

Réalisez votre ordinateur individuel

LES PSEUDO-DISQUETTES LA DOUBLE DENSITÉ A 4 Mhz

C'est un article entièrement logiciel que nous vous proposons ce mois-ci, article dont le texte d'accompagnement va être assez court afin de laisser un maximum de place dans nos colonnes pour les listings des programmes que nous vous offrons.

Nous allons aborder trois sujets importants : le premier est consacré à l'utilisation des pages 2 et 3 de la mémoire comme pseudo-disquettes, au moyen d'un logiciel qu'a bien voulu nous communiquer un dynamique réalisateur du système, M. Danis, que nous remercions ici. Le deuxième sujet intéressera sans doute la majorité des possesseurs de disquettes car il va être consacré au fonctionnement en double densité à 4 MHz sans aucune intervention matérielle sur le système. Il est encore l'œuvre de l'un d'entre vous, puisque c'est un programme de M. Millaud que nous remercions aussi sincèrement pour sa communication. Enfin, dernier point abordé ce mois-ci, la description d'un certain nombre de logiciels commercialisés par la société Micropross que nombre d'entre vous connaissent déjà pour s'y être approvisionnés en composants.

Nous souhaitons ainsi répondre à une des critiques les plus souvent formulées dans les réponses au questionnaire reçues jusqu'à ce jour : le manque de logiciel.

Les pseudo-disquettes

Les pages 2 et 3 de la carte RAM dynamique 256 Ko, restées inutilisées jusqu'à ce jour peuvent trouver une application très intéressante en tant que pseudo-disquettes. Qu'est ce qu'une pseudo-disquette ? Tout simplement une portion de mémoire (les pages 2 et 3 ou 2 seulement dans ce cas) qui, grâce à un programme adéquat, se comporte comme une disquette souple classique. Bien sûr, le contenu de cette pseudo-disquette est perdu

lors de l'arrêt de l'appareil puisqu'elle est constituée par de la RAM, mais ce n'est pas bien grave car l'intérêt d'un tel système se manifeste ailleurs.

L'intérêt d'une pseudo-disquette est évident pour tous les possesseurs d'un seul lecteur qui peuvent ainsi travailler comme s'ils en avaient deux ; sous réserve bien sûr de sauvegarder le contenu de la pseudo-disquette sur une vraie disquette avant d'arrêter l'appareil. Mais l'intérêt d'un tel système ne se limite pas à cela et les possesseurs de deux lecteurs peuvent aussi exploiter avec succès une pseudo-dis-

quette ; en effet, celle-ci étant constituée par de la RAM, son temps d'accès est beaucoup plus court que pour des vraies disquettes, ce qui est très agréable lors des phases d'écriture de programmes en assembleur ou en Basic lorsque l'on est amené à charger très souvent des utilitaires du DOS. Il suffit de copier ceux-ci sur la pseudo-disquette pour les avoir sous la main quasi instantanément.

La mise en œuvre d'une pseudo-disquette nécessite un investissement très faible puisqu'il se limite à l'achat de 8 ou 16 boîtiers de RAM 64 K-bits pour équiper la page 2 ou les pages 2 et 3 de votre carte RAM dynamique. Il faut ensuite disposer du programme adéquat, mais son listing vous est donné ci-après grâce à l'amabilité de M. Danis, que nous tenons à nouveau à remercier pour sa communication.

Une telle adjonction vous permet de bénéficier de :

- 28 pistes de 8 secteurs de 256 octets ou 14 pistes de 16 secteurs de 256 octets, si vous équipez une page mémoire ;
- 56 pistes de 8 secteurs de 256 octets ou 28 pistes de 16 secteurs de 256 octets, si vous équipez deux pages mémoire.

Les deux configurations indiquées pour chaque cas correspondent à deux possibilités que vous offre le programme comme nous allons le voir ci-après. Commençons par le commencement, qui est la mise en place des RAM. Après vous être procuré les boîtiers adéquats et vous être muni du

« Haut-Parleur » n° 1683, équipez le bloc 2, ou le bloc 2 et le bloc 3 si vous voulez le plus grand pseudo-disque possible. Validez ensuite ce ou ces blocs en fermant les interrupteurs S2 pour le bloc 2 et S3 pour le bloc 3 (fig. 5, « Haut-Parleur » n° 1683, page 91).

Vérifiez alors soigneusement le fonctionnement de votre RAM dynamique ainsi équipée en chargeant le DOS et en faisant tourner quelques programmes pour vous assurer que tout fonctionne comme par le passé. Théoriquement, l'installation des blocs 2 et 3 ne doit pas perturber le bloc 1, mais il suffit que l'un des 74541 de commande des lignes d'adresses soit un peu faible pour que la capacité supérieure présentée sur ses sorties par l'adjonction des nouveaux boîtiers dégrade un signal qui, avec un bloc, restait correct. Une telle situation doit cependant rester exceptionnelle et doit pouvoir être résolue, si elle se produit, par un échange des 74541 de commande d'adresses des mémoires (les 74541 repérés 1, 2 et 3 sur la figure pré-citée).

