

**SYSTEME
D'EXPLOITATION
NEW DOS 80
2.0 G**



Copyright version originale © 1984

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, élaboré à partir de la documentation originale d'APPARAT, est une traduction soignée et a fait l'objet de longs tests de cohérence.

Néanmoins, il se peut que par défaut d'impression, quelques erreurs s'y soient glissées.

Soyez assurés que toute remarque sera la bienvenue et nous permettra de redistribuer à nos adhérents du CLUB GUEPARD, la ou les pages mises à jour pour remplacements ou additifs.

L'esprit de cette documentation est de vous apporter la possibilité d'aborder le NEWDOS 80 avec facilité.

Attention, cette version, considérablement modifiée, si elle reste parfaitement compatible avec l'originale possède des perfectionnements et des modules complémentaires.

Soyez donc patient et donnez-vous donc la peine de nous lire.

R.N.

SOMMAIRE

1 - INTRODUCTION AU NEWDOS 80 . 2 0 G

2 - BIBLIOTHEQUE DES COMMANDES DU DOS

3 - LES FACILITES DU NEWDOS 80 . 2 0 G

4 - LES MODULES DOS :
Structure et informations complémentaires.

5 - LES ROUTINES DU DOS

6 - LES UTILITAIRES DU DOS

7 - LES MYSTERES DU DIRECTORY

APPENDICES

A - GLOSSAIRE TECHNIQUE

B - MESSAGES D'ERREURS COMMENTES

1 - INTRODUCTION AU NEWDOS 80 . 2. 0 G.

PRESENTATION

Le **NEWDOS 80** Version 2 est un système d'exploitation PUISSANT mais très mal connu, sauf auprès des utilisateurs de TRS ou Vidéo-Génie.

Développé et mis au point par **APPARAT INC** colorado USA il a été modifié et "implémenté" par **HBN COMPUTEUR** pour être parfaitement adapté au **GUEPARD**.

Il est notre cheval de bataille tant ses performances sont extraordinaires, spécialement dans la gestion des périphériques, les paramètres du SYSTEME, la gestion des fichiers ...

Cette Version Française du **NEWDOS 80** porte la mention 2. 0 G, la spécialisant ainsi pour le **GUEPARD** et elle ne peut donc pas fonctionner sur un autre modèle, même supportant ce système d'exploitation. Mais tout en étant plus performante, elle permet d'accéder à toutes les disquettes enregistrées sur d'autres modèles compatibles, d'en extraire les logiciels, de les faire tourner sur le **GUEPARD**, d'en faire des copies etc... Il suffit pour cela d'utiliser le lecteur 1 avec les bons paramètres Disque (PDRIVE) et la commande DOS WRDIRP. Bien sûr, ces logiciels ne profiteront pas à fond des possibilités du **GUEPARD** (vidéo, clavier, etc...), mais ils restent parfaitement utilisables (même issus de disquettes TRS DOS).

Attention, l'inverse n'est pas forcément vrai, si vous utilisez dans vos logiciels des ordres ou compléments BASIC de la version 2. 0 G.

Quatre versions de la disquette SYSTEME sont disponibles :

A version lecteur 40 pistes double face en 80 colonnes.

B version lecteur 40 pistes double face en 64 colonnes.

C version lecteur 80 pistes double face en 80 colonnes.

D version lecteur 80 pistes double face en 64 colonnes.

Avec chaque appareil, en fonction de la version des lecteurs, sont fournies 2 disquettes, soit AB, soit CD.

La version 80 colonnes est la version standard utilisation.

La version 64 colonnes est fournie pour assurer la compatibilité vidéo avec les TRS mod I et III, et possède le module DEBUG.

Les disquettes "MAITRE" sont livrées en simple face. Un utilitaire est fourni pour faciliter la duplication dans la version de votre choix. Veillez à n'utiliser que des disquettes de bonnes qualités et d'en faire l'usage pour lequel elles sont prévues.

Etudiez attentivement les chapitres qui suivent, ils vous ouvrent les portes d'un DOS (Disk operating system : système d'exploitation) puissant et facile à dominer.

AVERTISSEMENTS

Le NEWDOS 80 Version 2 est cédé par APPARAT INC à HBN COMPUTEUR pour en faire la rediffusion, adapté sur son électronique GUEPARD.

HBN COMPUTEUR porte la responsabilité et la propriété des modifications effectuées sur le SYSTEME de base.

Toute utilisation ou reproduction de la Version Française sur d'autres appareils à des fins commerciales sont strictement interdites.

Le NEWDOS 80 . 2 0 G est fourni avec chaque appareil GUEPARD et HBN COMPUTEUR a fait le maximum pour que cette version vous donne entière satisfaction.

HBN COMPUTEUR vous remercie de toute critique constructive que vous pourriez lui faire parvenir pour l'améliorer encore.

HBN COMPUTEUR vous assure de son soutien technique pour corriger toute malfaçon qui aurait échappé à son contrôle, mais ne saurait être tenu pour responsable d'interruption de service, de pertes de profit, ou de dommages résultant de l'utilisation de ce système ou de logiciels utilisant ce système.

Une programmation soignée, une étude sérieuse de la documentation, de fréquentes copies de sauvegarde sont le gage d'une utilisation fructueuse.

Notre Adresse : HBN COMPUTEUR
 90, rue Charlier

 51100 REIMS
 FRANCE

2 - BIBLIOTHEQUE DES COMMANDES DU DOS

REPERTOIRE DES COMMANDES DU SYSTEME SOUS NEWDOS/80 :

2-1. APPEND	2-12. COPY	2-23. KILL	2-34. PURGE
2-2. ATTRIB	2-13. CREATE	2-24. LIB	2-35. R
2-3. AUTO	2-14. DATE	2-25. LIST	2-36. RENAME
2-4. BLINK	2-15. DIR	2-26. LOAD	2-37. ROUTE
2-5. BOOT	2-16. DO	2-27. MDDORT	2-38. STMT
2-6. BREAK	2-17. DUMP	2-28. MDCOPY	2-39. SYSTEM
2-7. CHAIN	2-18. ERROR	2-29. MDRET	2-40. TIME
2-8. CHNON	2-19. FORMAT	2-30. PAUSE	2-41. VERIFY
2-9. CLEAR	2-20. FREE	2-31. PDRIVE	2-42. WRDIRP
2-10. CLOCK	2-21. HIMEN	2-32. PRINT	
2-11. CLS	2-22. JKL	2-33. PROT	

CONVENTIONS DE NOTATIONS ET INFORMATIONS GENERALES.

Les commandes du DOS se terminent par un "ENTER": Dans les instructions et informations qui suivent, il ne sera plus indiqué : l'opérateur y pourvoira.

Les commandes du DOS sont limitées à un total de 80 caractères, y compris le "ENTER" final.

Les parenthèses sont utilisées pour démarquer un paramètre facultatif. Si l'on utilise ces paramètres facultatifs, les parenthèses sont bien sur inutiles.

ex. : (PROT = 8 x x), (ASE = yn), (ASC = yn).

doit être codé comme suit :

PROT = READ, ASE = N, ASC = Y (par exemple).

Les majuscules de A à Z et les caractères alphanumériques doivent être employés exactement comme ci-dessus.

Les minuscules et les mots avec ou sans chiffres décimaux représentent des sigles que l'utilisateur doit remplacer par des valeurs ou variables (voir exemple ci-dessus).

Dans certains cas où le sigle doit être impérativement remplacé par un caractère donné, le sigle contient, en minuscules, tous les caractères légaux pour cette variable.

Ceci permet de se souvenir des caractères légaux utilisables pour le remplacement de ce sigle. Par exemple, si ASC = Y et ASC = N sont les deux seules valeurs légales de ASC, alors le sigle sera ASC = yn.

Partout où la virgule est utilisée pour les commandes du DOS, elle peut être remplacée par un espace.

Les valeurs numériques sans suffixe H sont considérées comme des valeurs décimales et les valeurs hexadécimales doivent être nanties du suffixe H, à moins qu'il en soit spécifié autrement.

Ex. : 4000 H et 16384 représentent la même valeur.

Lors de la spécification d'un fichier disque, nous utiliserons le terme "carfich" : caractéristiques du fichier. Un "carfich" à la forme suivante:

nom 1 (/ext) (motpasse/) (: dn 1).

Les paramètres doivent être spécifiés dans l'ordre ci-dessus.

nom 1 est le nom du fichier composé de 1 à 8 caractères dans lequel le premier doit être une lettre majuscule et les autres des lettres ou des chiffres

ext est le nom de l'extension (par exemple CMD, BAS, OBJ, CIM TXT, DOC, COM, etc) qui classe le fichier. Un fichier ne nécessite pas obligatoirement une extension mais si elle est nécessaire elle devra comporter de 1 à 3 caractères dont le premier devra être une lettre majuscule et les autres une lettre ou un chiffre. Si un fichier a un nom d'extension, tous les "**carfich**" concernant le fichier doivent comprendre le nom de l'extension, à moins qu'un nom d'extension par défaut ait été prévu (par ex / CMD ou / JCL).

motpasse est un mot de 1 à 8 caractères dans lequel le premier doit être une majuscule et les autres des lettres ou des chiffres. Motpasse est la valeur donnée à la fois pour l'accès et la mise à jour de mots de passe pour un fichier lors de sa création. Motpasse est la valeur utilisée pour le contrôle du mot de passe lorsqu'un fichier existant est ouvert. Motpasse est exigé dans un "**carfich**" si les mots de passe sont validés et le fichier a des mots de passe attribués; sinon il est inutile.

dn 1 est le numéro de lecteur de la disquette contenant le fichier.

Ex :

FICHER 1 / BAS. MOTPASSE : 0

FICHER 2 : 3

FICHER 3 . MOTPASSE.

Le NEWDOS/80 accepte les minuscules pour toutes les commandes de la bibliothèque du DOS et pour toute autre entrée qui pourraient être demandée. Pour chaque commande de la bibliothèque du DOS le mot de code de la commande est spécifié avec une brève définition. Ensuite si la commande accepte des paramètres un synoptique de la commande est donné indiquant la liste de tous les paramètres exigés ou facultatifs. Viennent ensuite les explications de la commande des paramètres et options. En fin quelques exemples de commande du DOS sont donnés. Par contrainte de documentation, le synoptique de commande est quelquefois commenté sur plusieurs lignes de ce document; toutefois l'utilisateur devra se souvenir que les commandes sont juxtaposées. A moins qu'il en soit stipulé autrement, une commande de la bibliothèque du DOS est exécutable sous MINIDOS.

2.1 - APPEND Ajouter un fichier à la suite d'un autre.

APPEND, carfich 1, (TO) carfich 2.

Cette commande ajoutera le fichier carfich 1 à la suite du fichier carfich 2. L'EOF du FPDE du catalogue du fichier carfich 2 détermine le point auquel le fichier carfich 1 est ajouté. Ceci peut causer des ennuis si le fichier carfich 2 a des caractères explicites dans son EOF comme c'est le cas dans les fichiers BASIC ou ASSEMBLEUR. Le fichier carfich 1 n'est pas altéré. Le contenu initial du fichier carfich 2 n'est pas altéré ; le fichier est simplement ajouté.

APPEND n'est pas exécutable sous MINIDOS.

Exemple :

1) APPEND XXX : 1, YYY / DAT : 0.

Le contenu du fichier XXX dans le lecteur 1 est ajouté à la fin du fichier YYY / DAT qui se trouve dans le lecteur 0.

2) APPEND AAA TO BBB

Le contenu du fichier AAA est ajouté à la fin du fichier BBB. Le DOS recherche sur les disquettes présentes pour trouver les deux fichiers.

2.2 - ATTRIB apporte un complément d'attributs de fonctionnement à un fichier.

ATTRIB, carfich 1 (,INV) (,VIS) (,PROT = xxx)
(,ACC = motpasse 1) (,UPD = motpasse 2) (,ASE = e)
(,ASC = c) (,UDF = u),

Cette commande s'adresse au fichier "carfich 1". Au moins un des paramètres facultatifs doit être spécifié. Si des mots de passe sont attribués dans votre système alors le carfich 1 doit spécifier le mot de passe de mise à jour existant pour ce fichier s'il y a lieu.

INV rend le fichier invisible. A moins que l'option I soit spécifiée, le fichier ne sera pas listé par DIR.

VIS supprime la fonction invisible, que le fichier contienne cette fonction ou non.

PROT = xxx spécifie le niveau d'accès autorisé pendant les entrées/sorties du fichier si les mots de passe sont validés (voir option AA du système), et si un mot de passe d'accès, et non de mise à jour, a été utilisé pour ouvrir le fichier. Les niveaux sont définis pour les valeurs de xxx par :

LOCK : niveau 7 : pas d'accès permis au fichier excepté pour le chargeur du système.

EXEC : niveau 6 : accès permis seulement pour exécuter les fichiers comme programme. Le basic demandera soit RUN, soit LOAD avec l'option R et mettra hors circuit la touche BREAK évitant ainsi à l'utilisateur d'arrêter un RUN et interdisant l'exécution d'instructions directes.

READ : niveau 5 : accès permis pour exécuter ou pour lire le contenu d'un fichier.

WRITE : niveau 4 : accès permis pour l'exécution, lecture ou écriture d'un fichier.

RENAME ou NAME : niveau 2 : accès permis pour exécuter, lire, écrire ou renommer un fichier.

KILL : niveau 1 : accès permis pour exécuter, lire, écrire, renommer ou détruire un fichier.

FULL : niveau 0 toutes les opérations sont permises sur le fichier

ACC = motpasse 1. motpasse 1 sert de mot de passe d'accès au fichier. Si nul, des espaces lui sont affectés, sinon la valeur doit être de 1 à 8 caractères. Le premier étant une majuscule et les autres des lettres ou des chiffres. L'affectation d'un mot de passe d'accès par ce paramètre de ATTRIB est le seul moyen qui permettra l'utilisation de PROT=xxx pour la protection et ceci seulement si le mot de passe d'accès est différent du mot de passe de mise à jour. Si un mot de passe est spécifié lorsque le fichier est créé il est affecté à la fois au mot de passe de mise à jour et au mot de passe d'accès, le mot de passe de mise à jour est prioritaire à l'ouverture du fichier. Si des mots de passe sont validés et que le mot de passe affecté à carfich à l'ouverture n'est pas le mot de passe de mise à jour mais le mot de passe d'accès le niveau de protection en cours est stocké dans le FCB pour utilisation ultérieure des fonctions lire, écrire, charger, etc... du DOS. Si ultérieurement un accès est tenté violant le niveau d'accès, il apparaîtra la mention "ACCES IMPOSSIBLE (FICHIER PROTEGE)

UPD = motpasse 2. Motpasse 2 est affecté en tant que mot de passe de mise à jour pour le fichier. Il a la même configuration que le mot de passe d'accès. Lors de l'ouverture du fichier lorsque les mots de passe sont validés, le mot de passe spécifié sur le carfich est comparé au mot de passe de mise à jour du fichier. S'il y a concordance alors l'accès FULL est donné au fichier.

ASE = e où e est soit Y soit N. Ce paramètre a été rajouté afin de permettre au DOS d'allouer plus d'espace disquette à un fichier si ASE = Y ou interdire toute allocation supplémentaire si ASE = N. ASE = Y est la condition par défaut lorsqu'un fichier est créé.

ASC = c où c est soit N soit Y. Ce paramètre permet au DOS de désallouer un espace disquette si ASC = Y, et il est interdit si ASC = N. ASC = Y est la condition par défaut lorsqu'un fichier est créé.

UDF = u où u est, soit Y soit N. Ce paramètre a été rajouté pour marquer que le fichier a été mis à jour si UDF = Y et pour effacer cette marque si UDF = N.

Le système du DOS marque un fichier comme mis à jour, chaque fois qu'il est prêt à mettre à jour un secteur de ce fichier et qu'il trouve le catalogue de ce fichier non marqué comme mis à jour.

Exemple de commande ATTRIB :

1) ATTRIB, XXX / CMD : 1, UPD= ZXCVB, ACC = NMLKJ, PROT = EXEC.

Assigne au fichier XXX / CMD dans le lecteur 1 le mot de passe de mise à jour ZXCVB, le mot de passe d'accès NMLKJ et le niveau de protection 6 qui permet au programme d'être exécuté, mais interdit lecture et écriture. Puisque le carfich XXX / CMD = 1 ne spécifie pas un mot de passe, nous devons supposer que : soit la vérification du mot de passe a été invalidée (option SYSTEM AA = N), ou que le fichier n'avait pas un mot de passe en cours avant la commande ATTRIB.

2) ATTRIB, YYY / DAT . QZBV INV ASE = N ASC = N UDF = N

Cette commande vérifie si le fichier YYY / DAT a le mot de passe de mise à jour QZBV et si tel est le cas affecte au fichier la fonction invisible, signale qu'une allocation d'espace supplémentaire et que l'annulation des espaces excédent ne sont pas admises et termine en effaçant le drapeau 'mise à jour' du fichier.

- 2.3 - AUTO** permet d'exécuter une commande du DOS lors de l'initialisation.

AUTO (comdos)

Cette instruction permet à l'utilisateur de charger une commande du DOS de 1 à 31 caractères, qui sera appelée automatiquement au moment de la réinitialisation. Cette commande est stockée dans les derniers 32 octets du secteur GAT de la disquette système en cours.

Si comdos n'est pas spécifié, une commande nulle sera alors stockée dans le secteur GAT pour indiquer, à l'initialisation ou lors du branchement de l'appareil qu'aucune commande AUTO n'est prévue. - Si l'option SYSTEME à l'initialisation était $AB = N$ et $BC = Y$, alors l'appui sur la touche ENTER pendant la réinitialisation fera que la commande automatique se trouvant dans le secteur GAT sera ignorée, et le programme sautera à NEWDOS READY.

La fonction AUTO est particulièrement utile à l'opérateur qui exécute fréquemment la même commande ou chaîne de commandes au moment de l'initialisation (voir CHAIN, DO) et en mettant l'option $AB = Y$ ou $BC = N$, l'opérateur est forcé d'utiliser cette commande, ou chaîne de commandes, permettant ainsi aux personnels contrôlant l'activité des ordinateurs, de limiter l'activité des utilisateurs de l'appareil. (Mode exécution forcée).

