>

Cette commande permet la lecture et le chargement dans le buffer de transit d'un fichier librairie. Si le nom du fichier n'est pas spécifié, le nom pris par défaut est X\$\$\$\$\$\$\$.LIB. Le chargement permet la concaténation avec le texte déjà entré en mémoire.

A4-4 SORTIE DE L'EDITEUR :

Commande H :

Permet la sauvegarde du texte conservé en mémoire dans le fichier temporaire mais sans quitter l'éditeur.

Lorsque la sauvegarde est effectuée, le pointeur (CP) se trouve au début d'une zone mémoire vide.

Commande O :

Permet la réinitialisation de l'éditeur sans sauvegarde du texte en mémoire et retour aux conditions initiales.

Pour confirmation, l'éditeur demande si vous désirez réellement quitter le mode.

Commande Q :

La sortie par la commande Q permet le retour au système d'exploitation après que l'éditeur ait clos le fichier source et détruit le fichier temporaire (aucune sauvegarde du texte en mémoire).

A7 - MESSAGES D'ERREUR DE L'EDITEUR

Les messages d'erreur sont caractérisés par l'apparition à l'écran d'un message du type : BREAK "x" AT c
x désigne un des symboles définis dans la table suivante et c une lettre de commande à l'endroit à laquelle l'erreur s'est produite.

Symbole	Explication
#	Erreur sur recherche .ED ne peut pas trouver la chaîne spécifiée dans la commande ou la ligne de commandes.
?c	Lettre de commande non reconnue par l'éditeur.
O	Pas de fichier .LIB .ED ne peut pas trouver le fichier portant l'extension .LIB appelé par la commande R.
>	Buffer plein .ED ne peut sauvegarder d'autres caractères dans le buffer de transit ou la chaîne spécifiée dans une des commandes F, N ou S est trop longue.
E	Interruption de commande. Un appui sur une touche clavier a interrompu une commande en cours d'exécution
F	Erreur disque. Erreur bien souvent suivie par le message disque plein ou directory saturé

Exemples :

BREAK ">" AT A

Si vous venez de charger en mémoire une partie de texte contenue dans un fichier et que ce texte ne tient pas en mémoire, le (CP) se retrouve bloqué en fond de mémoire et vous ne pouvez éditer. Faites une réinitialisation par O ou H ou utilisez la commande OW pour stocker la moitié du buffer dans le fichier temporaire.

BREAK "#" AT F

Indique que la chaîne spécifiée par la commande n'a pas été trouvée. Le (CP) se retrouve alors en fond de mémoire. (utilisez B)

BREAK "F" AT F
DISK FULL

Indique que votre disque est plein. Utilisez la commande OX pour détruire des fichiers inutiles et libérer de la place ou sauvegardez sur un autre drive.

Si votre fichier est protégé par un attribut de lecture seule, l'éditeur vous renverra un message en conséquence.

ANNEXE B

LINK

Syntaxe : LINK d: <fichier, <o> =>fichier <o> <fichier <o>>

Explications :

Le programme permet la combinaison des fichiers relogeables produits par MAC pour ne donner qu'un seul fichier portant l'extension .COM utilisable comme fichier commande.

Les fichiers relogeables peuvent contenir des références externes et publiques. Ils peuvent aussi référencer des fichiers librairie. A ce moment LINK recherche dans le fichier librairie les modules donnés et les inclue dans le fichier de sortie.

Les options de LINK doivent être comprises entre des parenthèses et s'il y en a plusieurs, elles doivent être séparées par des virgules.

A	Mémoire additionnelle. Permet le stockage des données sur disque dans un fichier temporaire si l'espace mémoire est insuffisant.
B	Linkage du BIOS sous système CP/M 3 banké
Dhhhh	Origine des données. Identifie le début de mémoire comme origine pour zone commune et de données
Gn	Démarrage. Positionne l'adresse de démarrage au label n.
Lhhhh	Chargement. Change l'adresse de chargement des fichiers par l'adresse hhhh. L'adresse par défaut est 100 hexa.
Mhhhh	Taille mémoire. Utilisé pour les modules MP/M.
NR	Inhibe la création d'un fichier de la table des symboles
OC	Sortie sur fichier de commande (option par défaut)
OP	Sortie sur fichier page relogeable. Utilisation prévue pour système MP/M
OR	Sortie sur fichier de traitement système (MP/M)