Lorsque tout est correct, passez à l'édition du programme que nous avons baptisé PSEUDISK. Son listing d'édition vous est donné figure 1. Pourquoi le listing d'édition et non celui d'assemblage pourtant plus lisible ? Parce que son auteur a utilisé les possibilités d'assemblage conditionnel offertes par l'assembleur du DOS et que celles-ci ne peuvent apparaître sur le listing final. Cela ne doit vous poser aucun problème, il vous suffit en effet de frapper,


```
OPT PAG
TTL UTILISATION RAM EN PSEUDO DISQUES
PAG
```

```
* PROGRAMME ORIGINAL ECRIT PAR MR DANIS
* COMMENTAIRES REVISES
* LE 22 JUIN 1984
```

```
* C.TAVERNIER
* POUR LE HAUT PARLEUR
```

```
* PARAMETRES
```

```
* NB DE PAGES :
NBPAGE EQU 1 Mettre (1 ou 2)
```

```
* NB DE SECTEURS RESERVES POUR LE DIRECTORY EN PISTE 0 :
DIR EQU 4 Mettre (4 ou 12)
```

```
* N° DE LECTEUR ATTRIBUE PAR DEFAUT AU PSEUDO DISK :
DEF EQU 3 (1, 2 ou 3)
```

```
* CONSTANTES DU DOS
```

```
GETCHR EQU $CD15
PUTCHR EQU $CD18
PSTRNG EQU $CD1E
WARMMS EQU $CD03
DATE EQU $CC0E
GETHEX EQU $CD42
SYSFCB EQU $CB40
RPTERR EQU $CD3F
OUTDEC EQU $CD39
```

```
* EQUIVALENCES DIVERSES
IF NBPAGE=2
```

```
FISMAX EQU 2*$E0/SECMAX-1
ELSE
FISMAX EQU $E0/SECMAX-1
ENDIF
```

```
* SECMAX EQU DIR+4 Nb de secteurs max
TABLEC EQU $DE00
```

```
* ORG $C100
```

```
* DEBUT BRA DEB
FCB 13 Version 1.3
DEB JSR GETHEX Acquisition No de lecteur
LBCS ERREUR
TSTB No de lecteur attribue ?
BNE ACCORD Oui
CLRA
LDB #DEF
```

```
BRA CONTIN
ACCORD TFR X,D
CMPD #3
LBHI ERREUR
CONTIN STD BONLEC
LDX #MESG1 En-tete
JSR PSTRNG Ecriture
```

```
* TEST SI PREMIERE UTILISATION
* SINON DEMANDE SI FORMATAGE AUTORISE
* SI OUI FORMATE LA MEMOIRE SANS
* POSER DE QUESTION
```

```
LDD TABLEC+13
CMPD #LECT1
BNE PREPAS
LDX #MESG5
JSR PSTRNG
JSR GETCHR
ANDA #$F
CMPA #0
BNE FIN
LDX #MESG6
JSR PSTR
JSR GETCHR
ANDA #$F
CMPA #0
BNE FIN
BRA EFF
PREPAS STD SVECT+1
```

```
* CREATION SIR , FORMATAGE RAM
* INITIALISATION DES LECTEURS
```

```
*
EFF LDU #BUFSIR
LDX #100
LBSR MZERO
LDX #INISIR
LDU #BUFSIR+16
LDA #24
LBSR DEPL
LDX #DATE
LDU #BUFSIR+35
LDA #3
LBSR DEPL
LDX #MESG2
JSR PSTRNG
JSR FORMAT
LDX #MESG3
JSR PSTR
LDX #MESG4
JSR PSTRNG
LDX #BONLEC
CLRB Drapeau suppression zéros non significatifs
```

```
JSR OUTDEC
LDA #.
JSR PUTCHR
LDX #LECT1
STX TABLEC+13
LDX #TABLEC
LDU #DEPDIS
LDA #30
LBSR DEPL
FIN LDA #$0D
JSR PUTCHR
LDA #0A
JSR PUTCHR
JMP WARMS
```

```
*
ERREUR LDX #SYSFCB
LDA #15
STA 1,X
JSR RPTERR
JMP WARMS
```

```
*
* S/P D'AFFICHAGE CHAINE SANS PCRLF :
```

```
*
PSTR LDA 0,X+
CMPA #0A
BEQ RET
JSR PUTCHR
BRA PSTR
RET RTS
```

```
* CONSTANTES INITIALISATION SIR
```

```
INISIR FCC "PSEUDISK" Nom disque
FCB 0,0,0
FDB 1 No de volume
FCB 1,1 Début de chaine libre
FCB PISMAX,SECMAX Fin de chaine libre
FDB PISMAX*SECMAX Longueur chaine libre
FCB 0,0,0 DATE
FCB PISMAX No de pistes max
FCB SECMAX No de secteurs max
```

```
* MESSAGES :
```

```
MESG1 FCC "PSEUDISK Version 1.3"
FCB $0D,$0A
IF NBPAGE=2
FCC "PAGES 2 ET 3 ( 112 K DE RAM ) UTILISEES EN PSEUDO DISQUE"
ELSE
FCC "PAGE 2 ( 56 K DE RAM ) UTILISEE EN PSEUDO DISQUE"
ENDIF
FCB $04
MESG5 FCB $0D,$0A
FCC "***** ATTENTION *****"
```

```
FCB $0A,$0A,$0D
FCC "LE PSEUDO DISQUE A DEJA ETE FORMATE !"
FCB $0A,$0D
FCC "DESIREZ VOUS RECOMMENCER (O/N*) ? "
FCB $04
MESG6 FCB $0A,$0D
FCC "EN ETES VOUS SUR (O/N*) ? "
FCB $04
MESG2 FCB $0D,$0A
FCC "FORMATAGE MEMOIRE EN COURS"
FCB $0A,$0D,4
MESG3 FCB $0A,$0D
FCC "FORMATAGE TERMINE"
FCB $04
MESG4 IF DIR=4
IF NBPAGE=1
FCC "28 PISTES DE 8 SECTEURS"
ELSE
FCC "56 PISTES DE 8 SECTEURS"
ENDIF
ELSE
IF NBPAGE=1
FCC "14 PISTES DE 16 SECTEURS"
ELSE
FCC "28 PISTES DE 16 SECTEURS"
ENDIF
ENDIF
FCC " DISPONIBLES SOIENT : "
IF DIR=4
IF NBPAGE=1
FCC "216 SECTEURS"
ELSE
FCC "440 SECTEURS"
ENDIF
ELSE
IF NBPAGE=1
FCC "208 SECTEURS"
ELSE
FCC "432 SECTEURS"
ENDIF
ENDIF
FCC " LIBRES"
FCB $0D,$0A,$0A
FCC "LE PSEUDO DISQUE PORTE LE NUMERO DE LECTEUR : "
FCB $04
```

```
BUFSIR EQU *
```

```
*
ORG $EC00
```

```
* ATTENTION , NE PAS MODIFIER
* CETTE ORIGINE
```


LISTING SOURCE DE PSEUDISK

2-8-84

PAGE 5

```

* EQUIVALENCES DIVERSES
*
PILPEM EQU BUFFER+$$FF+$20 File système temporaire
PIACPU EQU $$EFA PIA CPU09
FLAG FCB $00 FLAG mode courant
      ($00=Mode disque, $$FF=Mode RAM)
BONLEC RMB 2 No de lecteur attribué au pseudo disque
JUMP EQU $7E
SVECT FCB JUMP Sauv. vect. "LECT" d'origine
RMB 2
*
* S/P DE RAZ ZONE MEMOIRE
* ENTREE : U=Adresse de début
          X=Longueur zone
*
MZERO CLR 0,U+
LEAX -1,X
BNE MZERO
RTS
*
* S/P DE TRANSFERT ZONE MEMOIRE
*
* ENTREE : X=Début zone à transférer
          U=Adresse du but
          A=Longueur zone à transférer
*
DEPL LDB 0,X+
STB 0,U+
DECA
BNE DEPL
RTS
*
* S/P DE FORMATAGE RAM
*

```

```

FORMAT PSHS U,Y
TFR S,Y
LDS #PILPEM
LBSR ARPAG2
LDU #F0000
LDX #E000
BSR MZERO
IF NBPAGE=2
LBSR ARPAG3
LDU #F0000
LDX #E000
BSR MZERO
ENDIF
JSR ARPAG2
LDD #F0002
BSR FORPG2
IF NBPAGE=2
LBSR ARPAG3
LDX #F0000
BSR FORPG3

```

LISTING SOURCE DE PSEUDISK

2-8-84

PAGE 6

```

ENDIF
CLRA
LDB #SECMAX
LBSR CALCUL
LDD #F0000
STD 0,U
LDA #PISMAX
LDB #SECMAX
LBSR CALCUL
LDD #F0000
STD 0,U
LBSR ARPAG1
LDD #F0003
LDX #BUFSIR
BRA ECRSEC
*
* S/P DE FORMATAGE PAGE :
*
FORPG2 LDX #F0000
FORPAG STD 0,X
LEAX #100,X
CMPX #DFFF
BHI FIN2
FORPG3 INCB
CMPB #SECMAX
BLS FORPAG
LDB #1
INCA
BRA FORPAG
FIN2 RTS
*
* S/P LECTEUR POUR PSEUDO DISQUE
*
* ECRITURE D'UN SECTEUR :
*

```

```

ECRIT PSHS U,Y
IFR S,Y

```

LISTING SOURCE DE PSEUDISK

2-8-84

PAGE 7

```

*
LECT PSHS U,Y
TFR S,Y
LDS #PILPEM
PSHS X
LBSR CALCUL
TFR U,X
LDU #BUFFER
LDA #F00
LBSR DEPL
LBSR ARPAG1
LDX #BUFFER
PULS U
LBSR DEPL
TFR Y,S
PULS U,Y
BON CLR8
RTS
*
* SELECTION LECTEUR SPECIFIE :
*
LECT1 LDB 3,X
CMPB BONLEC+1
BNE NBLECT
TST FLAG
BEQ RAMDEF
BRA FIN3
RAMDEF LDX #DEPRAM
PSHS U
LDU #TABLEC
LDA #30
LBSR DEPL
DEC FLAG
PULS U
FIN3 CLR8
RTS
*
NBLECT TST FLAG
BEQ LECCOUR
PSHS X,U
LDX #DEPDIS
LDU #TABLEC
LDA #30
LBSR DEPL
INC FLAG
PULS X,U
LECCOUR JMP SVECT
*
* INITIALISE LECTEUR (DEPART A FROID) :
*
INIT BSR INITAB
JMP TABLEC+21
*
* INITIALISE LECTEUR (DEPART CHAUD) :