Exemples de commandes AUTO :

1) AUTO BASIC RUN "XXX/BAS" : cette instruction active le BASIC, et fait démarrer l'exécution du programme BASIC XXX / BAS lors de chaque mise en route de l'appareil ou de chaque réinitialisation.

2) AUTO DO RSACTION : fait démarrer le chaînage à partir du fichier RSACTION / JCL, exécutant ainsi le DO et les autres commandes de programmes qui sont contenues dans ce fichier.

3) AUTO : fait que les réinitialisations et branchements de l'appareil qui suivront, entraîneront un READY normal, attendant que la commande du DOS lui soit transmise par le clavier.

- 2.4 - BLINK** Active/invalidé le clignotement du curseur.

BLINK, (yn)

BLINK ou BLINK,Y : le curseur clignote.

BLINK N : Le curseur est fixe.

L'option BH du système peut être utilisée pour définir la condition du curseur lors de la mise en route ou de la réinitialisation.

- 2.5 - BOOT** Effectue une réinitialisation du NEWDOS . 80

BOOT

Cette commande peut paraître stupide avec un RESET G plus rapide d'accès, mais son utilisation, bien que strictement équivalente, peut être dirigée par logiciel dans un fichier chaînage par exemple ou un CMD"BOOT" en BASIC pour réinitialiser des paramètres SYSTEME et un mode AUTO actif au RESET.

2.6 - BREAK Active ou invalide la touche BREAK.

BREAK (yn)

BREAK ou BREAK, Y : La touche BREAK est activée comme toute touche normale de saisie (code hexadécimal 01) jusqu'au NEWDOS READY suivant.

BREAK, N : La touche BREAK est désactivée jusqu'au retour au prochain NEWDOS READY (en fonction de l'option AG du système).

La commande BREAK est utile pour les programmes qui nécessitent une touche BREAK validée, validation que l'on peut obtenir par un DOS CALL (vecteur 4419 H). Le même principe s'applique aux programmes qui nécessitent une invalidation de cette touche BREAK. Emploi possible en CMD "BREAK (yn) ou sous CHAIN, DO.

NOTA : L'exécution de BREAK à partir de DOS READY est inutile si le retour immédiat à DOS READY réinitialise la touche BREAK suivant l'option AG du système.

2.7 - CHAIN Permet l'enchaînement de commande en substituant aux entrées clavier des données de fichiers Séquentiels sur disquette.

CHAIN, Carfich (, nom de section)

La commande DO lui est équivalente.

L'exécution de la commande CHAIN autorise, depuis un fichier sur disquette, le traitement d'une chaîne de caractères comme si elle avait été saisie au clavier.

Elle place le DOS en mode chaînage (s'il ne l'est pas déjà) :

Le fichier carfich (Séquentiel) est ouvert.

Si le nom de section n'est pas spécifié, l'exécution se fait depuis le début, dans le cas contraire, le nom de section est recherché et l'exécution débute à ce niveau.

L'exécution se poursuit alors comme s'il s'agissait d'entrée au clavier jusqu'à trouver

- La fin du fichier.
- La fin d'une section.
- Un arrêt temporaire par la commande CHNON, N ou CHNON, Y.

La commande CHAIN n'est pas acceptée sous MINIDOS.

Deux fichiers de chaînage ne peuvent être ouverts en même temps.

L'exécution d'une fonction CHAIN (ou DO) sous chaînage provoque la fermeture de l'ancien fichier à l'ouverture du nouveau.

Si l'extension n'est pas précisée dans le Carfich, le SYSTEME prend / JCL par défaut.

Exemple de commande CHAIN (ou DO)

1) CHAIN, XXX : Ø : Le chaînage s'exécute à partir du fichier XXX / JCL du drive Ø.

2) DO YYY / CHN : 1, QQQ : Le chaînage s'exécute à partir de la section nommée QQQ dans le fichier YYY / CHN du drive 1.

Un fichier de chaînage doit être créé avec soin pour ne pas créer de situation impossible au SYSTEME d'exploitation ou à l'utilisateur final. Voir le chapitre 3 - Section 2 pour plus de précision sur le CHAINAGE.

2.8 - CHNON modifie le déroulement d'un chaînage

CHNON, ynd

La commande CHNON s'utilise durant un chaînage. Dans le cas contraire, vous provoquerez l'erreur interne : FICHER NON OUVERT. Une commande CHNON ne doit pas être la dernière commande d'un fichier de chaînage ou d'une section de chaînage car cela serait sans signification.

CHNON, N suspend le chaînage en cours, conserve en mémoire la position courante dans le chaînage et redonne la main au clavier, au niveau d'APPELDOS en vigueur (ex. : si le chaînage est appelé du BASIC par un CMD «DO XXX», la main est rendue au basic, mode programme ou direct)

CHNON, Y fait reprendre le chaînage à la position suivant un CHNON, N. Si CHNON, Y est exécuté sous APPELDOS, elle stoppe l'exécution du chaînage qui ne reprendra qu'en mode direct (par exemple, sous basic, un programme en cours se terminera et le chaînage ne reprendra qu'au OK.)

CHNON, D fait reprendre le chaînage à la position suivant le CHNON, N. Mais si CHNON, D est exécuté sous APPELDOS, le chaînage reprend de suite et la main ne sera rendue au niveau d'APPELDOS que sur un arrêt du chaînage : CHNON, Y - CHNON, N - Fin d'une section - Fin du fichier de chaînage.

Voir le chapitre 3, section 2 pour plus de précision sur le CHAINAGE.

2.9 - CLEAR Remet à zéro la fonction ROUTE, les routines liées aux interruptions (comme CLOCK), les routines logiques, et l'espace mémoire après avoir remis à jour l' HIMEM.

CLEAR (,START = addr1) (, END = addr2) (, MEM = addr 3)

La commande CLEAR exécute les fonctions suivantes :

- 1 - La fonction du DOS : ROUTE, CLEAR
- 2 - Déconnecte les routines liées aux interruptions (la fonction affichage de l'heure par exemple).
- 3 - Déconnecte les routines X nom, comme le SPOOLER. Le SPOOLER doit donc être vidé avant un CLEAR au risque d'être perdu.
- 4- Place comme nouvel HIMEM la valeur addr3 ou à défaut la plus haute adresse mémoire
- 5- Remet à zéro le contenu de la mémoire compris entre addr1 et addr2
A défaut addr1 = 5200 H et addr2 = HIMEM

Dans tous les cas - addr2 doit être supérieur ou égal à 5200 H
- addr1 doit être inférieur ou égal à l'HIMEM,
donc à addr3

Exemples :

1) CLEAR, START = 6000 H, MEM = 0DFFFH. Toutes les routines sont déconnectées, l'HIMEM est fixé à 0DFFFH et l'espace mémoire entre 6000 H et 0DFFF est remis à zéro.

2) CLEAR Toutes les routines sont déconnectées, l'HIMEM est fixé au maximum et l'espace mémoire entre 5200 H et l'HIMEM est remis à zéro.

2.10 - CLOCK gère l'affichage permanent de l'heure

CLOCK (,yn)

CLOCK ou CLOCK, Y : autorise l'affichage de l'heure à la position 53-60 de l'écran sous la forme HH : mm : ss avec entretien chaque seconde

CLOCK, N : annule cette fonction d'affichage.

2.11 - CLS efface l'écran

CLS

CLS remplit l'écran d'espace de la couleur de fond d'écran en vigueur, réalisant ainsi son effacement.

Cet effacement n'affecte pas les réservations d'écran et s'opère dans la zone non réservée.

2.12 - COPY autorise la copie complète de disquettes, la copie de fichier, d'une série de fichier, avec ou sans intervention de l'opérateur.

Elle possède 6 formats :

1- COPY, Carfich1 (,TO), Carfich2 (,SPDN = dn3) (,DPDN = dn4)

2- COPY, \$ Carfich1 (,TO), Carfich2 (,SPDN = dn3) (,DPDN = dn4)

3- COPY, (:) dn1, Carfich1 (,TO), Carfich2 (,SPDN = dn3) (,DPDN = dn4)

* 4- COPY, (:) dn1, \$ Carfich2 (,TO), Carfich2 (,SPDN = dn3) (,DPDN = dn4)

5- COPY, (:) dn1, (= tc1) (,TO), (:) dn2 (= tc2), jj/mm/aa (,Y) (,N) (,NDMW) (,FMT) (,NFMT) (,SPDN = dn3) (,DPDN = dn4) (,SPW = motpasse1) (,NDPW = motpasse3) (,DDND) (ODN = nom1) (,KDN) (,KDD) (,NDN = nom2) (,SN = nom3) (,USD) (,BDU) (,UBB)

6- COPY, (:) dn1 (= tc1) (,TO), (:) dn2 (= tc2), jj/mm/aa, CBF (,Y) (,N) (,USR) (,/ext) (,UPD) (,ILF = Carfich3) (,XLF = Carfich4) (,CFWO) (,NDMW) (,FMT) (,NFMT) (,SPDN = dn3) (,DPDN = dn4) (,SPW = motpasse1) (,ODPW = motpasse2) (,NDPW = motpasse3) (,DDND) (,ODN = nom1) (,KDN) (,KDD) (,NDN = nom2) (,SN = nom3) (,USD) (,UBB) (,DDSL = ln1) (,DDGA = gc1)

La fonction COPY n'est pas exécutable sous MINIDOS (utilisez MDCOPY)

dn1 et dn2 sont des numéros de lecteur et peuvent être égaux.
Les : qui les précèdent sont facultatifs

Carfich1 représente les caractéristiques du fichier source

Carfich2 représente les caractéristiques du fichier destination

\$ précédent Carfich1 signifie que le fichier source (ou le fichier destination) sera dans le lecteur Ø durant la copie et se trouve sur une disquette ayant un système incompatible, ou pas de système du tout. Le module COPY en tiendra compte en se chargeant.
(avec deux lecteurs, cette utilisation devrait être très rare)

Durant l'exécution des FORMATS 2, 3, 4, 5 et 6 le SYSTEME peut vous demander différents mouvements de Disquette. Veillez à les exécuter correctement

SPDN = dn3 Paramètre disque de la disquette Source = dn3.
Ce paramètre permet l'exécution de la copie avec une disquette source, dont les paramètres disque sont différents des paramètres en vigueur sur le lecteur donné.

dn3 peut varier de 0 à 9 en accord avec le n° de lecteur équivalent de la commande PDRIVE (voir section 2.31)

DPDN = dn4 paramètre disque de la disquette destination = dn4
Ce paramètre permet l'exécution de la copie vers une disquette destination dont les paramètres disque sont différents des paramètres en vigueur sur le lecteur donné.

dn4 peut varier de 0 à 9 (voir SPDN)

SPDN et DPN peuvent concerner le même lecteur qui peut donc être concerné par 3 séries de paramètres différents : SYSTEM, source et destination.

Le format 1

concerne la copie d'un fichier unique

La ou les disquette (s) invoquée (s) doivent être placée (s) avant le lancement de l'ordre au DOS.

Le fichier source ne sera pas altéré

Le fichier destination, s'il existe déjà sera écrasé et remplacé.

Si les Carfich1 et 2 sont semblables ormis le numéro de lecteur, la commande peut être réduite.

EXEMPLE : COPY FICH1/DAT : 0, TO, FICH 1/DAT : 1
à COPY FICH1/DAT (:) 0 (, T0), (:) 1
donc à COPY FICH 1 : 0 : 1

Le format 2

est identique au format 1, mais le \$ indique qu'il y aura conflit, sur le lecteur 0, entre le SYSTEME et la disquette qui sera montée durant la copie. Le DOS vous donnera les ordres pour positionner vos disquettes durant la copie. Si la copie doit se faire entièrement sur le lecteur 0 uniquement, utilisez le format 4.

Le format 3

est identique au format 1, mais la copie ne peut s'effectuer que sur un seul lecteur et les disquettes source et destination sont différentes. Le DOS vous donnera les ordres pour positionner vos disquettes pendant la copie. Si le lecteur utilisé est le 0, et qu'une des deux disquettes source ou destination ne possède pas le SYSTEM compatible, utilisez le format 4.

Le format 4

est identique au format 3, mais permet la copie sur le seul lecteur 0, avec une des deux disquettes (source ou destination) ayant un SYSTEME incompatible ou pas de système du tout.

Le DOS vous donnera les ordres pour positionner vos disquettes pendant la copie

Le NEWDOS 80 ne possède pas de fonction BACKUP. Les formats 5 et 6 de COPY la remplacent avantageusement.

Le format 5 permet la copie complète d'une disquette, secteur par secteur

Le format 6 autorise la copie par fichier, autorisant ainsi de multiples possibilités de recopie rapide, au choix de l'opérateur.

Pour ces deux formats et pour un même lot de données, le format 5 est le plus rapide, mais le format 6 autorise des variations entre la disquette source et la disquette destination (paramètres disques, copie non intégrale...) et replace les fichiers en ordre serré sur le disque.

Le format 5

est le format de la copie complète, secteur par secteur d'une disquette. Les paramètres disques doivent être identiques pour la source et la destination (le nombre de piste excepté : TC). Par défaut ce sera ceux en vigueur. La disquette destination peut avoir plus de pistes que la source : le DIRECTORY de la disquette destination tiendra compte des pistes laissées libres si BDU est spécifié.

Les options du format 5 sont définies comme suit :

= **tc1** : nombre de pistes de la disquette source à recopier. Ce sera par défaut, celui des paramètres disques du lecteur

= **tc2** : nombre de pistes de la disquette destination. Ce sera par défaut, celui des paramètres disques du lecteur. tc2 doit être supérieur ou égal à tc1.

jj/mm/aa : sera la date de mise à jour (création) de la disquette destination. Ce sera, à défaut la date en vigueur sur le SYSTEM (horloge, temps réel). Si ce paramètre est le dernier utile dans la fonction COPY, il peut être omis. Sinon, son emplacement doit lui être réservé par deux virgules. Ex. : COPY Ø 1 , , FMT CBF

Y : le système effectuera la copie sans se soucier du contenu éventuel de la disquette destination.

Y exclue l'emploi de N, ODN, ODPW, DDND, KDN ou KDD et vice et versa.

Y est l'option par défaut d'emploi d'une de ces options.

N : avant d'effectuer la copie, le SYSTEM explore la disquette à la recherche de données lisibles. Si c'est le cas, la copie sera interrompue. option pratique pour éviter d'écraser des disquettes déjà employées. N exclue l'emploi de Y, ODN, ODPW, DDND, KDN ou KDD et vice et versa.

NDMW Le SYSTEM suppose que les disquettes sources et destination sont montées correctement et ne posera aucune question en cas d'erreurs qui compromettent la copie. La copie est stoppée pour retour au NEWDOS READY. Si ni NFMT, ni FMT ne sont spécifiés, FMT sera pris par défaut (formatage). NDMW est une option intéressante pour une copie sous APPEL-DOS (par ex. CMD"COPY) et que le programmeur ne désire pas d'intervention de l'opérateur. Son utilisation en mode direct, source d'erreur, n'est pas recommandée.

FMT Impose le formatage de la disquette destination. Son emploi exclu celui de NFMT et vice et versa. Par défaut de ces deux options, le SYSTEM posera la question :
FORMATAGE (O / N) ?
(sauf si NDMW est spécifié car il force FMT par défaut).

NFMT Interdit le formatage. Cela suppose que la disquette destination soit correctement formatée. Son emploi exclu FMT et vice et versa.

SPW = motpasse 1. Si les mots de passe sont validés (option AA = Y du SYSTEM) et si l'option AR du SYSTEM est à N, la fonction COPY compare motpasse 1 au mot de passe de la disquette SOURCE. S'il n'y a pas concordance, la copie est annulée.

NDPW = motpasse 3. Motpasse 3 doit être conforme, il sera affecté comme mot de passe de la disquette destination. Son emploi exclu celui de BDU et vice et versa.

DDND Le SYSTEM affiche le nom et la date de la disquette destination, permettant à l'opérateur de décider de la suite à donner à la copie. Son emploi exclu Y, N, NDMW et vice et versa.

ODN = nom 1. Si le nom de la disquette destination n'est pas égal à nom 1, le SYSTEM demandera à l'opérateur s'il désire poursuivre malgré tout. Son emploi exclu Y, N, NDMW et vice et versa.

KDN La disquette destination conservera son ancien nom au lieu de prendre, par défaut, celui de la disquette source. Son emploi exclu Y, N, BDU et NDN.

KDD La disquette destination conservera son ancienne date de mise à jour (ou création). Son emploi exclu Y, N, NDN et USD et vice et versa.

NDN = nom 2. La disquette destination prendra comme nouveau nom : nom 2. Nom 2 doit être conforme aux règles des noms de disquettes (8 caractères dont le premier en majuscule). Son emploi exclu BDU et KDN et vice et versa.

USD La disquette destination prendra comme date de mise à jour celle de la disquette source. Son emploi exclu KDD et BDU et vice et versa.

SN = nom 3 Si le nom de la disquette source n'est pas égal à nom 3, le SYSTEM demandera à l'opérateur s'il désire poursuivre malgré tout.

BDU Carte option interdit la mise à jour du BOOT/SYS et du PDRIVE. Elle est à utiliser lorsque la disquette source à :

- Un mauvais directory.
- Un directory non standard.
- Pas de directory du tout.

ou lorsqu'il faut recopier la disquette sans aucun changement.

UBB Carte option, qui force l'utilisation de tout l'espace mémoire pour la fonction COPY (jusqu'au HIMEM DOS) est inutile sur NEWDOS . 80 version 2. Il reste présent pour assurer la compatibilité avec les anciennes versions et donc d'éventuels anciens logiciels.

Le format 6

Permet, de par l'option CBF, la copie fichier par fichier (et non pas secteur par secteur comme en format 5).

Il permet l'emploi d'options nouvelles : USR, / EXT, UPD, ILF, XLF, DDSL, DDGA, et CFWO.

Toutes les options du format 5 restent valables, sauf BDU.

L'emploi de NFMT exclu celui de Y, N, KDN, KDD, NDN, BDU, USD, NPDW, DDSL et DDGA.

OPDW peut être employé, et les fichiers SYSTEM ne seront pas recopiés s'ils existent déjà sur la disquette destination.

CBF Est l'indicateur du format 6, imposant la copie fichier par fichier.

USR La copie ne touchera que les fichiers utilisateurs à l'exclusion des fichiers SYSTEM ou invisibles

/ EXT La copie ne touchera que les fichiers portant cette extension.

