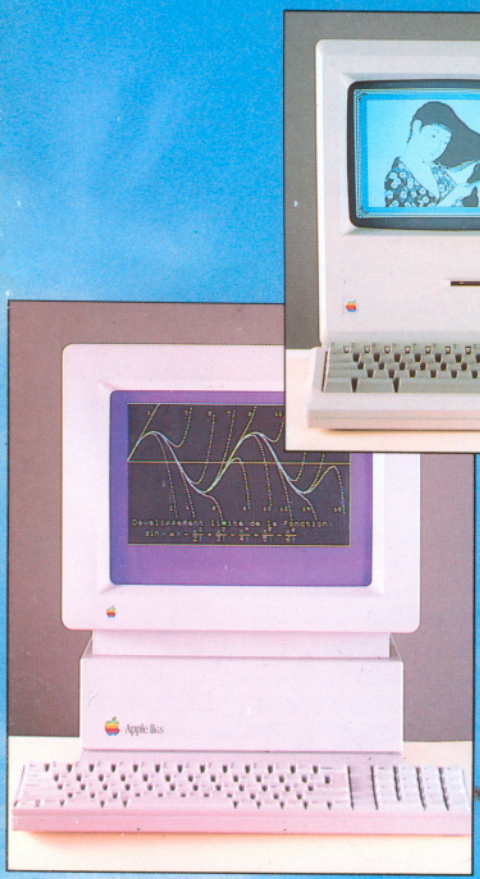
La revue francophone indépendante pour les utilisateurs des Apple][+, //e, //e+, //c, IIGS™ et Macintosh™

POM'S

- Communication
- Bancs d'essai
- Graphismes
- Utilitaires



ISSN 0294-6068
M 2366 - 31 - 45,00 F

Communication CalvaCom Apple // & Macintosh

L'ouverture des Apple à la communication dans les pages de Pom's se poursuit par un programme qui allie au «savoir communiquer» de CalvaCom l'indispensable liaison à l'ordinateur.

CalvaCom

Pour les lecteurs non encore au fait des possibilités de CalvaCom, une présentation sommaire s'impose. Il se différencie des serveurs habituellement consultés à l'aide du Minitel par plusieurs points :

- on y accède par le 36 13 (ou 36 00 36 01, une unité toutes les 6 mn moins réduction d'horaires), le coût de connexion étant réglé directement à CalvaCom (prélèvement ou carte bancaire) ;
- la consultation se fait sans caractère graphique : seuls les caractères ASCII arrivent au modem ou au Minitel. Ce dernier affiche les messages en mode "rouleau" tel une imprimante ;
- passer de menus en menus est bien plus simple et efficace que sur les serveurs *kiosque*, l'accès à l'information est particulièrement rapide ;
- il a été conçu dans l'optique de la connexion à l'ordinateur : les informations disponibles sont réellement exploitables par le micro.

CalvaCom met à la disposition des abonnés :

- une messagerie électronique qui permet à chacun de classer ses messages, de répondre simplement, de recevoir des messages directement de l'ordinateur de l'abonné pour éviter de perdre du temps à les composer en direct ;
- le service TELEX, en émission et réception ;
- des forums électroniques ;
- un système de téléconférence (convivialité) ;
- un système de petites annonces ;
- des magazines électroniques ;
- des bibliothèques de programmes téléchargeables ;
- les dépêches de l'Agence France Presse ;
- les cours de la bourse avec possibilité de passer des ordres ;
- des 'cités' centrées sur Atari, IBM, Apple //, Macintosh...
- ...

Le programme CLV_POMS

Proposé en version Apple // et en version Macintosh, CLV_POMS donne les possibilités suivantes :

- enregistrement de consultations CalvaCom. Les fichiers constitués sont récupérables par votre traitement de texte, votre tableur... et ils ne sont limités en dimension que par la capacité de votre support disque ;
- mode téléscripteur : impression au fur et à mesure de la réception, mode particulièrement efficace lors de l'"écoute" des dépêches AFP (qui *tombent* au fur et à mesure de l'actualité) ou des cours de la bourse ;
- exportation de fichiers vers CalvaCom : vous voulez passer une petite annonce, nous laisser un message (le code de Pom's est EMP11), participer à un forum, constituer un fichier sur le serveur pour utilisation ultérieure ? Préparez-le en traitement de textes et envoyez-le par cette option : il sera transmis sans protocole mais au bon débit, en respectant la mini-pause de fin de ligne ;
- procédures : c'est une utilisation particulière du mode précédent. Il est possible de préparer un texte comprenant les options choisies pour une consultation habituelle et de l'envoyer à CalvaCom pour atteindre plus rapidement et sans erreur les informations. Pour consulter automatiquement les publicités du forum Macintosh, la procédure pourrait être la suivante :

```
N°_abonné
Mot_de_passe
ci
mac
fo pub
```