```

LISTING SOURCE DE PSEUDISK

2-8-84

PAGE 8

```

*
CHAUD BSR INITAB
JMP TABLEC+24
*
INITAB LDX #DEPDIS
LDU #TABLEC
LDA #30
LBSR DEPL
CLR FLAG
RTS
*
* RECOUVRE VECTEUR LECTEUR RAM
*
DEPRAM JMP LECT
JMP ECRIT
JMP BON
JMP BON
JMP LECT1
JMP BON
JMP BON
JMP INIT
JMP CHAUD
JMP BON
*
* RECOUVRE VECTEURS LECTEUR DISQUE
*
DEPDIS RMB 30 Remplis quand PSEUDISK copie vecteurs
* Lecteur d'origine
*
* S/P DE CALCUL ADRESSE RAM D'UN SECTEUR :
*
* ENTREE : D=T/S Demandées
* SORTIE : U=Adresse RAM équivalente du pseudo disque
* Page arrêtée en fonction adresse
*
CALCUL DEC8

```



```

RTS
*
* S/P D'ARRET PAGE 3
*
ARPAG3 PSHS B
LDB *PIACPU5
ANDB #$FC
ORB #$03
STB *PIACPU5
PULS B
RTS
*
* S/P D'ARRET PAGE 2
*
ARPAG2 PSHS B
LDB *PIACPU5
ANDB #$FC
ORB #$02
STB *PIACPU5
PULS B
RTS
*
* S/P D'ARRET PAGE 1
*
ARPAG1 PSHS B
LDB *PIACPU5
ANDB #$FC
ORB #$01
STB *PIACPU5
PULS B
RTS

BUFFER EQU *

END DEBUT

```

Fig. 1. — Listing source de PSEUDISK

sous éditeur, le listing de la figure 1 tel qu'il est écrit dans ces pages. Seules trois informations sont à modifier en fonction de votre configuration et de ce que vous voulez faire.

Il faut d'abord définir le nombre de pages mémoire utilisables en pseudo-disquette en modifiant éventuellement le NBPAGE EQU 1 (valable si vous n'avez qu'une page, c'est-à-dire le bloc 2 seul équipé) en NBPAGE EQU 2 si vous disposez de 2 pages, c'est-à-dire des blocs 2 et 3 équipés.

Il faut ensuite définir le nombre de secteurs réservés au directory c'est-à-dire le nombre de secteurs qui vont servir de répertoire des noms de fichiers contenus sur la pseudo-disquette. Deux options vous sont offertes par le programme au niveau du DIR EQU 4 que vous pouvez transformer en DIR EQU 12. La première solution convient si vous voulez stocker assez peu de fichiers différents sur la pseudo-disquette et disposer d'un maximum de place pour ceux-ci. La deuxième solution convient si vous avez de nombreux fichiers différents à stocker mais elle vous laisse un peu moins de place. Nous vous conseillons de faire deux assemblages du programme ; l'un avec DIR mis à 4 et l'autre avec DIR mis à 12, et d'utiliser

ensuite l'un ou l'autre en fonction de vos besoins.

La dernière donnée à définir est le numéro de lecteur qui sera affecté au pseudo-disque par défaut. Cela se situe au niveau du DEF EQU 3 que vous remplacerez par un DEF EQU 1 si vous n'avez qu'un lecteur (qui est alors le numéro 0), ou par un DEF EQU 2 si vous avez deux lecteurs (qui portent alors les numéros 0 et 1).

Vous pouvez ensuite frapper le programme tel qu'il est publié en n'en modifiant aucune partie, sauf les commentaires éventuels si vous voulez gagner quelques minutes de frappe, mais ce n'est pas une bonne solution car cela conduit à des listings difficilement lisibles, surtout quelques mois après...

Pour ceux qui aiment le beau travail, appréciez l'assemblage conditionnel pour définir PISMAX en fonction de NBPAGE et surtout au niveau des messages que peut afficher le programme ; messages qui dépendent de NBPAGE et de DIR.

Veillez à ne pas modifier l'ORG \$ ECOO qui figure dans le programme ; en effet une partie de celui-ci doit résider en RAM commune de la carte CPU09 pour pouvoir avoir accès à toutes les pages sans problème.

Lorsque c'est fini, assemblez ce programme et donnez

lui un nom suivi d'une extension CMD pour pouvoir l'utiliser comme commande du DOS. Son essai est immédiat puisqu'il suffit de faire PSEUDISK (si tel est son nom) pour le lancer. Attention, aucun test n'est fait sur la présence de la RAM en pages 2 et 3 ; en conséquence PSEUDISK s'exécutera bien même s'il n'y pas de mémoire installée, mais il est évident que la pseudo-disquette ainsi formatée ne sera pas utilisable et que toute tentative d'aller la lire, par un DIR, par exemple, peut bloquer le système et nécessiter un RESET.

Deux autres points sont à signaler à propos de ce logiciel. Tout d'abord, la définition de numéro de lecteur faite par défaut grâce à ce que vous avez édité sur votre listing peut être modifiée en faisant PSEUDISK N, où N est le numéro qui sera alors donné au lecteur. Attention ! Si vous avez plusieurs lecteurs et que vous donniez au pseudo-disque le numéro d'un de vos vrais lecteurs, l'accès à ce dernier vous deviendra impossible, le pseudo-disque étant prioritaire. Cela peut être pratique ou non.

Le deuxième point important est que PSEUDISK réalise, à chaque appel, un formatage de la mémoire et détruit donc tout ce qu'elle pouvait contenir. Pour éviter des incidents fâcheux, un message de demande de confirmation apparaît lors du deuxième appel de PSEUDISK sans extinction du système entre les deux. Cela vous évite de détruire par inadvertance un pseudo-disque bien rempli.

Nous avons longuement essayé et utilisé ce programme et nous pouvons affirmer qu'il fonctionne sans problème ; un « bug » caché peut nous avoir échappé et, si tel était le cas, nous vous demandons de bien vouloir nous en informer.

La double densité à 4 MHz

Dans notre numéro 1698 de novembre 1983 nous avons publié les listings nécessaires pour faire fonctionner le DOS en double densité. Cette pos-

sibilité nécessitait cependant d'accélérer l'horloge du système en mettant sur la carte CPU09 un quartz à 5 MHz au lieu du 4 MHz initial. Si cela n'a pas posé de problème pour certains systèmes, d'autres ont refusé obstinément cette légère accélération ou, ce qui est pire, se sont mis à avoir un fonctionnement erratique en version accélérée. La solution à tous ces maux nous est offerte aujourd'hui par M. R. Millaud. Elle permet de faire fonctionner le DOS en double densité, sans aucun problème, et en conservant l'horloge initiale à 4 MHz. Cela présente plusieurs avantages dont le premier est d'être assuré que si votre système tourne sans problème en simple densité, il en sera de même en double puisque l'on ne change pas l'horloge. Le deuxième est que l'interface cassette est de nouveau utilisable, ce qui peut tout de même rendre service. Dernier point enfin, les moniteurs TAVBUG09 n'ont plus besoin d'être modifiés au niveau des constantes de programmation de vitesse de la liaison série de la carte CPU09.

Un lot fonctionnellement a été rendu possible grâce à l'étude très pointue des temps d'exécution des divers cas de lecture et écriture sur le disque en simple et double densité à laquelle s'est livré M. Millaud. Etude à l'issue de laquelle il a modifié les listings que nous avions publiés en novembre 1983 pour parvenir au fonctionnement annoncé.

Afin de ne pas faire des dizaines de renvois et des explications plus ou moins fumeuses, nous donnons ci-après l'intégralité des listings modifiés avec la procédure complète à suivre pour disposer d'un DOS double densité et, bien sûr, d'une commande FORMAT capable de formater les disques en simple et double densité également. Suivez donc scrupuleusement ce qui suit et vous obtiendrez satisfaction dans une petite demi-heure (ou moins si vous frappez vite... et bien !).