UPD La copie ne touchera que les fichiers ayant le flag de mise à jour. Ce flag est positionné par le DOS à l'écriture pour indiquer qu'au moins un secteur a été modifié ou ajouté dans le fichier depuis la dernière remise à zéro de ce flag (voir ATTRIB ou PROT). Ce flag n'est pas remis à zéro par la fonction COPY.

ILF = Carfich 3 Autorise uniquement la recopie des fichiers contenus dans le fichier séquentiel Carfich 3. Dans ce fichier, toute spécification débutant à un point virgule est considérée comme un commentaire, le fait qu'un fichier de Carfich 3 ne soit pas sur la disquette source ne provoquera pas d'erreur. Son emploi exclu XLF et vice et versa.

XLF = Carfich 4 Autorise la recopie de tous les fichiers sauf de ceux contenus dans le fichier séquentiel Carfich 4. Mêmes remarques que pour Carfich 3.

CFWO Impose au DOS de passer en revue avec l'opérateur l'ensemble des fichiers destinés à être recopié en posant la question :

CARFICH COPIE (O, N, R, A) ?

répondre : O pour oui à la recopie.
N pour non à la recopie.
R pour recommencer la revue en cas d'erreur.
A pour annuler la séquence de recopie.


Aucune copie ne sera effectuée avant la fin de la revue.

ODPW = motpasse 2 Si l'option NFMT est spécifiée dans la séquence, les mots de passe validés (option SYSTEM AA = Y) et si l'option SYSTEM AR = N, cette option compare le mot de passe de la disquette destination avec motpasse 2. S'il n'y a pas concordance la copie est annulée.

DDSL = In1 In1 indique le LUMP de départ du directory sur la disquette destination, à défaut ce sera celui des paramètres disques en vigueur (PDRIVE). Son emploi exclu NFMT et vice et versa.

DDGA = gc1 gc1 indique le nombre de GRANULES allouées au directory. Par défaut, ce sera celui des paramètres disques en vigueur (PDRIVE). Son emploi exclu NFMT et vice et versa.

Ces deux dernières options permettent de créer une nouvelle disquette en déplaçant et en augmentant la capacité de son DIRECTORY options intéressantes lorsqu'il reste des GRANULES libres sur la disquette et aucun emplacement pour les fichiers dans le DIRECTORY (Ø Fdes et xx GRANS).

Durant une copie (ou un formatage) sans option NDMW, un appui sur la flèche droite  exécute une pause et poursuit l'exécution avec ENTER.

Un appui sur la flèche haut  stoppe l'exécution et annule la fonction en cours.

Tout arrêt de séquence, quelques raisons que ce soient, laisse la disquette destination dans un état inconnu.

Exemples de copies :

1) COPY XXX : 1 YYY : 1 Format 1. Le fichier XXX est recopié en fichier YYY dans le lecteur 1 sur la même disquette qui doit être déjà en place.

2) COPY AAA, BBB : 2 Format 1. Le fichier AAA est cherché sur toutes les disquettes en place. S'il est trouvé, il sera recopié en BBB sur le lecteur 2.

3) COPY AAA : Ø : 3 Format 1. Le fichier AAA de la disquette en place dans le lecteur Ø est recopié sur la disquette en place Lecteur 3.

4) COPY XXX : 1 2 SPDN = 9 Format 1. Le fichier XXX sur la disquette en place dans le lecteur 1 est recopié sur la disquette du lecteur 2 dont les paramètres disques doivent correspondre à la ligne 9 du PDRIVE Ø (voir section 2-31).

5) COPY \$ XXX : 1, YYY : Ø Format 2. La disquette destination n'est pas la même que celle en place dans le lecteur Ø au lancement de la copie. Le DOS donnera les ordres nécessaires durant l'exécution pour les changements.

6) COPY \$ XXX : Ø YYY : 1 Format 2. Idem exemple 5 pour la disquette source cette fois-ci.

7) COPY 1 XXX YYY / DAT Format 3. La copie se fera dans le lecteur 1. Le DOS donnera les ordres nécessaires durant l'exécution pour la mise en place des disquettes sources et destination.

8) COPY Ø XXX / DAT / DAT Format 3. La copie se fera dans le lecteur Ø. Les disquettes source et destination doivent avoir le même SYSTEME que la disquette qui sert au lancement de la copie (sinon voir 9) le DOS donnera les ordres nécessaires durant l'exécution pour la mise en place des disquettes source et destination.

9) COPY Ø \$ XXX / DAT / DAT Format 4. Idem exemple 8 mis à part le fait qu'une des deux disquettes (ou les deux) source ou destination n'ont pas le même SYSTEME que la disquette initiale.

10) COPY, Ø 1, 06/01/84, FMT Format 5. Exemple d'un des plus simple mode pour la récopie complète d'une disquette du lecteur Ø vers le lecteur 1, précédée d'un formatage de la disquette destination, la date de création de celle-ci sera 06/01/84. Son nom et son mot de passe sera celui de la disquette source.

11) COPY, Ø 1, , NFMT Format 5. Idem exemple 10 mis à part la date qui est celle de l'horloge temps réel et l'absence de formatage qui suppose que la disquette destination est déjà au bon format.

12) COPY, Ø, Ø, Ø6/Ø1/84, NFMT, USD, KDN, ODN = Bonjour, SN = Au revoir Format 5. La copie se fera dans le lecteur Ø, sans formatage. La disquette destination conservera son nom et prendra la date de mise à jour de la disquette source. Le DOS s'inquiètera de la marche à suivre auprès de l'opérateur si les noms des disquettes ne correspondent à ceux annoncés et donnera les ordres nécessaires à la mise en place des disquettes.

Ce format de copie est inexécutable sous APPELDOS (BASIC par ex) à cause du trop grand nombre d'échange disquette.

13) COPY Ø 1, , NFMT, CBF Format 6. Copie fichier par fichier, du lecteur Ø vers le lecteur 1, sans formatage (donc sans altération de la disquette destination).

14) COPY Ø 1, , NFMT CBF USR Format 6. Idem exemple 13, mais la copie ne concerne pas les fichiers SYSTEME ou invisibles.

15) COPY Ø 1, , NFMT CBF USR CFWO Format 6. Idem exemple 14, mais avec une revue des fichiers pour soumettre leur copie à la décision de l'opérateur.

2.13 - CREATE pour un fichier.

Pré-allocation d'espace disquette

CREATE, carfich 1 (, LRL = ln) (,REC = nb) (, ASE = yn)
(, ASC = yn).

Cette commande permet la création d'un fichier en précisant, si nécessaire, sa longueur d'enregistrement, son nombre d'enregistrements, évitant ainsi la dispersion future du fichier sur le disque (réduction du temps d'accès) et évitant la perte du fichier par oubli du CLOSE (voir manuel SBASIC).

LRL = ln Précise la longueur réservée à
chaque enregistrement, ln doit être compris entre 1 et 256 par défaut
ln = 256

2.15 - DIR

Affichage du contenu de la disquette.

Le DIRECTORY (ou catalogue) est le sommaire et le directeur de la disquette. Toute recherche, allocation, suppression de fichiers et gestion de l'espace disque passent par lui.

DIR (dn1) (,A) (,S) (,I) (,U) (,/ ext) (,P)

Sur cette commande, le DOS effectue un CLS et affiche en première ligne le lecteur considéré (par défaut dn1 = 0), le nom de la disquette, sa date de mise à jour, le nombre de place libre pour gérer les fichiers (FDES) et l'espace libre sur la disquette (en GRANULES)

1 GRANS = 1,25 kilo octets. *Soit 280 grans pour 350 K*

Le DOS affiche ensuite les informations sur les fichiers, en fonction des paramètres facultatifs.

- A En son absence, les noms des fichiers et leur extension apparaissent 4 sur chaque ligne. Dans le cas contraire, une ligne est attribuée à chaque fichier pour afficher dans l'ordre :

Le nom
L'extension
Le repère de fin de fichier (EOF)
La longueur logique d'enregistrement
Le nombre d'enregistrements logiques.
Le nombre de granules alloués
Le nombre d'emplacements du DIRECTORY occupés (EXT)

12 indicateurs (FLAG) définis comme suit :

- S Indique un fichier système
- I Indique un fichier invisible.
- U Indicateur de modification (remis à zéro par PROT).
- E Indicateur du paramètre ASE = N (voir CREATE).
- C Indicateur du paramètre ASC = N
- } Libres pour définition ultérieure.
- }
- }
- U Indicateur d'un mot de passe de mise à jour non nul.
- A Indicateur d'un mot de passe d'accès non nul.
- L Indicateur de niveau de protection (voir ATTRIB).

- S Autorise l'affichage des fichiers SYSTEME (/SYS) son absence l'interdit.
- I Autorise l'affichage des fichiers invisibles.
- U N'autorise l'affichage que des fichiers modifiés depuis le dernier PROT (ou ATTRIB) : flag U présent
- /ext N'autorise l'affichage que des fichiers portant cette extension.
- P Déroute l'affichage vers l'imprimante.

Lorsque l'écran ne permet pas l'affichage complet, le DOS interrompt l'affichage et affiche un ? en bas à gauche dans l'attente de :

ENTER pour poursuivre
BREAK pour annuler la fonction DIR

Exemple : DIR 1 / CMD

DIR 1 / CMD : Affiche à l'écran les noms et extension des fichiers (4 par ligne) dont l'extension est / CMD.

2.16 - DO Carte commande est équivalente à CHAIN (section 2-7)

2.17 - DUMP Transfert de bloc mémoire sur fichier disque.

DUMP, Carfich 1, adddébut, addfin (,addentrée) (,addemp)

où addxxx est une valeur d'adresse inférieure à 65536 ou 10000 H

La commande DUMP transfère sur un fichier disque, dont les caractéristiques sont celles de Carfich 1, le contenu de la mémoire compris entre l'adresse adddébut et l'adresse addfin.

Cette commande opère en 2 modes, qui dépendent de la valeur de addentrée.

Si addentrée = 0FFFF H (65535), le bloc est transféré tel quel et pourra être lu et traité par SUPERZAP (option DMDB) pour déverminage ultérieur.

Si addentrée < 0FFFF H, le bloc est transféré comme un programme en langage machine pour être ensuite traité par le chargeur du NEWDOS 80 (pour chargement par LOAD, exécution si Carfich possède / CMD ...)

Si addentrée n'est pas spécifiée, elle prend la valeur 402D H, pointeur de retour au NEWDOS READY.

addemp permet, au chargement suivant en mémoire, de déplacer l'adresse de début à la nouvelle adresse, déplaçant ainsi l'ensemble du bloc en RAM

Exemples :

DUMP, PROG / CMD : 1, 5200 H, 9ABC H, 54ED H transfère sur fichier PROG / CMD dans le lecteur 1 le contenu de la mémoire compris entre 5200 H et 9ABCH dans le format compatible avec le chargeur du DOS (avec les caractères de contrôle du chargeur). Ce programme sera ensuite exécutable par la commande PROG et débutera à l'adresse 54ED H.

DUMP DUMP/MEM:1 6000H,7000H

DUMP, TROUBLE / MEM, 0, 65535, 65535 transfère sur fichier TROUBLE / MEM dans le lecteur 0 le contenu de la mémoire de 0 à 65535 sans altération (sans les caractères chargeur) permettant ainsi ultérieurement un déverminage au SUPERZAP (attention la zone 4000 H - 5200 H NOYAU DOS est altérée par la commande DUMP et ne peut donc être considérée comme le reflet de la mémoire au moment de la faille éventuelle).

2.18 - ERROR Affiche en clair les messages d'erreur du DOS

ERROR, xx

Affiche l'erreur numéro xx (xx entier entre 0 et 63).

Formate une disquette

MN 2 - 18

FORMAT X^1, \dots, Y , PFST = 22, PFTC
= 2

Si un ennui quelconque a détérioré un ou plusieurs secteurs sur les pistes 22 et 23, SUPERZAP et sa fonction CDS permet de sauvegarder les secteurs en bonne santé. Ce type de format permet de ne formater que 2 pistes (22 et 23). On peut ensuite y ramener les secteurs sauvegardés et réaménager les autres.

- 2.20 - FREE** Affiche la première ligne de la fonction DIR pour chaque disquette montée.

FREE (, P)

Affiche à l'écran le numéro du lecteur, le nom de la disquette, sa date de mise à jour, le nombre de pistes, le nombre d'emplacements libres pour les fichiers, et l'espace libre en granules, ceci pour chaque disquette montée à la commande.

L'option P permet de dérouter les informations à l'imprimante.

- 2.21 - HIMEM** Fixe la plus haute adresse mémoire utilisable par le SYSTEME

HIMEM (, addr)

Le NEWDOS mémorise en 4049 H sur 2 octets la valeur de l'adresse mémoire la plus haute utilisable par ses commandes (telle que COPY, BASIC, EDASS, DISASS, MOVECASS).

Attention le chargeur de programme ne tient pas compte de cette limite

La commande seule affiche à l'écran la valeur en cours

La commande suivie de **addr** en décimal ou hexadécimal (H) rectifie le HIMEM avec cette nouvelle valeur

Exemple : HIMEM, 49000
Fixe le nouvel HIMEM à 49000

- 2.22 - JKL** Assure la recopie d'écran sur imprimante

JKL (ou un appui simultané sur les 3 touches)

Cette commande assure la recopie intégrale de l'écran sur l'imprimante. L'option SYSTEME AX contrôle la transmission des codes graphiques.

BREAK stoppe l'impression en cours

JKL peut s'utiliser sous APPELDOS (par ex. CMD " JKL " sous BASIC)

2.23 - KILL

pour détruire un fichier

KILL , Carfich1

Cette commande efface le fichier du directory de la disquette et rend la place qu'il occupait, à nouveau disponible.

Si le numéro du lecteur n'est pas spécifié, le fichier est recherché sur tous les lecteurs à compter du 0

Il est toutefois possible de récupérer le fichier tant qu'il n'y a pas eu écrasement de ses secteurs sur la disquette.

La manipulation est hasardeuse, mais abordable par un professionnel. (voir MANUEL TECHNIQUE)

Exemple : KILL XXX : 1 le fichier XXX du lecteur 1 est effacé

2.24 - LIB

Affiche sur l'écran l'ensemble

des commandes du NEWDOS 80

LIB

Cette commande s'exécute sans paramètre. Elle rappelle l'ensemble des commandes du DOS. Pour mémoire, mis à part les commandes COPY, FORMAT, APPEND et CHAIN, elles sont toutes exécutables sous MINIDOS.

2.25 - LIST

Liste un fichier texte à l'écran

LIST , Carfich1 (, lindep) (, linnb)

Cette commande affiche le contenu d'un fichier à l'écran

Carfich1 peut ne pas être un fichier texte, mais dans ce cas le résultat ne sera pas compréhensible

Voici des exemples de fichiers texte : programme BASIC sauvegardé avec l'option A, fichiers séquentiels en BASIC, source d'assembleur, sources FORTRAN ou COBOL, fichier de traitement de texte...

Pour lister des fichiers non texte, utilisez SUPERZAP

Lindep permet de débiter le listing à ce n° de ligne. Une ligne est délimitée par un EOL (0DH) (équivalent de ENTER)

Linnb permet de limiter le nombre de lignes à lister.

Si linnb est spécifié, lindep doit l'être aussi

Durant le listing, un appui sur Flèche droite  effectue une pause dans le listing au prochain EOL et reprendra sur un ENTER. Un appui sur Flèche haut  arrête la fonction.

Exemple : LIST XXX : 1 , 1 , 6
Cette commande liste à l'écran les 6 premières lignes du fichier XXX du lecteur 1.

2.26 - LOAD
fichier en langage machine

Chargement en RAM d'un

LOAD , Carfich1

Cette commande charge en RAM le fichier Carfich 1 et place son point d'entrée (adresse) sur 2 octets en 4403 H (17411). Le fichier doit être dans le bon format (créé par EDASS ou la commande DUMP)

Si le fichier est prévu pour se charger entre 4000 H et 5200 H, noyau DOS, le fonctionnement ultérieur n'est plus garanti.

Cette commande est utilisée pour stocker en RAM un programme ou des données pour une utilisation ultérieure (par exemple sous USR en BASIC). Souvenez-vous, son point d'entrée est stocké en 4403 H

Exemple : LOAD , PROGUSE/CMD
Charge le fichier PROGUSE/CMD. le BASIC peut utiliser ce programme par un DEFUSRX = PEEK (17411) + 256 * Peek (17412)

2.27 - MDBORT Cette commande permet de quitter le MINIDOS pour retour au NEWDOS READY

2.28 - MDCOPY Copie de fichiers sous MINIDOS

MDCOPY , Carfich1 (, TO) Carfich2

La fonction COPY n'est pas exécutable sous MINIDOS. Elle écraserait la zone supérieure à 5200 H.

MDCOPY permet la copie, lente il est vrai, de fichier sans altérer les zones hors du noyau DOS.

2.29 - MDRET Cette commande permet de quitter le MINIDOS pour retourner au niveau d'appel. Le buffer clavier est retrouvé intact, comme laissé à l'appui de «DFG»
Un ENTER est quelquefois requis, dû à la gestion des interruptions.

Si le déroulement d'un programme BASIC peut être interrompu par BREAK, un «DFG» permet l'accès au MINIDOS. Un MDRET, ENTER reprendra l'exécution du programme.

2.30 - PAUSE Affiche un message et attend l'appui sur ENTER

PAUSE , message

En mode direct, la commande ne réaffiche pas le message.

En APPELDOS, la commande affiche le message.

Dans tous les cas, la commande affiche :
APPUYER SUR «ENTER» POUR CONTINUER
et attend l'appui sur ENTER pour poursuivre

Exemple : (sous BASIC) CMD «PAUSE, PLACEZ LA DISQUETTE TOTO DRIVE 1». Le message «PLACEZ LA DISQUETTE TOTO DRIVE 1» est affiché suivi du message «APPUYEZ SUR ENTER POUR CONTINUER» et le DOS attend l'appui sur ENTER pour poursuivre le programme (il ne fait aucune vérification sur l'exécution de l'ordre donné)

2.31 PDRIVE paramètres disque

Commande de gestion des

PDRIVE (, Motpasse 1 :) dn1, (dn2 (= dn3)) (, TI = Type OP)
(, TD = type 2) (, TC = tc) (, SPT = spt) (, TSR = r) (, GPL = g)
(, DDSL = l) (, DDGA = gd) (, A)

Le NEWDOS 80 est capable de gérer un mélange d'un maximum de 4 lecteurs de 5 pouces ou 8 pouces ayant des caractéristiques différentes, et sur ces lecteurs, d'exploiter des disquettes ayant les caractéristiques les plus diverses, à condition que le lecteur concerné en soit lui-même capable

Il existe en série sur le GUEPARD 2 types de lecteur possibles :

- les 40 pistes double face (2 têtes de lecture)
- les 80 pistes double face (2 têtes de lecture)

et le contrôleur de disque est capable de gérer la simple ou double densité. La configuration 80 pistes DF permet donc la lecture de toutes les sortes de disquettes (qui peut le plus, peut le moins).