OS	Sortie sur fichier page relogeable. système (MP/M)
Phhhh	Début de programme. Changement de l'adresse de début de programme à l'adresse hhhh
Q	Listing des symboles avec ? pour premier symbole
S	Recherche le fichier précédent comme librairie
\$CD	Destination des messages X : sortie console Y : sortie imprimante Z : sortie nulle
\$ID	Définition du stockage du fichier intermédiaire .d indique le drive
\$LD	Définition du stockage du fichier librairie
\$OD	Destination du fichier objet
\$SD	Destination du fichier symbole

Exemple :

A> LINK b:MYFILE <NR>

LINK80 utilise comme entrée le fichier MYFILE.REL du drive B et produit le fichier MYFILE.COM sur le drive B. L'option indique que le linkage se fait sans table des symboles

A> LINK M1, M2, M3

A> LINK M=M1, M2 M3

Linkage des fichiers M1, M2, et M3 pour former le fichier M1.COM

Linkage des fichiers M1, M2 et M3 pour former le fichier M.COM

A> LINK MYFILE, FILE5 <S>

Utilisation du fichier FILE5 comme fichier librairie et création du fichier MYFILE.COM

ANNEXE C MAC

Syntaxe : MAC fichier <\$option>

Explications :

MAC est le macro-assembleur du CP/M 3. Son but est de lire les fichiers de type .ASM, de les assembler pour former trois fichiers de sortie avec pour type .HEX, .PRN et .SYM. Le premier contient le programme en code objet hexadécimal.

Le second permet un listing console ou imprimante.

Le dernier est une liste de symboles utilisés dans le programme organisé sous forme de table triée.

Avant d'invoquer MAC, vous devez créer un fichier source de type .ASM qui contiendra des mnémoniques en langage 8080.

Option	Fonction (options d'entrée/sortie)
A	Unité source pour le fichier .ASM
H	Unité de destination fichier .HEX
L	Unité source pour la librairie des macro.
P	Unité de destination pour le fichier .PRN
S	Unité de destination pour le fichier .SYM

Modi- ficateur	Fonction
+L	Liste les lignes d'entrée à partir de la librairie macro des fichiers .LIB
-L	Pas de listing (option par défaut)
+M	Liste des lignes de macro lorsqu'elles sont en cours d'assemblage
-M	Pas de listing des macro.
*M	Liste uniquement les codes hexadécimaux générés par les expansions des macro
+Q	Liste tous les symboles locaux de la table des symboles
-Q	Pas de listing
+S	Concaténation du fichier symbole au fichier d'impression
-S	Pas de concaténation
+1	Produit un listing de première passe pour mise au point des macros dans le fichier .PRN
-1	Pas de listing

Exemple :

A> MAC SAMPLE

Assemblage du fichier SAMPLE.ASM

A> MAC SAMPLE \$PB AA HB SX

Assemblage du fichier SAMPLE en fichier .PRN sur le drive B, .ASM sur le A et .HEX sur le B. Le fichier .SYM est affiché à l'écran.

ANNEXE D TYPES DE FICHIER

Le système CP/M identifie les fichiers par un numéro de drive, un nom de fichier, une option désignant le type de fichier et un mot de passe optionnel. L'option permettant de définir le type de fichier est codée sur trois caractères séparée du nom de fichier par un point.

TYPE	DESIGNATION
ASM	Fichier source d'assemblage
BAK	Sauvegarde d'un fichier source utilisé par l'éditeur
BAS	Fichier source basic
COM	Fichier programme exécutable
ERL	Fichier relogeable du PASCAL MT+
HEX	Fichier programme format hexadécimal
INT	Fichier programme en langage intermédiaire du basic
IRL	Fichier indexé produit par LIB.COM
LIB	Fichier utilisé par MAC et RMAC comme fichier librairie Ces fichiers sont aussi utilisés par l'éditeur ED.COM
OVL	Fichier programme overlay produit par LINK-80
PAS	Fichier source produit par le pascal
PLI	Fichier source du PL/I-80
PRL	Fichier relogeable (Page Relogeable File)
PRN	Fichier imprimable sur écran ou imprimante
REL	Fichier relogeable produit par MAC et pouvant être lié par LINK-80
SUB	Fichier utilisé par la commande SUBMIT
SYM	Fichier de table de symboles généré par MAC, RMAC, LINK
SYS	Fichier système CP/M 3
TEX	Fichier source pour traitement de texte DIGITAL-RESEARCH
TOK	Fichier en langage intermédiaire pour PASCAL
SRF	Cross-reference produit par la commande XREF
\$\$\$	Fichier temporaire