Si vous avez fait les modifications de novembre 83 et que votre système fonctionne parfaitement à 5 MHz, vous êtes

libre ou non de faire ce qui est indiqué ci-après. Par souci d'homogénéité et de compatibilité entre les systèmes, nous vous recommandons cependant de le faire vu le peu de temps nécessaire.

La première chose à faire est de modifier les « drivers » des disquettes, c'est-à-dire les programmes du DOS qui lisent et écrivent sur la disquette. Pour cela, éditez puis assemblez le programme dont le listing vous est donné figure 2. Respectez scrupuleusement les adresses et instructions qui y figurent, car ce programme va devoir se raccorder à tout ce qui existe déjà et il faut donc qu'il soit exactement conforme à ce que nous publions.

L'assemblage de ce programme va vous donner un fichier que nous appellerons DISK.BIN. Faites alors, sur une copie de votre disquette système (en cas de fausse manœuvre) :

- RENAME TAVDOS09.SYS, TAVDOS09.BIN
- APPEND TAVDOS09 . BIN . DISK . BIN . TAVDOS09 . SYS
- LINK TAVDOS09.SYS

Ces commandes sont, bien sûr, à compléter en fonction de vos numéros de lecteurs si vous travaillez avec plusieurs drives et que, par exemple, DISK soit sur celui de travail et TAVDOS09 sur le disque système.

Arrivé à ce stade, vous disposez d'un DOS capable de lire une disquette double densité et d'y écrire. Par contre, un tel DOS ne peut pas se charger à partir d'une disquette double densité et vous ne disposerez d'aucun moyen pour fabriquer des disquettes de ce type. Nous allons donc maintenant modifier le chargeur du DOS et la commande FORMAT.

Editez et assemblez le morceau de programme visible sur le listing de la figure 3. C'est un extrait du chargeur du DOS qui se trouve normalement sur la piste 0 et qui charge le DOS en mémoire suite à une commande X.

Lorsque ces deux opérations sont faites, vous disposez sur disquette d'un fichier que nous appellerons LOADER.BIN dans la suite de ces explications.

Editez et assemblez alors le programme visible figure 4 qui est un extrait de la commande FORMAT. Les modifications ponctuelles situées au début du listing permettent de débloquent des morceaux de code qui étaient déjà prévus dans la commande FORMAT que nous vous avons fournie pour la double densité, tandis que la fin du listing est la partie écriture des pistes adaptée par M. Millaud.

Une fois ces opérations réalisées, vous disposez sur disquette d'un fichier que nous appellerons MODIFORM.BIN dans la suite de ces explications. Vous allez alors effectuer les opérations suivantes :

- GET FORMAT.COMD qui charge votre commande FORMAT actuelle en mémoire et le chargeur du DOS puisque celui-ci fait partie intégrante de la commande FORMAT.
- GET LOADER.BIN qui place le morceau du chargeur modifié sur celui chargé en mémoire par l'opération précédente.
- SAVE.LOW LOAD.BIN, C100, C1F8 qui sauvegarde sur la disquette dans le fichier LOAD.BIN le chargeur du DOS modifié et capable de charger le DOS à partir d'une disquette simple ou double densité. Attention ! n'utilisez surtout pas SAVE pour faire cette opération, mais bien SAVE.LOW sans quoi vous détruiriez le chargeur en appelant SAVE.

Faites ensuite :

- GET FORMAT.COMD qui recharge FORMAT en mémoire (partiellement détruit par le SAVE.LOW précédent).
- GET MODIFORM.BIN qui charge les modifications de FORMAT.
- SAVE FORMATD.BIN, 0100, 07A4, 0100 qui sauvegarde FORMAT modifié mais sans le chargeur du DOS sur la disquette.
- APPEND FORMATD.BIN, LOAD.BIN, FORMATD.COMD qui met bout à bout FORMATD.BIN et le chargeur du DOS pour faire votre nouvelle commande FORMATD.

Cette nouvelle commande FORMATD fonctionne comme par le passé, mais pose une question de plus relative à la densité de la disquette à laquelle il suffit de répondre par

*MODIFICATION DES DRIVERS DES DISQUES FOUR
*FONCTIONNER EN DOUBLE DENSITE A 4 MHZ
*PROGRAMME DE MR R.MILLAUD
*MODIFIE PAR C.TAVERNIER POUR COMPATIBILITE
*AVEC EXTRAS DE MICROPROSS

*DEFINITION DES CONSTANTES

0002	DRQ	EQU	2
0001	BUSY	EQU	1
001C	RDMSK	EQU	\$1C
001B	VERMSK	EQU	\$1B
005C	WTMSK	EQU	\$5C
0000	NFINSK	EQU	0
5000	MISVIT	EQU	\$5000
E0B0	DRVREG	EQU	\$E0B0
E000	COMREG	EQU	\$E000
E001	TRKREG	EQU	COMREG+1
E002	SECREG	EQU	COMREG+2
E003	DATREG	EQU	COMREG+3
008C	RDCMNO	EQU	\$8C
008E	RDCMNI	EQU	\$8E
00AC	WTCMNO	EQU	\$AC
00AE	WTCMNI	EQU	\$AE
000B	RSCMND	EQU	\$0B
001B	SKCMND	EQU	\$1B
CC34	PRCNT	EQU	\$CC34

*TABLE DES FONCTIONS

DE00			ORG		#DE00
>DE00	7E	DE37	DREAD	JMP	READ
DE03	7E	DEEE	DWRITE	JMP	WRITE
DE06	7E	DF52	DVERFY	JMP	VERIFY
DE09	7E	DF78	RESTOR	JMP	RST
DE0C	7E	DF8B	DRIVE	JMP	DRV
DE0F	7E	DFB7	DCHECK	JMP	CHKRDY
DE12	7E	DFD4	DQUICK	JMP	QUICK
>DE15	7E	DE27	DINIT	JMP	INIT
>DE18	7E	DE31	DWARM	JMP	WARM
>DE1B	7E	DE96	DSEEK	JMP	SEEK

*DEFINITION DES RAM

DE1E	00		CURDRV	FCB	0
DE1F	0000	0000	DRVTRK	FDB	0,0
DE23	00		CURSID	FCB	0
DE24	00	00	DNSTY	FCB	0,0,0

*INITIALISATION

DE27	8E	DE1E	INIT	LDX	#CURDRV
------	----	------	------	-----	---------

DE2A	C6	09		LDB	#9
DE2C	6F	80	INIT2	CLR	0,X+
DE2E	5A			DECB	
DE2F	26	FB		BNE	INIT2

*POINT D'ENTREE CHAUD ET ARRET DES DRIVES

DE31	86	FF	WARM	LDA	##FF
DE33	B7	E0B0		STA	DRVREG
DE36	39			RTS	

*LECTURE D'UN SECTEUR

DE37	17	01A8	READ	LBSR	CHECK
DE3A	8D	5A		BSR	SEEK
DE3C	7D	DE23		TST	CURSID
DE3F	26	04		BNE	SAUT1
DE41	86	8C		LDA	#RDCMNO
DE43	20	02		BRA	SUIT1
DE45	86	8E	SAUT1	LDA	#RDCMNI
DE47	7D	CC34	SUIT1	TST	PRCNT
DE4A	20	03		BRA	READ2
DE4C	113F			SWI3	
DE4E	12			NOP	
DE4F	1A	10	READ2	SEI	
DE51	B7	E000		STA	COMREG
DE54	17	019E		LBSR	DEL28
DE57	5F			CLRB	
DE58	B6	E000	READ3	LDA	COMREG
DE5B	B5	02		BITA	#DRQ
DE5D	26	08		BNE	READ5
DE5F	B5	01		BITA	#BUSY
DE61	26	F5		BNE	READ3
DE63	1F	89		TFR	A,B
DE65	20	0A		BRA	READ6
DE67	B6	E003	READ5	LDA	DATREG
DE6A	A7	80		STA	0,X+
DE6C	5A			DECB	
DE6D	26	E9		BNE	READ3
DE6F	8D	15		BSR	WAIT
DE71	C5	10	READ6	BITB	##10
DE73	27	0C		BEQ	READB
DE75	34	04		FSHS	B
DE77	8E	DE24		LDX	#DNSTY
DE7A	F6	DE1E		LDB	CURDRV
DE7D	63	85		COM	B,X
DE7F	35	04		PULS	B
DE81	C5	1C	READB	BITB	#RDMSK
DE83	1C	EF		CLI	
DE85	39			RTS	