A chaque commande, PDRIVE correcte, le DOS affiche 10 lignes de caractéristiques disque numérotées de 0 à 9, alors que le SYSTEM ne supporte que le nombre de lecteurs exprimés dans l'option SYSTEM AL et ne saurait en tout cas excéder 4. (les lignes concernées par AL portent d'ailleurs un * pour confirmation des lecteurs en service). Les lignes supplémentaires permettent de mémoriser des caractéristiques disque pour de futures utilisations (comme dans COPY, FORMAT ou PDRIVE x = y)

A l'initialisation ou sur l'option A de PDRIVE 0, le DOS met à jour en RAM la table des paramètres disques pour les lecteurs en service, à condition que cette table ne présente pas d'erreur.

Si les mots de passe sont validés, Motpasse1 doit être spécifié et être celui de la disquette du lecteur dn1 ; dans le cas contraire, la commande sera invalidée

dn1 est le n° du lecteur dont on veut lister/modifier les paramètres

dn2 est le n° de ligne concerné par une modification

dn3 est le n° de ligne qui remplacera la ligne dn2

C'est le moyen le plus simple de modifier une ligne de paramètres avec une ligne existante préparée à l'avance et dont l'usage serait répétitif.

TI = Type op indicateur de type d'interface avec options. Le GUEPARD supporte les indicateurs A et C
Son TI normal est CK

Les options possibles sont les suivantes :

H indicateur pour lecture des lecteurs 8 pouces

I indicateur de numérotation secteur à partir de 1 (0 en mode normal)

J indicateur de numérotation piste à partir de 1 (0 en mode normal)

K indicateur de réservation de piste \emptyset par le SYSTEME. L'indicateur K force le Flag J. La piste \emptyset n'est plus accessible en mode normal. Cet indicateur impose de placer en TC, le nombre de pistes physiques moins une (79 ou 39 sur GUEPARD)

L indicateur qui permet de lire des disquettes 40 pistes sur des lecteurs prévus pour 80. Il force le système à effectuer 2 pas pour un pas de piste. L'écriture avec ce paramètre est réalisable mais déconseillée.

M indicateur de format disquette TRS DOS mod II'

TD = type 2 Indicateur de type de lecteur ou du type de formatage de la disquette dans le lecteur dn 2

type 2 peut être égal à :

- A = 5 pouces, simple densité, simple face.
- B = 8 pouces, simple densité, simple face.
- C = 5 pouces, simple densité, double face.
- D = 8 pouces, simple densité, double face.
- E = 5 pouces, double densité, simple face.
- F = 8 pouces, double densité, simple face.
- G = 5 pouces, double densité, double face.
- H = 8 pouces, double densité, double face.

TC = tc Indicateur du nombre de pistes sur le disque (moins la piste \emptyset en cas d'indicateur K dans TI)

SPT = spt Indicateur du nombre de secteur par piste. Le spt double d'une simple face sur une double face, les secteurs gérés par le NEWDOS sont de 256 octets. Suivant chaque TD, le spt maximum est de 10 pour A, 17 pour B, 20 pour C, 34 pour D, 18 pour E, 26 pour F, 36 pour G et 52 pour H.

TSR = r Indicateur de performance du lecteur pour effectuer un saut piste à piste. r = \emptyset pour un temps inférieur à 5 ms, 1 pour 10 ms, 2 pour 20 ms, 3 pour 40 ms. Le TSR normal du GUEPARD est \emptyset ce qui assure les temps d'accès disque les plus rapides

GPL = g Indicateur du nombre de granules par lump où g est compris entre 2 et 8.

Le lump représente l'unité de gestion de l'allocation d'espace disque dans le premier secteur du DIRECTORY : Table d'allocation des granules (GAT). Il y a 192 lump dans la GAT gérés par un octet dont chaque bit gère une granule (\emptyset = libre, 1 = occupé).

La granule est le minimum d'espace disque attribuable à un fichier. Sa taille est de 5 secteurs (1 GRANS = 5×256 octets = 1,25 Koctets).

Plus il y a d'espaces disque à gérer, plus il faut de granules par lump.

g = 2 assure la compatibilité avec le TRS MOD I

g = 6 assure la compatibilité avec le TRS MOD III

g = 8 permet la gestion de $192 \times 8 \times 5 = 7680$ secteurs.
(une disquette 80 pistes double face, double densité en contient 2844)

DDSL = l Indicateur de positionnement du premier lump du DIRECTORY. Ce paramètre, stocké dans le boot (3ème octets) permet la recherche et l'accès au directory d'une disquette. Pour déterminer le secteur relatif sur le disque, il suffit de multiplier l par g et par 5. l = 17 assure la compatibilité par les disquettes standard.

A Ce paramètre permet la mise à jour immédiate de la table en RAM pour les lecteurs valides par un "*" (option SYSTEM AL) et si la table complète est sans erreur.

PDRIVE \emptyset 1=7 TC=40 A
Idem exemple précédent en changeant TC=40.

PW = motpasse 2 Affecte un nouveau mot de passe à la disquette.

LOCK Affecte le mot de passe de la disquette, le nouveau éventuellement, comme mot de passe de mise à jour et mot de passe d'accès à tous les fichiers de la disquette.

UNLOCK Remplace tous les mots de passe fichiers par des blancs (deprotège tous les fichiers).

Exemples :

PROT, 2, RUF : tous les **FLAG U** de la disquette du lecteur 2 sont remis à zéro.

PROT, COUCOU : 1, NAME = AAA, PW = ADELE : Les mots de passe sont validés et **COUCOU** doit être celui de la disquette lecteur 1. Si c'est le cas, elle prend le nom **AAA** et son mot de passe devient **ADELE**.

2.34 - **PURGE** Destruction sélective des fichiers d'une disquette.

PURGE, (motpasse 1 :) dn 1 (, / ext) (, USR)

La disquette montée sur le lecteur **dn1** est visée par cette commande. Si les mots de passe sont validés, motpasse 1 doit être celui de cette disquette.

Pour chaque fichier (sauf **BOOT/SYS** et **DIR/SYS**) le DOS demandera à l'opérateur s'il faut le détruire (**O/N/A**) ou quitter la fonction purge: **O** pour oui, **N** pour non, **A** pour arrêt.

/ ext Cette option présente, la revue ne concernera que les fichiers portant l'extension **/ ext**.

USR Cette option présente, la revue ne concernera que les fichiers **NON SYSTEME** ou non invisibles.

Exemples :

PURGE, Ø Pour chaque fichier de la disquette présente lecteur **Ø** une revue sera faite pour détruire éventuellement ou stopper la fonction.

PURGE, 1, / CMD Idem exemple précédent, mais lecteur 1 et pour les fichiers **CMD** uniquement.

2.35 - **R** Répétition de la dernière commande DOS.

Cette commande provoque la réexécution de la dernière commande DOS (sauf **R** elle même).

Cette commande n'est pas exécutable sous **BASIC** via **CMD** "cde DOS".

Cette commande est nulle si l'option **SYSTEM BE = N**

Exemple : **DIR 1**
R
R
s'exécutera comme
DIR 1
DIR 1
DIR 1

2.36 RENAME

Change le nom et/ou l'extension d'un fichier

RENAME Carfich1, (, T0) Carfich2

Cette commande recherche le fichier Carfich1. Si le n° de lecteur n'est pas spécifié, la recherche se fera sur chaque disquette montée, à partir du lecteur 0. Le premier fichier trouvé prendra le nom et/ou l'extension carfich2.

Exemple : RENAME XXX / DAT : 1 YYY / OBJ
Le fichier XXX / DAT sur le lecteur 1 prendra le nom et l'extension YYY/OBJ

2.37 - ROUTE

Assure l'aiguillage des périphériques

ROUTE , périph 1 (, périph 2) (, périph 3)

ROUTE , CLEAR

Le but de cette commande est de faciliter la gestion des périphériques (écran, imprimante, série, clavier, divers...) en permettant les aiguillages multiples au travers des routines spécialisées (DRIVER) sans intervention sur le logiciel de fonctionnement.

Par exemple : driver l'affichage vidéo vers l'imprimante tout en conservant ou non l'affichage vidéo.

remplacer les entrées claviers par l'entrée série.

périph 1 est le périphérique (entrée ou sortie) à aiguiller

périph 2, périph 3, périph 4 ... sont les périphériques vers qui aiguiller périph 1.

Attention sur une même commande ROUTE les périphériques ne peuvent être que du type ENTREE ou SORTIE : Pas de mélange.

Périphériques d'entrée : KB = clavier
RI = entrée série

Périphérique de sortie : DO = écran
PR = imprimante
RO = sortie série

Périphérique Fantome: NL = aucun périphérique

Périphériques à définir par l'utilisateur

MM = adresse d'entrée du DRIVER

Dans cette routine, les 12 premiers octets sont réservés par le DOS. la routine ROUTE effectue sur CALL au 13è octet suivant l'adresse spécifiée et tous les registres sauf AF, doivent être restitués par la routine.

Si MM est un périphérique entrée, l'octet d'entrée doit être fourni en A

Si MM est un périphérique sortie, l'octet à sortir est fourni dans C

ROUTE, CLEAR remet tous périphériques en gestion normale

Exemple : ROUTE, PR, DO Affiche à l'écran plutôt qu'à l'imprimante
ROUTE , DO , DO , PR Affiche l'écran à l'écran et à l'imprimante
ROUTE , PR , MM = 0FE80H Déroute la sortie imprimante vers une routine située en 0FE80H. (point d'entrée 0FE8CH)
ROUTE , DO , NL l'écran n'affiche plus rien
ROUTE , PR L'imprimante reprend sa sortie normale

2.38 - STMT

l'écran

Affiche un message à

STMT, msg

En mode direct, cette commande présente peu d'utilité et le message étant sur l'écran, elle ne l'affiche pas et retour à NEWDOS READY

Elle est utile sous APPELDOS ou en mode CHAINAGE

Elle affiche alors le message msg et poursuit sans rien attendre

Exemple : STMT , phase 1 terminée

2. 39 - SYSTEM

Affiche ou permet la mo-

dification de paramètres assurant la gestion du SYSTEME

SYSTEM , (Motpasse1 :) dn1 (, AA = yn) (, AB = yn) (, AC = yn)
(, AD = yn) (, AE = yn) (, AF = yn) (, AG = yn) (, AI = yn)
(, AJ = yn) (, AL = al) (, AM = am) (, AN = an) (, AO = ao)
(, AP = op) (, AQ = yn) (, AR = yn) (, AS = yn) (, AT = yn)
(, AU = yn) (, AV = av) (, AW = aw) (, AX = ax) (, AY = yn)
(, AZ = yn) (, BA = yn) (, BC = yn) (, BD = yn) (, BE = yn) (, BF =
yn) (, BG = yn) (, BH = yn) (, BI = bi) (, BJ = bj) (, BK = yn)
(, BM = yn) (, BN = yn)

Attention, le fait de changer les paramètres du SYSTEM sur le lecteur n'affecte pas immédiatement le fonctionnement en cours. Il faut re-faire une initialisation (ou un BOOT ou CMD «BOOT»)

Si les mots de passe sont validés, motpasse1 doit être le mot de passe de la disquette du lecteur dn1

Si aucun paramètre optionnel n'est spécifié, la commande n'effectue qu'un affichage des paramètres en vigueur sur dn1.

AA = yn si AA = y, les mots de passe sont validés
 si AA = n les mots de passe ne sont pas en
 vigueur

AB = yn si AB = y le SYSTEME est en mode
EXECUTION FORCEE. Les options AD = N ,
AE = N , AF = N seront forcées à l'initialisa-
tion. La touche ENTER ne stoppera plus la com-
mande AUTO. Le SYSTEME doit débiter par
l'exécution d'une commande AUTO et débou-
cher sur un programme utilisateur.
La touche BREAK est inefficace.
Si le SYSTEME se trouve en position d'erreur ou
de retour au DOS sans AUTO, il affichera
«MODE EXECUTION FORCEE»

AB = n le SYSTEME est en mode normal

AC = yn si AC = yn , SHIFT Ø est validé pour l'accès
ou miniscules
 si AC = n, SHIFT Ø est invalidé

AD = yn si AD = y ' JKL ' est validé (recopie écran)
 si AD = n ' JKL ' est invalidé

AE = yn si AE = y ' 123 ' est validé (accès DEBUG)
 si AE = n ' 123 ' est invalidé

AF = yn	si AF = y 'DFG' est validé (accès MINIDOS) si AF = n 'DFG' est invalidé
AG = yn	si AG = y la touche BREAK est validé si AG = n la touche BREAK est invalidé
AI = yn	si AI = n CTRL Ø est validé pour l'accès au souligné si AI = n CTRL Ø est invalidé
AJ = yn	si AJ = y ESC Ø est validé pour l'inversion clavier majuscules / minuscules si AJ = n ESC Ø est invalidé
AL = al	al (1-4) indique le nombre de lecteurs en service (afin de limiter d'inutiles explorations) La valeur d'origine est 2
AM = am	am est le nombre d'essais d'écriture / lecture sur disque avant de déclarer une erreur. La valeur d'origine est 10.
AN = an	an est le n° du lecteur par défaut à la commande DIR. La valeur d'origine est Ø
AO = ao	ao est le n° du lecteur à partir duquel le SYSTEME débute ses recherches, quand le n° du lecteur n'est pas spécifié dans la commande. Les lecteurs inférieurs sont ignorés. La valeur d'origine est Ø.
AP = ap	ap est la valeur désirée de l'HIMEM à l'initialisation (sauf si ap = Ø HIMEM = maximum). la valeur d'origine est Ø
AQ = yn	si AQ = Y la touche CLEAR est validée si AQ = n la touche CLEAR est invalidée
AR = yn	si AR = Y les formats 5 et 6 de COPY sont autorisés sans mot de passe, même si les mots de passe sont validés. si AR = N ils sont toujours requis s'ils sont validés.
AS = yn	si AS = Y , le BASIC convertit les chaînes de caractères de minuscules en majuscules ; ceci n'affecte que l' affichage écran et non le texte en mémoire. si AS = N, le Basic affiche les chaînes telles quelles.
AT = yn	si AT = y , toutes les entrées claviers se font depuis le fichier chaînage si AT = N, les entrées caractères par caractères (comme INKEY\$) se font depuis le clavier.
AU = yn	si AU = y, l'autorépétition des touches est validée en accord avec les paramètres AV et BJ si AU = n, l'autorépétition est invalidée
AV = av	av gère la vitesse d'autorépétition des touches la valeur d'origine est 16

AW = aw	aw est le nombre d'essais d'écriture avec vérification sur disques autorisés en relation avec le paramètre AM La valeur d'origine est 2
AX = ax	ax est la valeur maximum du code ASCII transmissible à l'imprimante. Au dessus de cette valeur, le SYSTEME remplace le code par un espace ou un point. La valeur d'origine est 90/5A H
AY = yn	si AY = Y , le BEEP clavier est validé à chaque appui sur une touche si AY = N , le BEEP est invalidé
AZ = yn	si AZ = Y , la reprogrammation des touches de fonction est possible si AZ = N , elle n'est pas autorisée.
BA = yn	si BA = Y , le SYSTEM effectue un ROUTE DO , NL à l'initialisation empêchant tout affichage à l'écran, tant qu'un ROUTE, CLEAR n'aura pas rétabli la connection. si BA = N le SYSTEM initialise avec un ROUTE , CLEAR
BC = yn	si BC = Y l'opérateur est autorisé à stopper un chaînage si BC = N inhibe ces possibilités
BD = yn	si BD = Y l'appui sur la touche ENTER durant l'initialisation inhibe le mode AUTO et renvoie au NEWDOS READY si BD = N inhibe cette possibilité
BE = yn	si BE = Y , la commande R du DOS est validée si BE = N , la commande R du DOS est invalidée
BF = yn	si BF = Y , les minuscules sont validées en mode SHIFT si BF = N , les minuscules sont validées en mode direct
BG = yn	si BG = Y , le DOS interprète les minuscules comme des majuscules si BG = N , le DOS n'effectue plus la transformation imposant ainsi les commandes en majuscules.
BH = yn	si BH = Y le curseur clignote à l'initialisation si BH = N le curseur est fixe à l'initialisation
BI = bi	bi = choix du caractère curseur
BJ = bj	bj gère le temps qui s'écoule entre l'appui sur une touche et le déclenchement de l'autorépetition. La valeur d'origine est de 25.
BK = yn	si BK = Y la commande DOS WRDIRP et les fonctions W et C du DIRCHECK sont autorisées. si BK = N Ces commandes sont invalidées.

BM = yn si BM = Y la vérification des disquettes s'effectue sur l'ensemble après formatage, effectuant ainsi une double vérification
 si BM = N seule la vérification normale, piste après piste s'effectue

BN = yn si BN = Y l'écriture des secteurs DIRECTORY, s'effectue compatible avec le TRS DOS mod I
 si BN = N l'écriture des secteurs DIRECTORY s'effectue compatible avec le TRS DOS mod III

Exemple : SYSTEM ,Ø, AL = 4 , AA = Y , AY = N
 réécrit les paramètres SYSTEME de la disquette du lecteur Ø avec AL = 4 (4 lecteurs) AA = Y (mode exécution forcée) AY = N (sans beep clavier). Il faudra effectuer une réinitialisation pour valider ces changements. Attention au mode exécution forcée.

2.40 - TIME

Gestion de l'heure de l'horloge.

TIME (, hh : mm : ss)

Sans paramètre, TIME affiche l'heure de l'horloge temps réel.