*ATTENTE DE FIN DE COMMANDE

DE86	7D	CC34	WAIT	TST	PRCNT
------	----	------	------	-----	-------


```

DEB9 20 03      BRA WAIT1
DEBB 113F      SWI3
DEBD 12        NOP
DEBE F6 E000    WAIT1 LDB COMREG
DE91 C5 01      BITB #BUSY
DE93 26 F1      BNE WAIT
DE95 39        RTS

*RECHERCHE D'UNE PISTE

DE96 7F DE23    SEEK CLR CURSID
DE99 34 16      PSHS A,B,X
DE9B 30 B9 0101 LEAX 257,X
DE9F BF DF32    STX WRITES+5
DEA2 4D        TSTA
DEA3 27 23      BEQ SEEKS
DEA5 BE DE24    LDX #DNSTY
DEAB F6 DE1E    LDB CURDRV
DEAB E6 85      LDB B,X
DEAD 27 19      BEQ SEEKS
DEAF C6 01      LDB #1
DEB1 B6 DE1E    LDA CURDRV
DEB4 27 04      BEQ SEEKS
DEB6 5C        SEEK6 INCB
DEB7 4A        DECA
DEBB 26 FC      BNE SEEK6
DEBA 53        SEEK5 COMB
DEBB C4 FB      ANDB #%11111011
DEBD F7 E080    STB DRVREG
DECO 35 16      PULS A,B,X
DEC2 C1 12      CMPB #18
DEC4 23 08      BLS SEEK1
DEC6 20 06      BRA SEEK2
DEC8 35 16      SEEKS PULS A,B,X
DECA C1 0A      CMPB #10
DECC 23 03      BLS SEEK1
DECE 7A DE23    SEEK2 DEC CURSID
DED1 F7 E002    SEEK1 STB SECREG
DED4 B1 E001    CMPA TRKREG
DED7 27 12      BEQ SEEK4
DED9 B7 E003    STA DATREG
DEDC 17 0116    LBSR DEL28
DEDF 86 1B      LDA #SKCMND
DEE1 B7 E000    STA COMREG
DEE4 17 010E    LBSR DEL28
DEE7 8D 9D      BSR WAIT
DEE9 C5 10      BITB #*10
DEEB 16 0107    SEEK4 LBRA DEL28

*ECRITURE D'UN SECTEUR

DEEE 17 00F1    WRITE LBSR CHECK
DEF1 8D A3      BSR SEEK
DEF3 34 08      PSHS DP
    
```

```

DEF5 86 E0      LDA #*E0
DEF7 1F 8B      TFR A,DP

        OOE0 SETDP #E0

DEF9 7D DE23    TST CURSID
DEFC 26 04      BNE SAUT2
DEFE C6 AC      LDB #WTCMNO
DF00 20 02      BRA SUIT2
DF02 C6 AE      SAUT2 LDB #WTCMNO
DF04 7D CC34    SUIT2 TST PRCNT
DF07 20 03      BRA WRITE2
DF09 113F      SWI3
DF0B 12        NOP
DF0C 1A 10      WRITE2 SEI
DF0E A6 80      LDA O,X+
DF10 D7 00      STB COMREG
DF12 17 00E0    LBSR DEL28
DF15 D6 00      WRITE3 LDB COMREG
DF17 C5 02      BITB #DR0
DF19 26 12      BNE WRITES
DF1B D6 00      LDB COMREG
DF1D C5 02      BITB #DR0
DF1F 26 0C      BNE WRITES
DF21 D6 00      LDB COMREG
DF23 C5 02      BITB #DR0
DF25 26 06      BNE WRITES
DF27 C5 01      BITB #BUSY
DF29 26 EA      BNE WRITE3
DF2B 20 0C      BRA WRITE6
DF2D 97 03      WRITE5 STA DATREG
DF2F A6 80      LDA O,X+
DF31 8C 0000    CMPX #NFINSC
DF34 26 DF      BNE WRITE3
DF36 17 FF4D    LBSR WAIT
DF39 30 1F      WRITE6 LEAX -1,X
DF3B C5 10      BITB #*10
DF3D 27 0C      BEQ WRITEB
DF3F 34 04      PSHS B
DF41 BE DE24    LDX #DNSTY
DF44 F6 DE1E    LDB CURDRV
DF47 63 85      COM B,X
DF49 35 04      PULS B
DF4B C5 5C      WRITEB BITB #WTMSK
DF4D 1C EF      CLI
DF4F 35 08      PULS DP

        0000 SETDP 00

DF51 39        RTS

*VERIFICATION DE L'ECRITURE

DF52 17 008D    VERIFY LBSR CHECK
    
```

```

DF55 7D DE23    TST CURSID
DF58 26 04      BNE SAUT3
DF5A 86 8C      LDA #RDCMNO
DF5C 20 02      BRA SUIT3
DF5E 86 8E      SAUT3 LDA #RDCMNO
DF60 7D CC34    SUIT3 TST PRCNT
DF63 20 03      BRA VERIF2
DF65 113F      SWI3
DF67 12        NOP
DF68 1A 10      VERIF2 SEI
DF6A B7 E000    STA COMREG
DF6D 17 0085    LBSR DEL28
DF70 17 FF13    LBSR WAIT
DF73 1C EF      CLI
DF75 C5 18      BITB #VERMSK
DF77 39        RTS

*RESTORE LA TETE SUR LA PISTE 0

DF78 34 10      RST PSHS X
DF7A 8D 0F      BSR DRV
DF7C 86 0B      LDA #RSCMND
DF7E B7 E000    STA COMREG
DF81 8D 72      BSR DEL28
DF83 17 FF00    LBSR WAIT
DF86 35 10      PULS X
DF88 C5 D8      BITB #*DB
DF8A 39        RTS

*SELECTION DU DRIVE

DF8B A6 03      DRV LDA 3,X
DF8D 81 02      CMPA #2
DF8F 23 05      BLS DRV2
DF91 C6 0F      LDB #*0F
DF93 1A 01      SEC
DF95 39        RTS
DF96 8D 55      DRV2 BSR FNDTRK
DF98 F6 E001    LDB TRKREG
DF9B E7 84      LDB O,X
DF9D C6 01      LDB #1
DF9F B7 DE1E    STA CURDRV
DFA2 27 04      BEQ FIN
DFA4 5C        RET INCB
DFA5 4A        DECA
DFA6 26 FC      BNE RET
DFAB 53        FIN COMB
DFA9 F7 E080    STB DRVREG
DFAC 8D 3F      BSR FNDTRK
DFAE A6 84      LDA O,X
DFB0 87 E001    STA TRKREG
DFB3 8D 40      BSR DEL28
DFB5 20 27      BRA OK
    
```

```

*TEST DE L'ETAT DU DRIVE

DFB7 A6 03      CHKRDY LDA 3,X
DFB9 B1 02      CMPA #2
DFBB 23 05      BLS TEST
DFBD C6 80      LDB #*80
DFBF 1A 01      SEC
DFC1 39        RTS
DFC2 F6 E000    TEST LDB COMREG
DFC5 C4 80      ANDB #*80
DFC7 27 15      BEQ OK
DFC9 BE 5000    LDX #MISVIT
DFCC 30 1F      TEMPO LEAX -1,X
DFCE 30 1F      LEAX -1,X
DFD0 30 01      LEAX +1,X
DFD2 26 F8      BNE TEMPO
DFD4 F6 E000    QUICK LDB COMREG
DFD7 C4 80      ANDB #*80
DFD9 27 03      BEQ OK
DFDB 1A 01      SEC
DFDD 39        RTS
DFDE 5F        OK CLRBP
DFDF 1C FE      CLC
DFE1 39        RTS

*TEST POUR READ , WRITE ET VERIFY

DFE2 34 16      CHECK PSHS X,D
DFE4 F6 E000    PAPRET LDB COMREG
DFE7 C4 80      ANDB #*80
DFE9 26 F9      BNE PAPRET
DFEB 35 96      PRET PULS X,D,PC

*RECHERCHE D'UNE PISTE

DFED BE DE1F    FNDTRK LDX #DRVTRK
DFF0 F6 DE1E    LDB CURDRV
DFF3 3A        ABX
DFF4 39        RTS

*Sous PROG. DE DELAI

>DFF5 17 0000    DEL28 LBSR DEL14
>DFFB 17 0000    DEL14 LBSR DEL
DFFB 39        DEL RTS

        END
    
```

0 ERREUR(S) DETECTEE(S)

Fig. 2 - Listing des drivers des disquettes double densité

*DEFINITION DES CONSTANTES

```
C107          ORG    $C107
C107 00      DNS    FCB    0
C108 C100    FDB    $C100
C10A 0000    FDB    0
C10C 00      CURSID FCB    0
          C300  SCTBUF EQU  $C300
```

*LECTURE D'UN SECTEUR
*SIMPLE ET DOUBLE DENSITE

*CONSTANTES

```
E000 COMREG EQU  $E000
E001 TRKREG EQU  COMREG+1
E002 SECREG EQU  COMREG+2
E003 DATREG EQU  COMREG+3
E080 DRVREG EQU  COMREG+$80
0002 DRQ    EQU  2
0001 BUSY   EQU  1
001C RDMSK  EQU  $1C
008C RDCMNO EQU  $8C
008E RDCMN1 EQU  $8E
001B SKCMND EQU  $1B
```

C16E ORG \$C16E

00E0 SETDP \$E0

```
C16E 34 0A    READ  PSHS  DP,A
C170 86 E0    LDA    #$E0
C172 1F 8B    TFR    A,DP
C174 35 02    PULS  A
C176 8D 3D    BSR    SEEK
C178 7D C10C  TST   CURSID
C17B 26 04    BNE    SAUT1
C17D 86 8C    LDA    #RDCMNG
C17F 20 02    BRA   SUT1
C181 86 8E    SAUT1 LDA  #RDCMN1
C183 97 00    SUT1  STA  COMREG
C185 8D 65    BSR   DEL28
C187 5F      CLR  B
C188 8E C300 LDX  #SCTBUF
C18B 96 00    READ3 LDA  COMREG
C18D 85 02    BITA  #DRQ
C18F 26 08    BNE  READ5
C191 85 01    BITA  #BUSY
C193 26 F6    BNE  READ3
C195 1F 89    TFR  A,B
C197 20 09    BRA  READ6
```

```
C199 96 03    READ5 LDA  DATREG
C19B A7 80    STA  0,X+
C19D 5A      DECB
C19E 26 EB    BNE  READ3
C1A0 8D 0C    BSR  WAIT
C1A2 C5 10    READ6 BITB #10
C1A4 27 03    BEQ  READ8
C1A6 73 C107 COM  DNS
C1A9 C5 1C    READ8 BITB #RDMSK
C1AB 35 08    PULS DP
C1AD 39      RTS
```

*ATTENTE DE FIN DE COMMANDE

```
C1AE D6 00    WAIT  LDB  COMREG
C1B0 C5 01    BITB  #BUSY
C1B2 26 FA    BNE  WAIT
C1B4 39      RTS
```

*SEEK

```
C1B5 7F C10C  SEEK  CLR  CURSID
C1B8 7D C107  TST   DNS
C1B8 27 0E    BEQ  SIMPLE
C1BD 34 02    PSHS  A
C1BF 86 FA    LDA  #Z11111010
C1C1 97 80    STA  DRVREG
C1C3 35 02    PULS  A
C1C5 C1 12    CMPB  #18
C1C7 23 11    BLS  SEEK1
C1C9 20 0C    BRA  SEEK2
C1CB 34 02    SIMPLE PSHS  A
C1CD 86 FE    LDA  #Z11111110
C1CF 97 80    STA  DRVREG
C1D1 35 02    PULS  A
C1D3 C1 0A    CMPB  #10
C1D5 23 03    BLS  SEEK1
C1D7 7A C10C  SEEK2 DEC  CURSID
C1DA D7 02    SEEK1 STB  SECREG
C1DC 91 01    CMPA  TRKREG
C1DE 27 0C    BEQ  DEL28
C1E0 97 03    STA  DATREG
C1E2 8D 08    BSR  DEL28
C1E4 86 1B    LDA  #SKCMND
C1E6 97 00    STA  COMREG
C1E8 8D 02    BSR  DEL28
C1EA 8D C2    BSR  WAIT
```

*BOUCLE DE DELAI

```
>C1EC 8D C1EF  DEL28 JSR  DEL14
C1EF 8D C1F2  DEL14 JSR  DEL
C1F2 39      DEL  RTS
```

*MODIFICATIONS A APPORTER A LA COMMANDE FORMAT
*POUR FONCTIONNER EN DOUBLE DENSITE A 4 MHZ
*PROGRAMME DE MR R. MILLAUD

*MODIFICATIONS PONCTUELLES

```
019E          ORG    $19E
019E 21      FCB    $21
0216          ORG    $216
0216 10      FCB    $10
0266          ORG    $266
0266 14F6    FDB    $14F6
0277          ORG    $277
0277 2198    FDB    $2198
```

*SOUS PROGRAMME D'ECRITURE D'UNE PISTE

* CONSTANTES

```
E000 COMREG EQU  $E000
E001 TRKREG EQU  COMREG+1
E002 SECREG EQU  COMREG+2
E003 DATREG EQU  COMREG+3
E080 DRVREG EQU  COMREG+$80
DE1E CURDRV EQU  $DE1E
DE1F DRVTRK EQU  CURDRV+1
DE23 CURSID EQU  CURDRV+5
00F4 WTCMDO EQU  $F4
00F6 WTCMD1 EQU  $F6
0002 DRQ    EQU  2
0001 BUSY   EQU  1
0800 WORK   EQU  $0800
14F6 SWKEND EQU  $14F6
2198 DWKEND EQU  $2198
0024 DRN    EQU  $24
0027 DENSE  EQU  $27
```

```
0710          ORG    $710
0710 8E 0024  WRTTRK LDX  #DRN
0713 30 1D    LEAX -3,X
0715 A6 03    LDA  3,X
0717 17 0083  LBSR FNDTRK
071A F6 E001  LDB  TRKREG
071D E7 84    STB  0,X
071F C6 01    LDB  #1
```

```
0721 B7 DE1E  STA  CURDRV
0724 27 04    BEQ  FIN
0726 5C      RET  INCB
0727 4A      DECA
0728 26 FC    FC    BNE  RET
072A 53      FIN  COMB
072B 0D 27    TST   DENSE
072D 27 02    BEQ  SIMPLE
072F C4 FB    ANDB #Z11111011
0731 F7 E080  SIMPLE STB  DRVREG
0734 8D 67    BSR  FNDTRK
0736 A6 84    LDA  0,X
0738 B7 E001  STA  TRKREG
073B 8D 59    BSR  DELAY
073D 0D 27    TST   DENSE
073F 27 05    BEQ  SID
0741 8E 2199  LDX  #DWKEND+1
0744 20 03    BRA  POSTES
0746 8E 14F7  SID  LDX  #SWKEND+1
0749 BF 0783  POSTES STX  WRTTR4+5
074C 8E 0800  LDX  #WORK
074F 34 0C    PSHS B,DP
0751 86 E0    LDA  #$E0
0753 1F 8B    TFR  A,DP
```