Le paramètre hh : mm : ss permet de remettre cette horloge à l'heure désirée.

hh de 00 à 23
mm de 00 à 59
ss de 00 à 59

Exemple :

TIME , 22 : 32 : 00

2.41 - VERIFY

Impose une lecture de vérification à chaque écriture sur les disquettes.

VERIFY (, yn)

Le NEWDOS 80 effectue toujours une vérification à l'écriture lorsque l'écriture s'effectue sur des enregistrements inférieurs au secteur (256 octets). Il ne l'effectue pas si l'écriture se fait par secteur complet (via l'adresse 4439 H).

VERIFY, Y ou VERIFY impose cette vérification.

VERIFY, N n'impose pas cette vérification.

Les fonctions COPY et les sauvegardes de BASIC et EDASS vérifient leurs écritures après l'écriture complète du BLOC.

Les fichiers BASIC effectuent également leurs vérifications à l'exception des fichiers à accès direct de LRL = 256 octets (sauf si VERIFY est actif).

2.42 - WDIRP

Réécrit les secteurs directory des disquettes.

WDIRP, dn 1

Cette commande, équivalente à la fonction W de DIRCHECK, réécrit les secteurs du DIRECTORY en fonction de l'option BN du système.

L'usage de cette commande étant relativement dangereux pour la disquette, on lui préférera l'utilisation de DIRCHECK et son option W.

dn1 indique le n° du lecteur où se situe la disquette visée.

3 - LES FACILITES DU NEWDOS 80

- 3.1 MINIDOS
- 3.2 CHAINAGE
- 3.3 APPELDOS
- 3.4 RECOPIE D'ECRAN
- 3.5 PROGRAMMES SOUS INTERRUPTIONS
- 3.6 DEBUG (64)

3.1 - MINIDOS

Il arrive souvent, dans le déroulement d'un programme principal, d'avoir besoin d'exécuter une commande DOS (comme DIR, ou PRINT...) sans perdre le fil de l'exécution en cours, pour poursuivre ensuite normalement.

Le **MINIDOS** offre cette possibilité

Pour y avoir accès, il faut que les options SYSTEM AB, AF, soient conformes à
AB = N
AF = Y

L'accès est assuré depuis le clavier par l'appui simultané des touches 'DFG'

Attention : à ne pas faire s'il y a des accès disques en cours

Sur cette action, le programme principal est interrompu et les registres sauvegardés et le message MINI-DOS READY est affiché

Toutes les commandes du DOS, à l'exception de APPEND, CHAIN (ou DO), COPY et FORMAT, sont exécutables alors sous MINI-DOS

Des programmes (externes aux commandes DOS) ne peuvent s'exécuter sous MINI-DOS

Des copies simples de FICHIER sont réalisables par **MDCOPY**

Le retour au programme principal est assuré par la commande **MDRET**. Le programme principal reprendra son exécution où il avait été interrompu par 'DFG' (déroulement de séquence ou saisie clavier, le buffer étant laissé intact)

Le retour au DOS est assuré par **MDBORT**. Dans ce cas, le retour se fait au NEWDOS READY

Exemple : Sous BASIC, lancez le programme
 10 - PRINT "HELLO" : GOTO 10
 RUN

Pendant le défilement, frappez 'DFG'

Sous MINIDOS READY, exécutez les commandes suivantes :

DIR
FREE
CLOCK
CLOCK, N
LIB
PDRIVE Ø
MDRET

et le programme BASIC reprend son exécution normalement.

Refrappez 'DFG' puis **MDBORT** . Vous êtes revenu au NEWDOS
READY

3.2 - CHAINAGE

Les commandes DOS CHAIN et DO présentent deux syntaxes différentes pour une même action, pour des questions de compatibilité avec d'autres systèmes d'exploitation. Dans ce chapitre, nous n'emploierons que CHAIN (DO restant bien sûr son équivalent total)

Il arrive souvent qu'un utilisateur, pour lancer un programme donné, ait des actions répétitives à effectuer depuis le clavier, sources d'erreur et de perte de temps, surtout si l'utilisateur, novice, n'est pas lui-même concepteur du programme. Le CHAINAGE offre la possibilité de convertir en variables sur fichier séquentiel, toutes les saisies au clavier et en faire ainsi une routine utilisable par la commande unique CHAIN

Exemple : Soit pour lancer le programme PROBAS/BAS
L'obligation d'effectuer depuis le DOS

```
HIMEM, ØE800 H
PROMACHINE 1
Y                ; réponse à question de Promachine1
50              ; réponse à question de Promachine1
PROMACHINE2
3723            ; réponse à question de Promachine2
BASIC, RUN "PROBAS/BAS"
Y                ; réponse à question de PROBAS/BAS
```

Le rangement de ces séquences en fichier séquentiel nommé XXX/JCL permet leur exécution automatique (avec affichage à l'écran) par la commande **CHAIN XXX**

A la fin de l'exécution du chaînage, la main sera redonnée au clavier normal dans le programme en cours

Un lancement automatique serait de plus assuré à l'initialisation par l'exécution préalable de la commande **AUTO CHAIN XXX**

utilitaire CRECHAIN dans le BASIC

Le NEWDOS n'assume pas l'élaboration des fichiers de chaînage, mais un utilitaire **CRECHAIN**, fourni avec le SYSTEME INITIAL, permet de créer, éditer, modifier, les fichiers de CHAINAGE. Il se trouve également sur la disquette un exemple de fichier de chaînage

TSTCHAIN/JCL

A l'utilisation, en fonction de l'option AT du SYSTEM (voir section 2.39), la fonction permet :

- si AT = Y toutes les entrées claviers sont assurées par le fichier (en BASIC INPUT, LINEINPUT, INKEY\$)
- si AT = N les entrées claviers par caractères (comme INKEY\$) sont assurées par le clavier lui-même

Cette option permet d'offrir des choix durant un enchaînement programmé

L'allocation sur disque se faisant par granules (1280 octets), il est quelquefois regrettable d'ouvrir un fichier de chaînage pour n'en occuper que 100 à 200 octets. La possibilité de diviser un fichier de chaînage en SECTIONS (/./Ø voir plus loin) que l'on peut appeler de façon sélective et qui se comportent comme un seul fichier, permet d'éviter ce gaspillage d'espace disque.

On peut interrompre et reprendre un enchaînement en cours par les commandes DOS CHNON, Y CHNON, N CHNON, D (voir section 2.8)

Le mode chaînage peut interpréter 6 codes de contrôle en début de variables du fichier séquentiel. Ces codes ne sont pas affichés durant l'exécution mais assurent l'exécution d'une commande particulière.

- / . / Ø (ou 80 H) ce code identifie le début d'une SECTION
Exemple : / . / Ø xxxx identifie le début de la SECTION XXXX
- / . / 1 (ou 81 H) ce code équivaut à la fonction PAUSE du DOS. La fin d'enregistrement est affichée et le système attend l'appui sur ENTER pour poursuivre.
- / . / 2 (ou 82 H) ce code équivaut à la fonction REM du BASIC. La fin d'enregistrement n'est pas affichée sans marquer d'arrêt. Elle sert de commentaire au listing du fichier
- / . / 3 (ou 83 H) ce code équivaut à la fonction STMT du DOS.. La fin d'enregistrement est affichée sans marquer d'arrêt
- / . / 4 ce code permet de sauter d'une section à une autre dans un même fichier
Exemple : / . / yyyy le chaînage se poursuit dans le même fichier à la section yyyy
si la section n'existe pas, une erreur est générée : EOF RENCONTRE
- / . / 5 (ou 85 H) suivi de Y, N, D équivaut à la commande CHNON du DOS (voir section 2.8)

Exemple de fichier de chaînage :

```
- / . / Ø SECTION 1
- DIR 1
- BASIC
- CMD " S
- / . / 4 SECTION 3
- / . / 2 ON NE VIENT PAS ICI ET JE NE SERS
  A RIEN
- / . / Ø SECTION 3
- BASIC
- 10 INPUT "VOTRE AGE" ; A
- 20 IF A ≥ 40 THEN PRINT "LA FLEUR DE
  L'AGE" ELSE PRINT
  " TOUT A APPRENDRE"
- RUN
- 42
- / . / 3 JE NE SUIS PAS FATIGUE
- / . / 1 ON CONTINUE ?
- DIR Ø
- LIB
- / . / 3 MAINTENANT JE STOPPE
```

Voyez l'utilitaire de création de fichier chaînage (section 6-2) et essayez donc ce petit encas.

3.3 - APPELDOS

Le NEWDOS 80 2.0 permet à n'importe quel programme en langage machine d'effectuer un CALL 4419 H (voir section 5) et d'exécuter par là même une commande DOS ou de faire tourner un autre programme utilisateur. Cette possibilité c'est l'APPELDOS. Le BASIC exécute cette routine au travers de la commande CMD "ComDOS"

Le programme demandeur doit définir une commande DOS dans un buffer de son choix, terminée par un CR (0D H) et le registre HL pointant sur le 1er caractère avant le CALL 4419 H. Le DOS transpose alors cette commande dans son propre buffer en effectuant la conversion minuscules/MAJUSCULES

Attention, le DOS n'effectue aucun contrôle sur le conflit latent entre le programme appelé et le demandeur, le programme appelé se chargeant au mépris de l'écrasement de tout ou partie du demandeur : au programmeur d'y pourvoir.

Les registres ne peuvent servir à transmettre des paramètres d'un programme à un autre sous APPELDOS. 2 octets en mémoire centrale sont réservés à cet effet en adresse 4403 H et 4404 H.

Au retour d'un APPELDOS, le programme demandeur doit tester 3 octets

- Carry à 1 une erreur DOS s'est produite et a déjà été affichée (pour stopper sous APPELDOS, JP 4030 H)
- NZ et NC une erreur Dos s'est produite et n'a pas été affichée et le code erreur est dans A (bit 6 et 7)
- NC et Z tout a bien marché. Tous les registres sont rendus intacts sauf AF. 4403 H et 4404 H peuvent resservir dans l'autre sens de boîtes aux lettres.

3.4 - RECOPIE D'ECRAN

Le NEWDOS permet, depuis le clavier, de déclencher à tout moment une RECOPIE ECRAN sur Imprimante, par appui simultané des 3 touches 'JKL'.

Cette possibilité est soumise à l'option SYSTEM AD = Y et le DOS ne doit pas utiliser sa zone de recouvrement (4D00 H à 51FF H)

Ces conditions remplies, l'appui sur 'JKL' provoque une interruption du programme principal, la sauvegarde de ses états, l'édition complète de l'écran (en accord avec l'option AX pour les graphiques et soulignés)

Si l'imprimante n'est pas disponible, le DOS vous l'annoncera et vous pourrez y apporter remède.

Un appui sur la touche BREAK interrompt l'impression et renvoie au programme principal

En fin d'écran, la main est rendue au programme principal.

Attention la routine JKL n'étant pas en mémoire centrale, est appelée du disque pour exécution. Elle écrase donc la zone de recouvrement du DOS.

3.5 - PROGRAMMES SOUS INTERRUPTIONS

Toutes les 25 ms, l'électronique génère un top, appelé interruption, lequel peut provoquer par l'intermédiaire du Microprocesseur Z 80, l'exécution d'une routine ou d'un programme. Voir les routines DOS 4410 H et 4413 H (section 5)

Cette particularité est fort intéressante car elle permet, hors fonctionnement d'un programme principal, une inspection permanente d'un paramètre (par exemple l'heure) pour déclencher une routine (exemple un REVEIL)

3.6 - DEBUG (EN VERSION 64 COLONNES)

Ce DEBUG est une aide primaire à la programmation en langage machine. Il doit être utilisé par un initié à ce type de programmation.

Il permet l'inspection et la modification mémoire, l'inspection et la modification sur disque, une exécution pas à pas, ...

L'appel de DEBUG se fait par appui simultané des touches '123' (validé si l'option SYSTEME AE = Y).

A l'entrée, DEBUG sauvegarde tous les registres, affiche la page mémoire selon le mode et l'adresse précédent la dernière sortie et attend vos ordres (curseur bas gauche).

Toutes les commandes doivent se terminer par 'ENTER'.

Commandes :

X fait passer (si ce n'était pas le cas) en mode ETATS.

c'est à dire : 4 lignes avec 64 octets de mémoire : AIRE 1
la 5^è ligne affiche le contenu des registres AF, BC, DE et HL.

4 lignes avec 64 octets de mémoire : AIRE 2
la 10^è ligne affiche le contenu des registres AF', BC', DE' et HL'.

4 lignes avec 64 octets de mémoire : AIRE 3
la 15^è ligne affiche le contenu des registres PC, SP, IX, IY.

l'affichage des registres AF et AF' comprend les codes des registres d'état. — Si le bit d'état est à 0 et les codes suivants s'ils sont à 1 : S = signe

Z = zéro

L = non utilisé

H = half - carry

L = non utilisé

P = débordement

N = soustraction

C = carry

S fait passer (si ce n'était pas le cas) en mode PAGES qui affiche 256 octets en 16 lignes de 16 débutant à l'adresse arrondie module 256.

- (n) **D addr** en mode **PAGES**, les 256 octets du bloc contenant **addr** sont affichés et **n** (n° d'aire) est mis en mémoire.

en mode **ETATS**, **n** décide de l'aire désignée par la demande et qui affichera les 64 octets contenant l'adresse **addr** (à défaut **n**= 1).

Exemple : en mode **S (PAGES)** **D4207** affiche la page de 4200 H à 42FF H.

en mode **X (ETATS)** **2 D4207** affiche dans l'aire n° 2 de 4200 H à 4240 H.

- (n) ; en mode **PAGES**, les 256 octets suivant en ; précédent en —
(n) — sont affichés.

en mode **ETATS**, les 64 octets suivant en ; précédent en — sont affichés dans l'aire **n** (1 par défaut).

M addr le mode **PAGES** est forcé. la page contenant **Addr** est affichée et le curseur clignote sur l'octet concerné. Vous pouvez modifier en frappant directement les caractères souhaités. Le curseur avance automatiquement d'une position, les 4 flèches permettent de se déplacer dans la page. 'ENTER' clôture le mode modification. Une erreur de touche quitte le mode modification et affiche **ERROR 'ENTER'** pour reprendre le cours normal.

F (addr) (,01) (,02) (,03) (,04)

effectue la recherche à partir de l'adresse **addr** (1 à défaut) la chaîne d'octet 01, 02, 03, 04 s'ils sont spécifiés. Si trouvé, le mode **X** est forcé et la chaîne de caractères est affichée à la 3^e ligne de l'aire 1. Sinon, le mode **X** est forcé et le dernier cadran de 64 octets (FF00 H, FFFF H) est affiché Aire 1.

Exemple : **F 6000, 44, CD, 09** , la chaîne 44 CD 09 est cherchée à compter de l'adresse 6000 H.

Attention, la chaîne à chercher est stockée en 51xx H, donc un trouvé à cet endroit là n'est pas forcément différent de votre demande.

- I** Exécute l'instruction courante du programme interrompu par **DEBUG '123'** s'il y a lieu (**ERROR** sinon) et revient à **DEBUG**.

Ceci vous permet de suivre pas à pas le déroulement des registres et d'une zone de mémoire pendant l'exécution d'un programme utilisateur.

Attention, durant une séquence, il peut se produire :

— Une routine complète d'interruption qui perturbe les états.

— le pas à pas interdit les **JP** ou **CALL** sous 5200 H.

- C** Exécute également un mode pas à pas, mais si l'instruction est un **CALL**, l'exécution se poursuit jusqu'au **RET** avant le retour au **DEBUG**.

R reg, val place dans le double registre **reg** la valeur **val**.

Exemple : RDE, C000 place C000 dans DE
RHL, 7100 place 7100 dans HL.

L dn, drs charge dans le buffer disque du DOS (4200 - 42FF) le secteur relatif **drs** du lecteur **dn**.

L'affichage est forcée en mode PAGE et pointe sur 4200 - 42FF H.

Cette fonction déclenche une ERROR si les mots de passe sont validés.

Exemple : L1, 150 charge et affiche le buffer disque avec le 150^e secteur et la disquette du lecteur 1.

WR dn, drs Ecrit sur la disquette, sur le secteur relatif **drs**, dans le lecteur **dn**, le contenu du buffer disque 4200 - 42FF H.

Pendant l'affichage du Buffer, toutes les modifications (ou recherches) sont possibles (fonction M et F), elles ne seront prises en compte qu'à l'écriture.

Soyez prudent dans l'utilisation de cette fonction, ne vous trompez pas de lecteur, de secteur et assurez-vous du bon contenu du buffer.

Q Retour au NEWDOS READY

G (addr1) (, addr2) (, addr3)

Cette fonction restaure les registres et quitte le DEBUG.

Si **addr1** est spécifié, un branchement se fait à cette adresse (à défaut, le branchement se fait à l'adresse suivante dans la pile, c'est-à-dire retour au CALLER).

addr2 et **addr3** permettent d'effectuer des retours commandés au DEBUG en y plaçant des RST 30 H. L'octet antérieur est sauvegardé et sera restauré au retour au DEBUG.

addr2 et **addr3** doivent être supérieures à 5200 H.

Exemples : G 8000, 8050, 8300 place deux RST 30 H en 8050 et 8300 pour retour au DEBUG et se branche en 8000 après avoir restauré les registres.

G restaure les registres et retourne à l'adresse mémoire du registre de pile (là d'où il est parti par '123').

4 - LES MODULES DU DOS

Structure et informations

Le NEWDOS 80 2 . 0, malgré toute l'étendue de ses possibilités ne prend pratiquement pas de place en mémoire centrale.

Contrairement à certains de ses grands successeurs, il n'occupe pas plus de 64 K, mais au contraire se réserve à peine 5 K.O pour fonctionner, entre 4000 H et 5200 H.

Cette zone comprend une zone de communication, un noyau DOS résident et une zone de recouvrement qui sert au chargement de fichiers SYSTEMES en provenance du disque et qui ont pour fonction d'exécuter la commande DOS pour laquelle ils sont sollicités.

C'est donc grâce à cette astuce, que 23 fichiers SYSTEMES, portant l'extension /SYS assurent depuis le disque, sur les ordres du NOYAU DOS RESIDENT, l'ensemble des commandes DOS, et de certaines commandes BASIC.

L'objet de cette section est d'ailleurs de passer en revue l'ensemble de ces fichiers, afin que vous sachiez ce qu'il peut vous en coûter d'en supprimer quelques uns du disque.