00E0

```
0755 7D DE23  TST   CURSID
0758 26 04    BNE  SAUT1
075A C6 F4    LDB  #WTCMDO
075C 20 02    BRA  SUT1
075E C6 F6    SAUT1 LDB  #WTCMD1
0760 A6 80    SUT1  LDA  0,X+
0762 D7 00    STB  COMREG
0764 8D 30    BSR  DELAY
0766 D6 00    WRBL LDB  COMREG
0768 C5 02    BITB #DRQ
076A 26 12    BNE  WRTTR4
076C D6 00    LDB  COMREG
076E C5 02    BITB #DRQ
0770 26 0C    BNE  WRTTR4
0772 D6 00    LDB  COMREG
0774 C5 02    BITB #DRQ
0776 26 06    BNE  WRTTR4
0778 C5 01    BITB #BUSY
077A 26 EA    BNE  WRBL
077C 20 0D    BRA  WRTTR8
077E 97 03    WRTTR4 STB  DATREG
0780 A6 80    LDA  0,X+
0782 8C 14F7  CMPX #SWKEND+1
0785 26 DF    BNE  WRBL
0787 30 1F    LEAX -1,X
0789 8D 03    BSR  WAIT
078B 35 0C    WRTTR8 PULS B,DP
```

Fig. 3. - Listing de la modification du chargeur du DOS


```

0000          SETDP 00
07BD 39          RTS
07BE B6 E000    WAIT LDA COMREG
0791 85 01      BITA #BUSY
0793 26 F9      BNE WAIT
0795 39          RTS

*BOUCLE DE DELAI
>0796 17 0000   DELAY LBSR DELAY2
>0797 17 0000   DELAY2 LBSR DELAY4
079C 39          DELAY4 RTS

079D 8E DE1F    FNDTRK LDX #DRVTRK
07A0 F6 DE1E    LDB CURDRV
07A3 3A         ABX
07A4 39         RTS

END
    
```

0 ERREUR(S) DETECTEE(S)

Fig. 4 - Listing de la modification de la commande FORMAT

O ou N selon ce que vous désirez.

Lorsque ces opérations sont terminées, essayez le fonctionnement de tous vos programmes pour lesquels vous ne devez constater aucune anomalie. En effet, les modifications indiquées ne touchent pas aux tables de sauts du DOS et seuls les programmes mal écrits et utilisant des parties du DOS en principe non accessibles à l'utilisateur pourraient être affectés.

- Avec cette modification, les amateurs de fortes capacités devraient être satisfaits puisque l'on dispose ainsi de :
- 702 secteurs de 256 octets en simple face 40 pistes
 - 1404 secteurs de 256 octets en double face 40 pistes
 - 1422 secteurs de 256 octets en simple face 80 pistes
 - 2844 secteurs de 256 octets en double face 80 pistes, soit plus de 700 kilo-octets sur une seule disquette.

Réponses à quelques questions

La double densité a fait couler beaucoup d'encre, tant dans votre courrier que sur certaines feuilles dites d'informations imprimées par certains, aussi estimons-nous utile de mettre les choses au point, maintenant que vous êtes tous à même de l'utiliser.

Le premier point concerne la précompensation d'écriture. Cette technique est fortement

conseillée en double densité pour les lecteurs 8 pouces. Pour les lecteurs 5 pouces, elle est également souhaitable mais cela n'a rien d'impératif et, si on ne l'utilise pas, tout au plus risque-t-on quelques secteurs défectueux de temps en temps lors du formatage en double densité. Cette précompensation dépend de la qualité des disquettes et surtout de celle des lecteurs. Avec du bon matériel, bien réglé et bien entretenu, elle est inutile (en 5 pouces, répétons le).

Du fait de cette « inutilité », il est possible d'utiliser la carte IFD09 en double densité sans modification, puisque nous l'avions conçue pour cela à l'origine.

Pour répondre à d'autres questions et critiques à ce sujet : nous aurions pu monter sur cette carte un WD 2795 mais :

- A l'époque de la description de la carte dans ces pages, ce circuit plafonnait à 900 F l'unité !
- Pour être bien utilisé et pour que le circuit de séparation de données interne soit bien réglé, il faut disposer d'un oscilloscope. Vous n'êtes pas tous en possession de cet engin et le réglage « pifométrique » en mettant le potentiomètre « au milieu de la zone où cela fonctionne » (sic) est une absurdité. En effet, à quoi sert de prendre un tel circuit pour être dans les meilleures conditions possibles si c'est pour le régler n'importe comment !

Le dernier point concerne les disquettes. Ne nous faites

pas dire ce que nous n'avons pas écrit ; les disquettes certifiées double face, double densité, 96 TPI sont meilleures que les simple face, simple densité 48 TPI. Nous n'utilisons cependant que ces dernières (environ 20 F par 10 chez les bons revendeurs) sans avoir constaté le moindre problème.

Si vous tenez à vos fichiers, il vaut bien mieux faire plusieurs copies soigneusement actualisées de ceux-ci sur des disquettes normales que n'utiliser qu'une « super » disquette dont vous n'aurez pas de copie sous prétexte qu'elle est certifiée.

Du logiciel comme s'il en pleuvait

Après le logiciel gratuit, voyons le logiciel payant, toujours dans le but d'essayer de répondre positivement à vos critiques, que nous estimons tout à fait justifiées, il est vrai.

La société Micropross de Colmar commercialise depuis quelque temps déjà des logiciels qu'elle a bien voulu soumettre à nos essais, et nous croyons utile de vous en dire quelques mots ci-après car ils sont à même de répondre aux aspirations d'un certain nombre d'entre vous.

Précisons tout de suite que tous ces logiciels sont vendus sur disquette simple face, simple densité, 40 ou 80 pistes (à préciser à la commande car 40 est pris par défaut), et qu'ils sont fournis avec une notice sous forme de fichier texte sur la disquette. C'est une bonne solution mais elle handicape ceux d'entre vous qui n'ont pas d'imprimante, surtout pour les programmes complexes dont les notices sont assez longues. Ce n'est pas dramatique car, en général, l'utilisation de ces programmes ne peut se concevoir qu'avec une imprimante (imaginez une facturation sans imprimante !)

Les premiers programmes proposés sont des extensions du Basic que nous vous avons fourni. Deux programmes, BASEDIT et BASCOPY, permettent d'enchaîner les éditions de programmes Basic ou de pren-

dre des morceaux de programmes déjà existants pour en éditer un nouveau. Cela peut être utile pour ceux d'entre vous qui programment beaucoup dans ce langage et qui ont ainsi de nombreux modules susceptibles d'être utilisés dans divers programmes. Un troisième programme, appelé EXT BASIC, est à ajouter (au moyen d'un APPEND mais c'est expliqué dans la notice) à TBASIC pour lui donner un certain nombre de fonctions supplémentaires très intéressantes dont voici un extrait :

- Fonction SCROLL accessible avec un USR ;
- Tracé automatique d'un cadre graphique autour de l'écran ;
- Mise en marche et suppression du curseur ;
- Mise en marche ou suppression de l'écho du clavier sur l'écran ;
- Hard copie de l'écran sur imprimante ;
- Fonction INKEY\$;
- Activation et désactivation directe de l'imprimante sans sortir du Basic ;
- Enfin et surtout éditeur de texte sous Basic !

Toutes ces fonctions s'appellent avec des USR aussi simplement que si les instructions Basic correspondantes existaient. Ainsi, pour avoir un cadre autour de l'écran, il suffit d'écrire dans un programme USR(5). L'éditeur n'est pas très souple d'emploi et est assez simple ; c'est cependant très satisfaisant pour corriger, même de façon importante, des programmes Basic tout en restant sous Basic. Enfin, la fonction hard copie d'écran est appréciable. Nous décernerons donc une bonne note à ces extensions. Attention ! elles sont indispensables si vous voulez utiliser les autres logiciels Basic proposés par Micropross que sont la gestion de budget familial, la facturation, la gestion de stock et la gestion d'étiquettes.