4.1 - LES FICHIERS /SYS INDISPENSABLES SUR CHAQUE DISQUETTE

DIR/SYS (2 à 6 GRANULES) c'est le DIRECTORY de la disquette. **FORMAT** assure sa mise en place et le DOS s'occupe de sa mise à jour (voir SECTION 7)

BOOT/SYS (1 GRANULE) Ce fichier occupe la première granule du disque et contient tous les paramètres essentiels (ou ne les contient pas) pour initialiser le SYSTEME. **FORMAT** et **COPY** créent et mettent à jour ce fichier.

NE JAMAIS DETRUIRE CES FICHIERS SUR UNE DISQUETTE. ELLE DEVIENDRAIT INUTILISABLE.

4.2 - LES FICHIERS / SYS, MODULES DU DOS

SYS 0 / SYS	16 GR	Module de chargement du DOS, contient le chargeur, le noyau résident, le moniteur, le SBASIC.
SYS 1 / SYS	1 GR	Module d'exécution des commandes DOS. Exécute LIB.
SYS 2 / SYS	1 GR	Module de création de fichiers, allocation espace disque, de gestion des mots de passe et de chargement de programme. Exécute RENAME et LOAD .

SYS 3 / SYS	1 GR	Module de fermeture des fichiers, de destruction des fichiers, de gestion des interruptions. Exécute BLINK, BREAK, CLOCK, JKL, VERIFY, KILL et le plus gros de PURGE.
SYS 4 / SYS	1 GR	Module d'affichage des erreurs DOS
SYS 5 / SYS	1 GR	Module DEBUG (version 64 colonnes).
SYS 6 / SYS	8 GR	Recouvre la zone 4D00 H - 6FFF H Exécute FORMAT, COPY et APPEND
SYS 7 / SYS	1 GR	Exécute TIME, DATE, AUTO, ATTRIB, PROT, DUMP, HIMEM et la première partie de PURGE, SYSTEM et PDRIVE.
SYS 8 / SYS	2 GR	Exécute DIR et FREE.
SYS 9 / SYS	1 GR	Exécute BOOT, CHAIN, CHNON, MDCOPY, PAUSE et STMT.
SYS 14 / SYS	1 GR	Exécute CLEAR, CREATE, ERROR, LIST et PRINT.
SYS 16 / SYS	1 GR	Exécute le plus gros de PDRIVE.
SYS 17 / SYS	1 GR	Exécute WRDIRP et le plus gros de SYSTEM.
SYS 23 / SYS	1 GR	Module de programmation des touches de fonction + CAR, FONCAR, FONECR.

4.3 - LES FICHIERS / SYS, MODULES BASIC

SYS 10 / SYS	1 GR	Exécute GET et PUT
SYS 11 / SYS	1 GR	Exécute RENUM.
SYS 12 / SYS	1 GR	Exécute REF
SYS 13 / SYS	1 GR	Module d'affichage des erreurs BASIC (1ère partie) et de l'appel de RENUM.
SYS 18 / SYS	1 GR	Exécute les commandes en mode direct. Doit être sur la disquette SYSTEME pour utiliser BASIC.
SYS 19 / SYS	1 GR	Exécute les commandes BASIC, RUN, LOAD, MERGE, SAVE, CMD "F = DELETE", doit être sur la disquette SYSTEME pour utiliser BASIC.
SYS 20 / SYS	1 GR	Exécute la plupart des instructions du BASIC Disque, doit être sur la disquette SYSTEME pour utiliser BASIC.

SYS 21 / SYS	1 GR	Exécute CMD "O" (tri machine).
SYS 24 / SYS	1 GR	Module d'affichage des erreurs BASIC (2ème partie).

4.4 - UTILITAIRES PRESENT SUR LA DISQUETTE SYSTEME MAITRE.

EDASS	Editeur. Assembleur Z 80 sur disque (voir section 6.1).
CRECHAIN	Utilitaire de création de fichier de chaînage modification et édition. (voir section 6.2).
TST CHAIN / JCL	Fichier de test des commandes DO et CHAIN (voir section 6.2).
COPIMAS / JCL	Utilitaire de recopie sélective des disquettes SYSTEME. Passage de simple face en double face.
FONC / CMD	Utilitaire de gestion des touches de fonction sauvegarde, chargement, édition : écran et imprimante. (voir section 6.3).
AZERTY / CMD	Fichier de transfert clavier QWERTY → AZERTY
QWERTY / CMD	Fichier de transfert clavier AZERTY → QWERTY
BASIC / FON	Fichier fonction BASIC (voir section 6.3).
EDASS / FON	Fichier fonction EDASS (voir section 6.3).

4.5 - DISQUETTES A SYSTEME REDUIT

Il est possible, après recopie de sécurité, de créer des disquettes à SYSTEME réduit en détruisant (si les mots de passe sont inactifs) les fichiers SYSTEME de votre choix.

Attention, le minimum requis est composé des fichiers BOOT / SYS et DIR / SYS bien sûr et des fichiers SYS 0 / SYS à SYS 4 / SYS.

Tous les autres sont facultatifs, mais leur absence vous prive d'une (ou plusieurs) fonction DOS.

L'utilisation du BASIC requiert la présence des modules :
SYS 13 / SYS, SYS 18 / SYS, SYS 19 / SYS, SYS 20 / SYS,
SYS 24 / SYS.

Les autres sont facultatifs mais vous privent de certaines commandes.

L'appel à une commande qui aboutirait à un fichier SYSTEME absent vous rendra la main et affichera le message d'erreur :
'PROGRAMME SYSTEME NON TROUVE'

ATTENTION : Une fois réduite, une disquette mini-SYSTEME ne peut être reconfigurée en SYSTEME plus élaborée par simple recopie du (ou des) modules DOS requis. En effet, leur place sur la disquette doit être bien précise et il est fort improbable qu'il retrouve la même.

5 - LES ROUTINES DOS

Cette section, destinée aux initiés du langage Assembleur, indique les principaux points d'entrée du DOS et les routines qui y sont liées, ainsi que le moyen d'en tirer honnêtement parti.

Ces routines ont des conditions d'entrée et de sortie dont certaines se retrouvent souvent, et dans l'étude qui va suivre nous emploierons un code pour définir ces conditions.

- A** seul le registre AF est altéré par la routine. Tous les autres sont sauvés à l'entrée et restitués à la sortie.
- B** A la sortie Z si aucune erreur durant l'exécution NZ si erreur et A contient le code de l'erreur DOS.
- C** A l'entrée, DE doit indiquer l'adresse du 1er caractère d'un FCB en RAM (FCB voir section 7).
- D** Si l'entrée doit se faire par un CALL (fin de routine par RET).
- E** Si l'entrée doit se faire par un JP (retour DOS en fin de routine par JP 402D H).

5.1 - 0013 H LECTURE D'UN OCTET SUR LE DISQUE

A.B.C.D. l'octet lu est rendu dans le registre A.

5.2 - 001B H ECRITURE D'UN OCTET SUR LE DISQUE

A.B.C.D. l'octet à écrire doit être placé dans le registre A.

5.3 - 402D H RETOUR AU DOS SANS ERREUR

E. Le DOS teste son état en suivant :

sous MINIDOS ou APPELDOS, le pointeur de pile est remis en l'état d'avant la dernière commande DOS, sinon il reprend sa place et la touche BREAK est validée (ou non) suivant l'option AG du SYSTEME.

sous APPELDOS ou si le chaînage ne doit pas se poursuivre en DOS, tous les registres sauf AF sont restaurés dans leur état d'entrée, le flag Z à 1, et un retour est effectué au niveau d'APPELDOS.

si le mode exécution forcée est en service et le chaînage inactif le message 'MODE EXECUTION FORCEE' est affiché et le DOS attend l'appui de 'ENTER' pour réinitialiser.

sous APPELDOS et en chaînage, le DOS appelle la nouvelle ligne du fichier chaînage.

sous MINIDOS, le message MINIDOS READY est affiché et le DOS attend une nouvelle commande.

sous chaînage, le DOS appelle la nouvelle ligne du fichier chaînage.

5.4 - 4030 H RETOUR AU DOS AVEC ERREUR DEJA AFFICHEE

E

L'exécution sera la même qu'un JP 402D H sauf :

- si un chaînage est en cours, il est stoppé et annulé
- le retour sous APPELDOS s'effectue avec le Flag C à 1

5.5 - 4400 H Idem 402DH

5.6 - 4405 H EXECUTION D'UNE COMMANDE DOS AVEC RETOUR AU DOS

E

Le retour au DOS est effectué, le pointeur de pile est remplacé en zone DOS

Le registre HL doit pointer sur une commande n'excédant pas 80 caractères et terminée par un 0D H

Le DOS transfère cette commande dans son buffer et l'exécute

5.7 4409 H RETOUR AU DOS AVEC ERREUR

E Si le bit 7 du registre A est à 0

le code erreur est dans A, et suivant l'état du DOS:

- Un chaînage en cours est annulé
- un retour APPELDOS est effectué registre NZ et NC à 1 et le code erreur DOS dans le registre A
- le message d'erreur est affiché suivi d'un JP 402D H

D Si le bit 7 du registre A est à 1 et 4409 H appelé par un CALL

- le message d'erreur correspondant est affiché et un RET est effectué avec F altéré.

5.8 - 440D H ENTREE AU DEBUG (version 64 colonnes)

D

5.9 - 4410 H BRANCHE UNE ROUTINE UTILISATEUR SUR LES INTERRUPTIONS

D

Utilise tous les registres

DE doit pointer sur la routine utilisateur dans le format suivant :

- 1 et 2^{es} octets réservés au DOS, valeur sans objet
- 3^{ème} octet nombre de 25 ms entre chaque appel routine (ex. chaque seconde = 40 (28 H))
- 4^{ème} octet position du compteur (à l'initial 0 < 4^{ème} oct <= 3^{ème} oct)

Lorsque ce 4^{ème} octet est à 0, on exécute la routine et on repart pour un cycle. La valeur du 3^{ème} octet est remise dans le 4^{ème}, les registres sont préservés et un call 5^{ème} et 6^{ème} octet est effectué. Les interruptions sont invalidées et doivent le rester

5.10 - 4413 H DEBRANCHE UNE ROUTINE INTERRUPTION

- D Tous les registres sont altérés. La routine pointée par DE est débranchée et ne participe plus aux interruptions

5.11 - 4416 H RELANCE LE MOTEUR DES DRIVES

- D.A. Si les Drives tournent, cette routine resélectionne le DRIVE courant, et relance les moteurs pour 2 secondes 4 environ. AF est altéré.

5.12 - 4419 APPELDOS

- D.A. Cette routine exécute une commande DOS et RET. Elle utilise la pile du caller (programme appelant). La commande à exécuter doit être pointée par HL, ne pas excéder 80 caractères et se terminer par un ØD H.
Si l'APPEL DOS appelle un programme, il doit se terminer par un des trois JP possible (402D H, 4030 H ou 44Ø9 H) et avant de quitter, il peut stocker 2 octets de paramètres en 4403 H, 4404 H.

5.13 - 441C H TRANSFERT DE CARFICH

- D Cette routine extrait du texte pointé par HL le carfich et le replace dans un buffer (32 Octets) pointé par DE et clôture avec un Ø3 H. AF, BC, HL sont altérés

Si le premier caractère est un A-Z ou Ø-9 ou *, le texte est transposé dans le buffer pointé par DE, tant que le caractère suivant est A-Z, Ø-9, /, . ou :, ou jusqu'au 32ième caractère.

Z est mis à 1 si tout est correct, sinon à Ø

Cette routine ne fait aucun autre contrôle de validation sur Carfich.

5.14 - 4420 H OUVERTURE D'UN FCB POUR UN FICHIER (nouveau ou existant)

- A.B.D. Cette routine fait appel à 4424H. Si le fichier existe, le RET est assuré avec Z et NC à 1

Si le fichier n'existe pas, il est créé par cette routine. Si le n° du drive est spécifié, et si la disquette le permet (FDE libre), le fichier est créé sur ce lecteur. Si le n° de Drive n'est pas spécifié, la recherche se fait sur les disquettes montées en fonction de l'option AO du SYSTEME, jusqu'à trouver un FDE libre, sinon une erreur est générée.

Créer un fichier consiste à placer un FPDE dans le FDE libre (voir SECTION 7). Le registre B doit contenir la largeur d'enregistrement souhaitée (Ø = 256)

Une fois le fichier créé, un appel est fait à 4424 H pour réaliser le OPEN et un RET est effectué avec Z et C à 1

Le DOS ouvre le fichier sans se préoccuper de la place restant libre sur le disque.

5.15 - 4424 H. EXECUTION D'UN OPEN (FCB en RAM) POUR UN FICHIER EXISTANT

A. B. D. A l'entrée DE pointe sur un buffer contenant les CARFICH'S du fichier à ouvrir, HL pointe sur un Buffer de 256 octets destiné aux entrées / sorties secteur du disque et B contient la LRL (longueur d'enregistrement). La recherche sur disque se fait comme pour la routine 4420 H. Si le fichier n'est pas trouvé, une erreur est générée.
Si les mots de passe sont validés, une investigation est menée pour vérifier la concordance sur l'accès demandé et le mot de passe. S'il y a lieu, une erreur est générée, sinon le FCB contiendra le niveau d'accès autorisé.

Le FCB est alors mis en buffer pointé par DE (en lieu et place du Carfich) (voir section 7)

5.16 - 4428 H FERMETURE D'UN FCB (CLOSE)

ABCD Cette routine détruit la connection entre le FCB et le buffer d'E/S disque alloué au fichier. Si le 4^e bit du 2^e octet est à 1, le buffer est transposé sur disque (call 4439 H), et le FPDE est remis à jour sur le disque, s'il y a lieu. L'allocation (ou le contraire) d'espace sera faite conformément aux paramètres du FCB

5.17 - 442C H DESTRUCTION D'UN FICHIER (KILL)

ABCD Cette routine nécessite d'avoir transposé le FCB en RAM (donc un OPEN). le FCB est rempli de 00

5.18 - 4430 H CHARGEMENT D'UN FICHIER PROGRAMME

AB.D AF, BC et HL sont altérés et en RET, HL contient l'adresse d'entrée du programme.

A l'entrée DE doit pointer sur un buffer contenant le Carfich (32 octets Max)

5.19 - 4433 H CHARGEMENT ET LANCEMENT D'UN FICHIER PROGRAMME

E AF et BC sont altérés, l'exécution est identique à celle de 4430 H mais avec JP HL

En cas d'erreur, le retour se fait en 4409 H.

5.20 - 4436 LECTURE D'UN SECTEUR DISQUE OU TRANSFERT D'UN ENREGISTREMENT DU BUFFER FCB AU BUFFER UTILISATEUR

Si le bit 7 du 2^e octet du FCB est à 0, le secteur pointé est lu et transféré dans le Buffer FCB. Sans erreur, le secteur suivant est pointé (+ 256 au NF)

Si le bit 7 du 2^e octet du FCB est à 1, la LRL est transférée du Buffer FCB vers un buffer pointé par HL. A chaque octet, le NF est incrémenté. En fin de buffer FCB, le secteur suivant est lu automatiquement et on continue jusqu'à LRL.

En cas d'erreur, le mouvement s'interrompt, laissant NF pointé sur l'octet suivant.

Si le bit 1 du 1^e octet FCB est à 1, NF et EOF sont considérés comme des RBA de la disquette, autorisant ainsi l'utilisateur à lire la disquette plutôt qu'un fichier.

5.21 - 4439 H ECRITURE (SANS VERIFIER) D'UN SECTEUR SUR LE DISQUE OU TRANSFERT DU BUFFER UTILISATEUR VERS LE BUFFER FCB

ABCD Cette routine procède à l'écriture comme la précédente à la lecture.

5.22 - 443C H ECRITURE AVEC VERIFICATION D'UN SECTEUR SUR LE DISQUE

ABCD Cette routine procède comme la précédente mais avec vérification de l'écriture sur le disque.

5.23 443F H REPLACE EN DEBUT DE FICHIER

ABCD Si le FCB attend une écriture sur disque (bit 4 du 2^e octet à 1) l'écriture est lancée via 4439 H. Le NF est placé à 0. le bit 5 du 2^e octet est placé à 0, indiquant que le buffer FCB ne contient pas le secteur courant.

5.24 - 4442 H POSITIONNE LE FCB SUR L'ENREGISTREMENT SPECIFIE

ABCD Le NF est positionné sur le RBA de l'enregistrement souhaité en relatif (0 = le 1^{er}) dans BC

Une écriture éventuelle du secteur courant est faite sur le disque en fonction du changement de secteur impératif et du bit 4 du 2^e octet.

5.25 - 4445 H RECULE LE FCB D'UN ENREGISTREMENT

ABCD Idem 4442 H, avec $NF = NF - LRL$

5.26 - 4448 H POSITIONNE LE FCB SUR L'EOF

ABCD Idem 4442 H avec $NF = EOF$

5.27 - 444B H ALLOCATION D'ESPACE DISQUE

ABCD Si le NF est plus élevé que le nombre de secteurs jusqu'à alloués, cette routine effectue, dans la mesure du possible, l'allocation complémentaire.

Ce qui permet en plaçant NF à la position souhaitée, de pré-allouer de la place sur disque.

5.28 - 444E H POSITIONNE LE FCB SUR LE RBA SPECIFIE.

ABCD Idem 4442 H avec $NF = RBA$ fourni dans HL et C (3 octets).

5.29 - 4451 H REMET A JOUR L'EOF DANS LE FPDE

ABCD Si l'EOF du FCB est différent de celui du FPDE, celui-ci est remis à jour.

5.30 - 445B H SELECTIONNE ET DEMARRE LE LECTEUR SPECIFIE

AB A doit contenir le n° du lecteur.

5.31 - 445E H TESTE SI UNE DISQUETTE EST PRESENTE

AB Effectue un CALL 445B H et teste si une disquette tourne (index Pulse). Sinon renvoi le code de l'erreur : "PERIPHERIQUE INDISPONIBLE" dans A.

5.32 - 4461 H BRANCHE LA ROUTINE * NOM

D Le registre HL pointe sur une routine utilisateur en RAM pour enchaînement.

Les 12 premiers octets de la routine sont définis comme suit :

- 4 octets réservés pour le DOS
- 8 octets pour le nom de la routine, complétés éventuellement de blanc à droite.

Si une routine du même nom existe déjà et est branchée, le code erreur FICHIER DEJA EXISTANT est retourné en A avec NZ à 1. Sinon la routine se branche et Z est positionné à 1. HL, DE, BC et AF sont altérés.

Ces routines répondent aux commandes DOS * nom (, param), dans ce cas, le DOS cherche cette routine dans ses branchements, la trouve, pointe HL sur les paramètres et JP au 13e octet, le retour se fera via 402D H, 4409 H ou 4030 H. Les adresses 4403 H et 4404 H peuvent être utilisées comme transit de données.

Si la routine n'est pas trouvée, le code erreur "FICHIER INEXISTANT" est retourné avec NZ à 1.

5.33 - 4464 H DEBRANCHE LA ROUTINE * NOM

D Idem 4461 mais pour la débrancher.

5.34 - 4467 H AFFICHE UN MESSAGE A L'ECRAN

AD Le message pointé par HL, jusqu'au 0D H inclus ou jusqu'au 03 H exclus est affiché sur l'écran.

5.35 - 446A H AFFICHE UN MESSAGE SUR L'IMPRIMANTE

AD Idem 4467 H mais sur l'imprimante.

5.36 - 446D H CONVERSION DE L'HEURE AU FORMAT HH : MM : SS

D Conversion de l'heure au format hh : mm : ss et stockage dans un Buffer de 8 octets pointé par HL.

AF, BC, HL, DE sont altérés.

En sortie HL pointe vers le 9e octet du Buffer.

5.37 - 4470 H CONVERSION DE LA DATE AU FORMAT JJ/MM/AA

D Idem 446D H, mais pour la date.

5.38 - 4473 H ADJONCTION PAR DEFAULT D'UNE EXTENSION FICHIER

D Si le carfich pointé par DE ne possède pas d'extension, cette routine y place l'extension pointée par HL, le résultat ne doit pas excéder 31 caractères. AF et HL sont altérés.

6 - LES UTILITAIRES

REPERTOIRE DES UTILITAIRES

6.1 - EDASS / CMD

6.2 - CRECHAIN + TSTCHAIN / JCL

6'3 - FONC / CMD + BASIC / FON + EDASS / FON

6.1 - EDASS : EDETEUR ASSEMBLEUR SUR DISQUE

Cette section ne constitue pas une formation au langage assembleur.

De nombreux ouvrages en traitent parfaitement bien le sujet.

Elle est un mode d'emploi de l'éditeur / assembleur disque fournit avec le NEWDOS 80 . 2 0.

Par Convention :

- . Le point à la place d'un numéro de ligne représente la ligne courante, la dernière assemblée, listée ou éditée.
- # Le dieze à la place d'un numéro de ligne représente la première ligne du buffer de texte.
- * L'astérisque à la place d'un numéro de ligne représente la dernière ligne du buffer de texte.

Les Commandes :

A Assemblage :

A (/ op1) (/ op2) ... (/ opn)

Cette commande effectue l'assemblage du buffer texte avec options facultatives.

NL : Le listing n'est pas affiché sur l'écran. Seules les erreurs apparaissent sur l'écran.

NO : Pas de sauvegarde sur disque.

NS : La table des symboles n'est pas imprimée.

LP : Edition totale sur imprimante.

WE : Arrêt de l'assemblage sur chaque erreur.
'ENTER' pour continuer.

Sans l'option NO, l'EDASS demande le nom du fichier sous lequel doit être sauvegardé le code objet. Vérifie ensuite l'existence éventuelle du fichier sur les disquettes et vous propose de le créer ou d'utiliser l'ancien.

Exemple : A / NO / WE Assemble avec arrêt sur erreur sans sauvegarde sur disque.

A

TOTO/CMD

B Retour au DOS

B 'ENTER' retourne au DOS par réinitialisation.

D Destruction de lignes

D aaaa (: bbbb)

détruit la ligne aaaa ou toutes les lignes entre aaaa et bbbb dans le buffer de texte.

Exemple : D. détruit la ligne courante.
D 100 détruit la ligne 100.
D 100 : 200 détruit les lignes de la
100 à la 200.
D # : X vide le buffer de texte.

E Editeur de lignes

E aaaa

Edite la ligne aaaa et permet sa modification éventuelle.

Sous commande de l'éditeur de ligne

- A reprise de l'édition
- (n) C changement de (n) caractères.
- (n) D destruction de (n) caractères.
- E fin d'édition avec acceptation des modifications.
- H destruction de la fin de ligne + insertion.
- I insertion d'une chaîne.
- L liste la ligne et reprend l'édition en début de ligne.
- Q fin d'édition sans modification.
- X curseur en fin de ligne + insertion.
- 'ENTER' pour terminer l'édition avec modification et affichage fin de ligne.

F Recherche

F (chaîne)

Cette commande effectue la recherche d'une chaîne dans le buffer de texte à compter de la ligne courante plus une, jusqu'en fin de texte.

Pour commencer au début, effectuez un P # , puis F (chaîne).

La chaîne ne peut dépasser 16 caractères.

Si trouvée, la ligne contenant la chaîne est affichée.

F seul recherche la dernière chaîne recherchée et gardée en mémoire.

Exemple : F 7000 H

H Edition imprimante

H aaaa (: bbbb)

liste une ligne aaaa, ou un groupe de lignes aaaa à bbbb sur imprimante. Le pointeur ligne courante est remis à jour sur la dernière ligne imprimée, elle fonctionne sur l'imprimante comme P sur l'écran.

Si l'imprimante n'est pas prête ou n'est pas branchée, elle affiche le message 'IMPRIMANTE INDISPONIBLE'. Un BREAK ou une intervention sur l'imprimante débloquent cette situation.

Exemple H # : * listing total buffer source.

I Insertion

I aaaa (, inc)

Cette commande permet de lancer l'éditeur et d'effectuer de l'insertion de ligne et de redémarrer une saisie avec un incrément de inc (l'ancien ou 10 par défaut). C'est le mode normal de saisie dont on sort avec un BREAK ou lorsque le Buffer est plein.

Exemple : I100, 5 lance l'insertion en 100 avec un incrément de 5.

LD Chargement de source

LD = Carfich

Cette commande permet de charger un fichier texte (source) depuis le disque et de le placer dans le buffer texte, en le mélangeant ou non aux sources éventuellement déjà en place.

Il suffit de répondre correctement aux questions posées.

Exemple : LD = BORD / SRC

N Renumérotation

N (aaaa (, inc))

Cette commande permet la renumérotation du texte en buffer, la première prenant le n° aaaa (100 par défaut) et les suivantes avec l'incrément spécifié (inc) à défaut avec l'inc en service précédemment (à défaut 10).



Exemple : N 10, 5

Renumérotation depuis 10, 15, 20, 25...

P Listing écran

P (aaaa (: bbbb)

Affiche à l'écran la ligne aaaa (à défaut 15 lignes avec la ligne courante en haut qui devient celle du bas pour une nouvelle page) ou un groupe de ligne aaaa à bbbb. Le pointeur de ligne courante prend la valeur de la dernière ligne affichée.

Les flèches haute  et basse  permettent de remonter pas à pas dans le listing (ligne par ligne).

*P #: * du debut à la fin*

Exemple : P # Affiche la 1ère ligne.

P Affiche les 15 premières lignes.

P Affiche les 15 suivantes.

P 10:200 de la ligne 10 à 200

R Remplacement

R (aaaa (, Inc)

Cette commande permet de remplacer une ligne ou un groupe de ligne ou de passer en insertion avec écrasement.

Si la ligne existe, elle est remplacée, sinon elle est insérée.

Inc suit les mêmes lois que celui de N ou I

Le pointeur est remis à jour sur le dernier n° de ligne introduit.

Exemple : R. remplace la ligne courante.

R 80, 10 remplace les lignes à partir de 80 avec un increment de 10.

T Impression texte imprimante

T (aaaa (: bbbb))

Cette commande est identique à H mais seul le texte est imprimé (sans les numéros de ligne).

WD Sauvegarde de source sur disque.

WD = Carfich / SRC *(SRC = SOURCE)*

Cette commande permet la sauvegarde sur disque du fichier source.

Une recherche s'effectue pour retrouver éventuellement la trace de ce fichier.

L'éditeur demande s'il faut créer ou éventuellement écraser l'ancien.

Après vos réponses, la sauvegarde s'effectue et l'on revient à l'éditeur.

Le texte est toujours intacte.

6.2 - CRECHAIN TSTCHAIN / JCL

Le programme BASIC CRECHAIN et le fichier TSTCHAIN / JCL ont été créés pour vous permettre de mieux approcher les fichiers chaînage, et de construire facilement vos propres séquences.

Il vous faut avoir bien assimilé les fonctions CHAIN (DO), CHNON (section 2.7 et 2.8, et section 3.2) avant d'aborder ces utilitaires.

Puis après avoir pris connaissance du fichier TSTCHAIN / JCL dans le paragraphe suivant, le faire s'exécuter par la commande DOS, DO TSTCHAIN ou CHAIN TSTCHAIN.

Vous constaterez alors les énormes possibilités de ces fonctions.

Ensuite charger le programme BASIC CRECHAIN, par la commande DOS : BASIC, RUN "CRECHAIN" (ou la touche F14) et suivre les instructions du paragraphe 6.2.2 et les menus du logiciel.

6.2.1 - TSTCHAIN / JCL : Un fichier chaînage.

```
1 : / . / 2 CE MESSAGE NE SERA PAS AFFICHE.  
2 : / . / 3 CE MESSAGE SERA AFFICHE.  
3 : FREE  
4 : BASIC  
5 : CMD "S = DIR Ø"  
6 : / . / 1 TEST DE LA COMMANDE PAUSE  
7 : BASIC  
8 : 1Ø PRINT "PAS MAL, HEIN ?"  
9 : 2Ø ? "IL FAUT LE FAIRE !"  
10 : 3Ø PAUSE 5ØØ  
11 : RUN  
12 : CMD "S  
13 : CHAIN, TSTCHAIN / JCL : Ø SECTION 2  
14 : / . / Ø SECTION 1  
15 : FREE  
16 : / . / 3 TESTS TERMINEES  
17 : / . / Ø SECTION 2  
18 : DIR Ø  
19 : / . / 4 SECTION 1
```

```
graph TD
    L13[13 : CHAIN, TSTCHAIN / JCL : Ø SECTION 2] --> L14[14 : / . / Ø SECTION 1]
    L17[17 : / . / Ø SECTION 2] --> L14
```

Si vous croyez avoir tout compris, essayez 'DO TSTCHAIN'

6.2.2 - CRECHAIN

Après son chargement, le menu principal vous offre ses instructions :

CREATION / EDITION DE FICHIER CHAINAGE

INSTRUCTIONS GENERALES

EFFACEMENT DU TEXTE EN MEMOIRE.	1
CHARGEMENT D'UN FICHIER DEPUIS LE DISQUE	2
SAUVEGARDE D'UN FICHIER SUR LE DISQUE	3
EDITION / CREATION DU TEXTE EN MEMOIRE.	4
RETOUR AU NEWDOS READY.	5

COMMANDE

- 1 Vide le buffer en mémoire, vous annonce si votre manœuvre est inutile et revient au menu général
- 2 Vérifie que le buffer est vide, vous demande s'il faut détruire le texte existant : si non, retour au menu général, si oui, vous demande les carfich du fichier à charger (/ JCL par défaut d'extension).

Ø affiche les fichiers / JCL du lecteur Ø.

1 affiche les fichiers / JCL du lecteur 1.

et reitère sa question.

'ENTER' retourne au menu général, sinon on charge le fichier dans le buffer en décomptant le nombre de lignes présentes, avec retour au menu général.

- 3 Permet la sauvegarde du fichier en mémoire avec les mêmes avantages (DIRECTORY) que l'option 2 (ajoute l'extension / JCL par défaut).
- 4 Vous place en mode Editeur de lignes avec son propre menu
- 5 Autorise le retour au NEWDOS READY. Si vous aviez apporté une modification au texte sans avoir effectué de sauvegarde, la question vous sera posée : faut-il réellement quitter sans sauvegarder ?

COMMANDES D'ÉDITION

MM, NN, TT sont des numéros de ligne et peuvent être remplacés par :

; = première ligne
/ = dernière ligne.
Ø = devant la 1ère ligne, si possible.

P	Listing des lignes sur imprimante.
L MM	Listing des lignes à partir de MM
;	Listing page suivante
R MM	Remplacement de la ligne MM
D MM NN	Destruction des lignes MM à NN
I MM	Insertion de lignes après la ligne MM
X MM	Ajouter à la ligne MM
C MM NN TT	Insertion d'une recopie des lignes MM à NN après la ligne TT
M MM NN TT	Destruction des lignes MM à NN et insertion après la ligne TT
Q	Retour aux instructions générales.

Commande ?

Une édition nouvelle se lance par un IØ

Toutes les instructions nécessaires à l'élaboration d'un fichier sont claires et faciles à exploiter. Reportez-vous à la section 3.2 et profitez de l'exemple pour vous faire la main.

Bon chañnage !

6.3 - FONC / CMD BASIC / FON EDASS / FON

Le clavier du GUEPARD et son 'DRIVER' permettent de gérer 15 touches de fonctions directes, 27 touches d'aides à la programmation indirecte (ESC + à A - Z) et une tabulation d'espace.

L'utilitaire FONC / CMD permet de sauvegarder l'état du clavier, de le transformer en chargeant un nouveau fichier, de lister l'état des touches sur ECRAN ou sur IMPRIMANTE.

Il se charge depuis le DOS en frappant FONC 'ENTER' ou F15 (disquette SYSTEME MAITRE) et affiche son menu :

UTILITAIRE TOUCHES DE FONCTION

Copyright HBN COMPUTEUR & RINCENT GERARD 1984

- 1 : Sauvegarde
- 2 : Chargement
- 3 : Listing
- 4 : Retour au DOS

Commande ?

- 1 assure la sauvegarde de l'état du clavier reprogrammé par vos soins en offrant 2 possibilités.
 - en FICHIER DISQUE normal pour être rappelé à un autre moment où son utilisation se revelerait fructueuse.
 - en FICHIER SYSTEME ce qui aurait pour effet de le placer dans le SYS Ø / SYS assurant ainsi son chargement automatique à l'initialisation de cette disquette (lecteur Ø).
- 2 Assure le chargement d'un fichier dont vous aurez à préciser le carfich.
- 3 Assure le listing avec le choix ECRAN ou IMPRIMANTE.
- 4 Assure le retour au NEWDOS READY.

Sur la disquette SYSTEME sont fournis deux fichiers fonctions à usage différent. BASIC / FON et EDASS / FON.

Comme leur nom l'indique, leur vocation est très nette, sans avoir la prétention d'être parfaits, ils vous donnent une excellente idée des possibilités de ce clavier en aide à la programmation ou à la gestion rapide.

7 - LES MYSTERES DU DIRECTORY

Le DIRECTORY d'une disquette (DIR / SYS) est le centre de renseignements indispensables à la gestion du support magnétique.

Il doit toujours être présent sur une disquette, le module FORMAT s'occupe de sa création, les autres modules d'accès disque de sa mise à jour.

Il démarre toujours au début d'un Lump, sa position étant déterminée par le paramètre DDSL du PDRIVE.

Sa taille varie de 2 à 6 granules en fonction du paramètre DDGA du PDRIVE.

L'utilisation de DEBUG en format 64 colonnes doit vous permettre de mieux suivre ce chapitre en faisant venir en RAM les différents secteurs du Directory pour mieux suivre nos commentaires.

'123' vous entrez sous DEBUG

S pour passer en mode page.

LØ, 17Ø et vous êtes devant la 1ère page du DIRECTORY.

7.1 - QUELQUES DEFINITIONS

La granule : c'est l'unité élémentaire d'allocation disque. Que votre fichier fasse 10 octets ou 1156 octets, le SYSTEME attribue l'espace disque 5 secteurs par 5 secteurs et 5 secteurs = 1 granule = 1280 octets = 1,25.KO.

Le Lump : c'est l'unité de gestion élémentaire de granules sur le disque (voir 7.2 GAT). Un Directory possède 192 Lump (octets de la GAT à vocation de gestion de granules).

FDE : voir section 7.4

FPDE : voir section 7.5

FXDE : voir section 7.6

FN : FCB Next field - voir 7.6. pointe le caractère courant dans le buffer FCB.

Hash code : Code sur un octet des carfich's d'un fichier dans la table HIT (issu d'un savant calcul binaire)

7.2 - LA TABLE D'ALLOCATION DE GRANULES (GAT)

Le premier secteur du Directory est la GAT.

Chacun des octets des 12 premières lignes est un LUMP qui peut lui-même gérer de 2 à 8 granules (selon le paramètre GPL du PDRIVE).

En principe, dans le format compatible, seul les 6 premières lignes sont utilisées. Mais avec les types de lecteurs actuellement en service, et à haute capacité, l'ensemble de la table est utilisée. Chaque octet peut donc gérer jusqu'à 8 granules selon le principe suivant : la gestion de la 1ère granule est sur le bit Ø, la 2ème sur le bit 1, ... la 8ème sur le bit 7, s'il y a lieu : bit à Ø = granule libre, bit à 1 = granule prise (ou inexistante) donc non disponible.

Exemple : avec GPL = 2 (1 Lump gérant 2 granules)

octet à FF = 2 granules occupées

FE = granule basse libre.

FD = granule haute libre.

FC = 2 granules libres.

La 13ème ligne contient les mots de passe en CE H - CF H

La 14ème ligne contient le nom de la disquette D0 H à D7 H

et la date de création D8 H à DF H

Les lignes 15 et 16 contiennent le contenu de la commande AUTO (voir section 2.3), terminé par un ØD H (32 caractères maxi). Si l'ØD H est au début, il n'y a pas de commande AUTO.

7.3 - LA TABLE D'INDEX CODEE (HIT)

La page suivante (2è secteur du Directory) est la HIT (hash Code Index Table).

Elle sert d'index de recherche pour tout le reste du Directory. Chacun des octets de cette table pointe vers une zone de 32 octets où sont rangée les carfichs d'un fichier, le FDE.