Au niveau des utilitaires, signalons l'existence d'une disquette EXT DOS (qui n'a rien à voir avec l'EXT DOS des informations 6809), qui supporte un certain nombre de programmes en langage machine que voici :

— VDISK qui est un programme analogue à celui que nous allons publier aujourd'hui sous le nom de PSEUDISK. Pour la petite histoire, précisons que PSEUDISK était en notre possession avant la réception des disquettes Micropross et qu'il ne faut pas voir là un quelconque « coup en traître ».

— RESTAURE qui permet de tester l'intégrité d'une disquette, d'éliminer les secteurs défectueux et de rechaîner les secteurs vides.

— SECTEUR qui n'est autre que le célèbre EXAMINE de chez TSC et qui permet de lire, écrire ou modifier tout secteur d'une disquette, de connaître le chaînage des secteurs, etc... Seul reproche fait à Micropross : la notice trop succincte de ce programme qui laisse dans l'ombre certaines de ses possibilités.

— REGVIT est un programme original permettant de régler la vitesse de rotation des lecteurs. Il procède par mesure du temps entre les impulsions d'index et affiche le temps théorique (200 ms) et le temps mesuré qui est réactualisé toutes les secondes environ. Ce programme fonctionne très bien sauf si on lui donne un numéro de lecteur égal ou supérieur à 2, ce qui le bloque et oblige à faire un RESET.

— LIST 1 est une commande LIST dérivée de celle du DOS mais offrant des options supplémentaires dont pagination à partir du numéro de votre choix, et surtout possibilité d'envoyer à l'imprimante des chaînes de caractères de configuration pour sélectionner divers modes de fonctionnement de celle-ci. Les indications de configuration de ces chaînes sont données dans la notice de LIST 1, ce qui permet à chacun de configurer le programme pour sa machine.

— TESTMEM qui est un programme de test mémoire, pouvant fonctionner dans toutes les pages et qui procède par écriture, attente, lecture de 55 et de AA sur l'étendue mémoire spécifiée. Ce programme fonctionne, mais nous ne sommes pas de l'avis de Micropross qui affirme que si le test a pu tourner plusieurs heures sans défaut, la mémoire

est bonne. En effet, comme nous l'avons déjà expliqué, la RAM dynamique peut avoir deux types de problèmes : un problème de rafraîchissement, auquel cas ce test le détectera, mais n'importe quel test simple le verra aussi et en quelques minutes (à la limite, un I sous TAVBUG09 suivi par un D de la même zone une dizaine de minutes après suffit). Le deuxième type de défaut est celui des mémoires dites « pattern sensitive », c'est-à-dire sensibles à leur contenu. Il faut pour détecter cela plusieurs jours de test permanent en écrivant des valeurs aléatoires dans les mémoires. Nous mettrons donc un bémol quant à l'efficacité de TESTMEM.

Quoi qu'il en soit, cette disquette est intéressante, ne serait-ce que pour REGVIT, LIST 1, RESTAURE et SECTEUR, et elle a sa place chez tout amateur désirant prendre soin de son système.

Un certain nombre de programmes Basic, beaucoup plus importants que ceux que nous venons de voir, sont ensuite proposés. Ces programmes sont fournis en Basic « compilé » (fichiers.BAC), mais le programme source en clair (extension.BAS) figure aussi sur les disquettes ce qui permet toutes les adaptations à d'éventuels cas particuliers. Une documentation complète les accompagne, sous forme de fichier texte sur disquette. Ces programmes sont les suivants :

— BUDGET qui est une gestion de budget familial, très performante et très souple d'emploi.

— FACTURES qui est un programme de facturation à même d'intéresser des PME, des petits commerçants ou des petits artisans.

— GESTOC qui est un programme de gestion de stocks visant les mêmes utilisateurs que ceux de FACTURES.

— GESTETIQ qui est un programme de gestion d'étiquettes permettant la réalisation de mailings et autres choses similaires.

— COMPTA qui est un programme de comptabilité générale suivant le nouveau plan comptable.

Nous avons essayé pour

l'instant le programme BUDGET, susceptible d'intéresser la majorité d'entre vous et nous en avons été très satisfait. Il faut cependant préciser qu'il nécessite EXT BASIC vu ci-avant et, également, la carte IVG ou IVG09 et deux lecteurs de disquettes. De plus, la version qui nous a été prêtée comportait, dans tous les programmes, une grossière erreur au niveau des sous-programmes Basic d'adressage du curseur, rendant illisible tous les textes affichés à l'écran. Si vous êtes dans ce cas, il suffit de chercher, dans les listings des programmes, la ligne contenant :

PRINT CHR\$(27); « = » ;
CHR\$(32+CO%); CHR\$(32+LI%); : RETURN

et d'y remplacer CO par LI et réciproquement. L'auteur des programmes semble avoir mélangé ligne et colonne dans l'adressage direct du curseur permis par TAVBUG09 V1.0. Cette correction faite (et qui devrait l'être sur les disquettes livrées par Micropross), le logiciel est tout à fait utilisable et nous n'avons pas réussi à le « planter » malgré beaucoup de bonne volonté !

La place nous étant comptée, nous vous dirons quelques mots des autres programmes dans notre prochain numéro.

En vrac

La liste des réalisateurs du système ayant répondu au questionnaire de notre numéro 1702 et ayant autorisé la communication de leurs nom et adresse va être éditée en fin de ce mois. Elle pourra être fournie à toutes les personnes qui en feront la demande à l'auteur dans les conditions suivantes :

— La demande doit être adressée directement à l'auteur de ces lignes et ne doit comporter aucune question technique.

— Elle ne sera fournie qu'aux personnes ayant répondu au questionnaire et ayant donné leur accord pour que leurs coordonnées y figurent.

— Elle sera fournie uniquement si votre demande est accompagnée d'une enveloppe autocollante, format 16 X 22, libellée à votre adresse et

affranchie à 10,70 F. Cette enveloppe devra être accompagnée de 10 timbres à 2,10 F pour frais de photocopie. Ces 10 timbres ne doivent pas être remplacés par un amalgame quelconque de timbres conduisant à la même somme.

— Les lecteurs étrangers voudront bien remplacer l'affranchissement et les 10 timbres par 15 coupons réponse internationaux.

— Toutes les demandes non exactement conformes à ces indications seront rejetées vu les pertes de temps qu'elles engendrent.

Vu le nombre de réponses au questionnaire reçues, un délai de 3 semaines sera nécessaire pour satisfaire les demandes de listes. Nous vous demandons de faire preuve de patience.

En raison de divers problèmes de mise au point et de concurrence malhonnête à laquelle se livrent certaines personnes par voie de petites annonces, les programmes GESFICHE, REPAR, ECHAUT et PROTEXT ont subi quelques retards de mise à disposition. Ces problèmes devraient être résolus lorsque vous lirez ces lignes pour GESFICHE. En ce qui concerne REPAR et ECHAUT, ils devraient être disponibles mi-octobre, quant à PROTEXT, nous avons décidé de n'en pas terminer le développement. Nous vous présentons nos excuses pour ces retards indépendants de notre volonté.

Conclusion

Nous en resterons là pour aujourd'hui. Notre prochain article sera consacré à la fin de ces présentations de logiciels avec ceux annoncés ci-avant, mais aussi avec les œuvres intéressantes de certains d'entre vous.

Nous vous parlerons également d'un dynamique club d'utilisateurs de « Tavernier 6809 » et aborderons, enfin, la nouvelle carte alphanumérique et graphique couleur que nous avons dû différer jusqu'à ce jour en raison de certains retards de fabrication de son circuit imprimé.

C. TAVERNIER