La ligne sur laquelle se trouve l'octet de la HIT pointe sur la même ligne des pages suivantes (FDE) et la place de l'octet sur cette ligne la nième page : ainsi le 1er octet pointe sur la page suivante (3è secteur = 1er secteur FDE), le 2è sur la 2è page et ainsi de suite jusqu'à la limite du Directory qui peut comprendre de 8 pages FDE (si DDGA = 2) à 28 (si DDGA = 6), ce qui à 8 emplacements de 32 octets = FDE donne un maximum de 224 FDE.

L'octet est à Ø si le FDE est libre ou inexistant.

L'octet contient le hash code du nom + extension du fichier.

C'est ainsi que se fait les recherches de fichiers sur disquette.

- Conversion Carfich en hash code.
- Recherche de l'octet dans la HIT
- Si trouvé, saut vers le FDE, analyse du FPDE pour vérification du Carfich.
- Sinon fichier non trouvé (inexistant dans ce Directory).

7.4 - LES SECTEURS FDE

Un FDE (File Directory Entry), une entrée directory pour un fichier se compose de 2 lignes de 16 octets, donc de 32 cases pouvant contenir un FPDE.

Le 1er octet d'un FDE en décrit le contenu.

- bit 4 : Ø = FDE libre (1 = occupé)
- bit 7 : Ø = FPDE dans FDE (1 = FXDE dans FDE).

1 secteur FDE contient 8 FDE possibles.

1 FDE peut donc contenir un FPDE ou FXDE ? !

qu'est-ce ?

7.5 - LE FPDE

FPDE : File Primary Directory Entry.

C'est la première série de renseignements sur un fichier (et éventuellement la seule s'il est court). Il occupe les 32 octets d'un FDE de Directory.

- 1er Octet bit 7 : \emptyset = FPDE (1 = FXDE)
bit 6 : 1 = SYSTEME (\emptyset = USER)
bit 5 : non utilisé
bit 4 : 1 = occupé (\emptyset = FDE libre)
bit 3 : 1 = invisible.
bit 2 à \emptyset : = Code de protection (niveau d'accès).
- 2e Octet bit 7 : \emptyset = Allocation suppl. autorisée ASE = Y (voir CREATE).
bit 6 : \emptyset = de allocation autorisée ASC = Y
bit 5 : 1 = UPDATE (voir DIR)
bit 4 à \emptyset : non utilisés.

3e Octet NON UTILISE

4e Octet Octet de poids faible de l'EOF (valeur relative de l'EOF dans le dernier secteur du fichier).

5e Octet indique la LRL (longueur d'enregistrement) (\emptyset = 256 Octets) A l'OPEN, ce n'est pas cette valeur qui est transférée dans le FCB, mais celle indiquée dans l'OPEN.

6e au 13e Octet : NOM du fichier comblé par des espaces à droite

14e au 16e Octet : EXTENSION du fichier comblé par des espaces à droite

17e – 18e Octet : Mot de passe (en code) de mise à jour

19e – 20e Octet : Mot de passe (en code) d'accès

21e Octet : Octet moyen de l'EOF

22e Octet : Octet de poids fort de l'EOF

23e au 30e Octet : Groupes de 2 octets pointant sur la position physique sur la disquette (Ext.)

- 1er octet : FF annonce la fin des Ext. du fichier
FE annonce que le prochain octet contient le DEC du prochain FXDE
FD à $\emptyset\emptyset$, indique le lump contenant le fichier

- 2e octet : si le 1er octet est inférieur à FE, les bits 7, 6, 5, indiquent la granule relative (\emptyset à 7) dans le lump désigné
Les bits 4, 3, 2, 1, \emptyset indiquent le nombre de granules – une, à compter de la 1ère.

31e – 32e Octet : Un Ext dont le 1er octet est forcément FF ou FE (annonce la fin du FPDE ou le FXDE d'extension)

7.6 - Le FXDE

File Extend Directory Entry. Lorsqu'un FDE ne suffit pas à contenir tous les EXT (4) d'un FPDE, le FXDE constitue une extension du FPDE

le 1er octet : Les bits 7 et 4 sont à 1 indiquant un FXDE occupé, les autres sont à 0

Le 2^e octet : contient le DEC du FPDE ou du FXDE précédent

Les 3^e à 22^e octets : sont à 0

Les 23^e à 32^e octet : sont codés comme un FPDE

7.7 - LE FCB

File control block. : bloc de contrôle fichier (également comme sous DCB ou IOB)

A chaque accès fichier du disque (lecture ou écriture), un lien est créé entre le fichier et le programme utilisateur

Ce lien, le FCB, est créé en RAM par l'OPEN et détruit par le CLOSE.

Durant l'utilisation du fichier, ce groupe de 32 octets est remis à jour (éventuellement). Attention, si l'EOF doit changer pour un fichier, seul le FCB est à jour. Le FDE du disque ne le sera que sur un CLOSE ou un PUT option &&.

Le FCB possède le format suivant :

1er octet	: bit 7	: 1 = FCB ouvert 0 = FCB fermé
	bit 6-2	: non utilisés
	bit 1	: 1 = le FN et l'EOF sont en RBA par rapport à la disquette
		0 = le FN et l'EOF sont en RBA fichier
	bit 0	: 1 = secteurs protégés à la lecture (idem le Directory)
		0 = secteurs normaux

(des CALL 0013 et 001B H doivent trouver les bits 1 et 0 à 0)

2 ^e octet	: bit 7	: 1 = la LRL est différente de 256 et la gestion secteurs est à la charge du DOS. FN s'incrémente caractère par caractère
		0 = la LRL est de 256. FN s'incrémente par 256

bit 6	:	Ø l'EOF est égal à FN après chaque écriture réussie (séquentiel par exemple, option "0")
	1	l'EOF est égal à FN, s'il dépasse l'EOF, après chaque écriture réussie
bit 5	:	Ø le Buffer FCB contient le secteur courant
	1	le Buffer FCB doit être mis à jour par une lecture
bit 4	:	Ø le Buffer FCB n'a pas été modifié ou a été sauvegardé sur disque
	1	Le Buffer FCB a été modifié et n'a pas encore été sauvegardé sur disque
		Durant le CLOSE, ce bit impose la sauvegarde sur disque avant la fermeture du FCB.
Bit 3	:	1 indique un FCB au format
		NEWDOS 80.20 sur les 18è — 32è octets
Bit 2—Ø	:	Code de protection d'accès
3è octet	:	bit 7—5 : idem FPDE 2è octet
	:	bit 4—Ø non utilisés
4è — 5è octet	:	Adresse en RAM du Buffer FCB . Buffer de 256 octets par lequel transite les données disques
6è octet	:	L'octet de poids faible du FN (position relative dans le secteur)
7è octet	:	Le n° du drive contenant la disquette
8è octet	:	Le DEC du FPDE du fichier
9è octet	:	L'octet de poids faible de l'EOF (position relative dans le secteur)
10è octet	:	la longueur d'enregistrement (LRL)
11è octet	:	l'octet de poids moyen du FN
12è octet	:	l'octet de poids fort du FN
		Le FN est le RBA de l'octet courant dans le FCB
13è octet	:	l'octet de poids moyen de l'EOF
14è octet	:	l'octet de poids fort de l'EOF
15è — 22è octet	:	idem les 23è—30è octets du FPDE
23è — 24è octet	:	ces octets pointent sur la granule relative dans le FXDE courant s'il y a lieu (ØFFFF H sinon)
25è — 32è octet	:	idem les 23è—30è octets du FXDE s'il y a lieu

Il est évident que ce chapitre n'est pas à la portée du débutant, et que sa lecture nécessite une solide connaissance du Z80, du langage machine, des routines d'entrée/sortie disques. Mais, il permet aux amateurs éclairés de gérer leurs propres routines d'accès disques ou de réparer leur directory défaillant.

APPENDICE A : GLOSSAIRE TECHNIQUE

Alphabétique :	Caractère compris entre A - Z et a - z
Alphanumérique :	Caractère compris entre A - Z, a - z et Ø - 9
AppelDOS :	voir section 3.3
bit :	composant élémentaire de la donnée : état d'une ligne de donnée (parmi les 8 du Z80) peut prendre la valeur Ø ou 1.
BOOT/SYS :	Un des deux fichiers de contrôle obligatoires sur chaque disquette (avec DIR / SYS). Mis en place par format.
Buffer :	espace de mémoire centrale servant de zone intermédiaire de stockage (tampon) entre les périphériques et le logiciel d'exploitation : Buffer, clavier, buffer disque ...
Chaînage :	procédé de substitution d'entrées au clavier par des caractères issus d'un fichier séquentiel sur disque (voir section 3.2)
Caractère :	(ou octet) : donnée élémentaire regroupant les états des 8 lignes du Z80 (bits), elle peut prendre 256 valeurs différentes ($2 \uparrow 8$), généralement exprimées en décimal (Ø - 255) ou en hexa (ØØ M - FF H) (base 16).
Carfich :	caractéristiques d'identification d'un fichier sur disque comprenant le nom, l'extension s'il y a lieu, le mot de passe s'ils sont validés, le n° du lecteur (facultatif) (voir manuel NEWDOS section 2, page 2.1).
DEC :	Directory entry code, un octet codé permettant l'accès rapide à un FDE donné dans le directory Format: rrrsssss sssss + 2 = secteur relatif du Directory rrr * 32 = octet relatif d'entrée FDE dans ce secteur
DIR/SYS :	L'autre fichier de contrôle indispensable sur chaque disquette contenant le directory (voir section 7) mis en place par FORMAT.
DOS :	Disk Operating System - Système d'exploitation des disques : désigne l'ensemble du logiciel assurant l'interface entre l'électronique et les progiciels, les langages, et l'utilisateur. Le NEWDOS 80.2.Ø est le DOS sous lequel travaille le SBASIC. Le CP/M3 est un autre DOS.
DRIVE :	mot anglais pour désigner un lecteur de disquette. Couramment employé en micro-informatique, DRIVE Ø = lecteur Ø.

EOF :	valeur de l'octet relatif (par rapport au 1er) marquant la fin d'enregistrement d'un fichier (End of file)
EOL :	octet de valeur 0D H marquant la fin de la ligne de caractère (End of line).
EOM :	End of message : 03 H. marque la fin d'un message DOS sans être lui-même affiché.
FCB :	voir section 7.7
FDE :	voir section 7.4
FPDE :	voir section 7.5
FXDE :	voir section 7.6
GAT :	voir section 7.2
Granule :	voir section 7.1
Hash Code :	voir section 7.1
HEXAdécimal :	système de notation numérique plus employé en informatique que le décimal. Sa base 16 permet de condenser la notation sur 2 caractères.

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
bit 0 :	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1
1 :	0	0	1	1	0	0	1	1	0	0	1	0	1	1	1	1	1
2 :	0	0	0	0	1	1	1	1	0	0	0	1	0	0	1	1	1
3 :	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	1	1

255 = FF H

HIMEM :	Le DOS gère l'espace mémoire présent sur les cartes (d'origine 64 kilo octets = 64 x 1024 caractères). On peut lui imposer d'en laisser une partie hors de son autorité en haut de mémoire (pour la préserver d'être recouverte si elle contient les données importantes pour l'utilisateur). Cette frontière haute s'appelle le HIMEM. Une fonction DOS permet de la fixer. A défaut le HIMEM est au maximum disponible.
HIT :	voir section 7.3
LRL :	longueur logique d'enregistrement par les fichiers non BASIC $0 < LRL \leq 255$ et $0 = 256$ (en BASIC $LRL = len$)
Lump :	voir section 7.1
Pistes :	(40 ou 80) surfaces concentriques sur le disque, divisées en secteur, où se place la tête de lecture.
RBA :	voir manuel BASIC. Section 9.11.3
Secteur :	partie d'une piste disque contenant 256 caractères en NEWDOS.

APPENDICE B : MESSAGES D'ERREUR COMMENTEES

Code message

Ø PAS D'ERREUR

1 DONNEES FICHIER INCORRECTES

Les données (DATA) en programme du fichier ne sont pas conformes à l'utilisation pour le DOS (fichier chaînage non séquentiel par exemple)

2 PISTE NON TROUVEE A LA LECTURE

Le contrôleur de disque n'a pas décodé la piste souhaitée à l'endroit où se trouve la tête. erreur souvent fatale (pour la disquette)

3 DONNEE PERDUE A LA LECTURE

Le contrôleur n'a pas pu retransmettre correctement la donnée au DOS

4 ERREUR DE PARITE A LA LECTURE

Le contrôleur ne peut accepter les données d'un secteur, l'octet de contrôle (CHECKSUM) a détecté une erreur de transmission.

On peut retenter l'essai, en général, la disquette est malade.

5 SECTEUR NON TROUVE A LA LECTURE

Le contrôleur de disque n'a pas décodé le secteur souhaité, erreur souvent fatale (pour la disquette)

6 LECTURE IMPOSSIBLE (SECTEUR VERROUILLE)

Tentative de lecture d'un secteur verrouillé dans la GAT

7 LECTURE IMPOSSIBLE (SECTEUR SYSTEME)

Tentative de lecture d'un secteur système protégé

8 PERIPH. INDISPONIBLE

La lecture de disque est sans disquette, ou elle est mal introduite, le loquet n'est pas fermé. . .

9 CODE D'ERREUR INDEFINI

Tout code d'erreur généré non classé aboutira ici.

10 PISTE NON TROUVEE A L'ECRITURE

Voir erreur 2 pour l'écriture

11 DONNEE PERDUE A L'ECRITURE

Voir erreur 3 pour l'écriture

12 ERREUR DE PARITE A L'ECRITURE

Voir erreur 4 pour l'écriture

13 SECTEUR NON TROUVE A L'ECRITURE

Voir erreur 5 pour l'écriture

14 ERREUR D'ECRITURE DRIVE

15 DISQUETTE PROTEGEE CONTRE L'ECRITURE

Le cran de protection de la disquette est masqué par un collant ou un obstacle imprévu. Vérifiez ce point précis et la disquette dans le bon sens.

16 PERIPH. INDISPONIBLE

Voir erreur 8

17 ERREUR DE LECTURE DIRECTORY

Vérifiez vos PDRIVE'S.. La compatibilité de vos disquettes. Elle peut être abimée

18 ERREUR D'ECRITURE DIRECTORY

19 NOM DE FICHIER INCORRECT

Première lettre A - Z , 8 caractères_ Alphanumériques après le 1er.

20 NUMERO DE PISTE TROP ELEVEE

demande d'accès piste au delà de celle autorisée au PDRIVE (TC)

21 FONCTION INCORRECTE SOUS APPELDOS

24 FICHIER INEXISTANT

Le fichier demandé n'est pas dans le Directory de la disquette

25 ACCES FICHIER REFUSE

Les mots de passe sont valides et le mot de passe annoncé est illégal ou manquant. L'accès est refusé jusqu'à ce qu'il soit bon.

26 DIRECTORY SATURE

Tous les FDE sont pris. Il n'y a plus de place pour créer un fichier. Voir une autre disquette, détruire des fichiers inutiles ou recopier CBF en augmentant DDGA.

27 DISQUETTE SATUREE

Plus aucune granule de libre. Il faut changer de disquette ou détruire d'autres fichiers.

28 EOF RENCONTRE

Voir fichiers en BASIC

29 EOF DEPASSE

Voir fichiers en BASIC

30 DIRECTORY SATURE. EXTENSION IMPOSSIBLE

Tous les FDE sont pris. Il n'y a plus de place pour étendre un fichier et créer un FXDE (voir erreur 26)

31 PROGRAMME NON TROUVE

Le programme appelé n'existe pas dans le directory

32 NUMERO DE DRIVE INCORRECT

Le lecteur concerné n'est pas en ligne (option du SYSTEME) ou dépasse 3 (autorisés 0 à 3)

33 MEMOIRE PERIPH. INDISPONIBLE

34 FORMAT DU FICHIER INCORRECT

35 ERREUR DE MEMOIRE

37 ACCES IMPOSSIBLE (FICHIER PROTEGE)

38 FICHIER NON OUVERT

39 INIT DU SYSTEME INCORRECTE

40 COMPTAGE PISTE INCORRECT

41 NUMERO DE FICHIER LOGIQUE INCORRECT

42 FONCTION DOS INCORRECTE

43 FONCTION INCORRECTE SOUS CHAINAGE

44 DONNEES DIRECTORY INCORRECTES

45 DONNEES FCB INCORRECTES

Voir section 7.7

46 PROGRAMME SYSTEME NON TROUVE

Appel à une routine DOS dont le fichier SYSTEME n'est pas sur la disquette

47 PARAMETRES INCORRECTS

Une erreur de syntaxe s'est glissée dans les paramètres de votre commande DOS

48 PARAMETRES DU FICHIER INCORRECTS

Idem 47 pour l'accès à un fichier

49 DISQUETTE INCOMPATIBLE (ENREGISTREMENT)

Format d'enregistrement de disquette illisible

50 ERREUR DE LECTURE DU BOOT

Code Message

51 ERREUR DOS FATALE

52 FORMAT DE LA COMMANDE INCORRECT

53 FICHIER DEJA EXISTANT

54 COMMANDE TROP LONGUE

55 ACCES DISQUETTE REFUSEE

Les mots de passe sont validés et le mot de passe disquette
n'est pas le bon (pour une copie par exemple, Attrib. . .)

56 FONCTION MINIDOS INCORRECTE

Voir section 3.1

57 ARRET DEMANDE

58 ERREUR DE COMPARAISON DE DONNEES

59 MEMOIRE INSUFFISANTE

60 DRIVES OU DISQUETTES INCOMPATIBLES

Vérifiez vos PDRIVE'S
Pour une copie, utilisez CBF (format 6)

61 PARAMETRE ASE = N. EXTENSION FICHIER IMPOSSIBLE

62 EXTENSION FICHIER NON TROUVEE A LA LECTURE

NEWDOS/80 READY

COPY,0 1,,NFMT

LANCEMENT DE LA COPIE

LA DISQUETTE SOURCE EST AUSSI LA DISQUETTE SYSTEME (0/N)?N

PRESSEZ "ENTER" SI LA DISQUETTE SOURCE EST DANS LE DRIVE 0

PRESSEZ "ENTER" SI LA DISQUETTE DESTINATION EST DANS LE DRIVE 1

COPIE

TERMINE

PRESSEZ "ENTER" SI LA DISQUETTE SYSTEME EST DANS LE DRIVE 0

NEWDOS/80 READY

*Copy de disquette du drive 0 → 1
sans disquette système*