Des pauses peuvent être incluses dans le fichier pour tenir compte du temps d'affichage des nouveautés après le mot de passe par exemple.

- impression d'un fichier de textes sur disque : il s'agira souvent du résultat de votre dernière connexion ou d'une dernière vérification de votre texte avant exportation.

Bien entendu, les programmes CLV_POMS recodent les minuscules accentuées tant en émission qu'en réception, tant sur le Mac que sur l'Apple //, tant avec un Minitel 40 colonnes que sur le 80 colonnes.

Configuration

CLV_POMS fonctionne sur :

Apple][+ avec ROM minuscules, carte langage et carte Super Série Apple ;
Apple //e avec carte Super Série Apple ;
Apple //c ;
Apple IIGS sur son port modem intégré ou avec une carte SSC ;
Macintosh 128, 512, Plus ou **SE**.

Il nécessite :
- un modem ou un Minitel (les modèles 80 colonnes sont bien plus agréables),
- un câble de liaison disponible à la revue.

La disquette

CLV_POMS, écrit en assembleur, est livré sur une disquette 3'5 pour Macintosh et au choix 5'25 ou 3'5 pour l'Apple //.

La version Apple // fonctionne sous ProDOS avec tous les volumes (disquettes 140Ko, 800Ko, disque /RAM, Speedisk, disques durs...).

La version Mac fonctionne avec tous les supports, tant en MFS qu'en HFS.

CLV_POMS : 200,00 F TTC, câble de liaison 225,00 F, bon de commande page 74.

Numéro 31
juillet-août 1987

Éditorial — *Hervé Thiriez* — Page 5

Peeks et Pokes longs sur GS — *Christian Piard* — Page 6

Le Basic et l'heure — Page 8

Graphes et fonctions — *Claude-Lucien CAUET* — Page 9

GOGO — *Patrice Neveu* — Page 22

Disquette Pom's 31 Apple // — Page 26

Essai ; une nouvelle carte : Speedisk — *Alexandre Duback* — Page 27

Essai ; une carte vocale : Valentine — Page 28

Écriture proportionelle en DHGR — *Frédéric Rosay* — Page 31

Un nouveau produit Pom's : Éditeur Vidéotex — Page 37

Essai Macintosh : Writer + — *Philippe Mathieu* — Page 40

Décompacteur de programmes Basic — *Julien Thomas* — Page 43

Disquette Pom's 31 Mac — Page 45

Un accessoire 'minimum' : MinAcc — *Alain Bohec* — Page 46

Kino — *Alain Bohec* — Page 47

Les nouvelles version d'InterPom's — Page 52

Le voyageur — *Serge Cattan* — Page 55

Turbo-Pascal : Graphisme, suite... — *Michel de Crémoux & Nicolas Martin* — Page 59

Bibliographie — *Alexandre Duback* — Page 62

LeMouse : souris & menus — *Le Pham Hiep* — Page 63

µ-informations — *Jean-Michel Gourévitch* — Page 67

Les annonceurs ; Apple : pages 38 et 39 ; LOGMA S.A. : page 76 ; mak fan : page 4 ; Q.S.I. : page 44.

Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39 51 24 43. Directeur de la publication : Hervé Thiriez

mak fan

mak fan

LA NOUVELLE REVUE DES FANAS DU MACINTOSH

C'est ***LA NOUVELLE REVUE*** *de l'utilisateur pointu de* ***"Macintosh".***
C'est l'espace où les utilisateurs de ***4D, Excel, Writer +, Jazz*** *et* ***TK Solver*** *se retrouvent.*
C'est le magazine de celui qui veut ***se dégourdir les doigts*** *et les neurones sur le clavier, avec les "tutorials" en Pascal, Assembleur, "C", Lisp et Modula.*
C'est la revue où vous pouvez écrire si vous avez une belle réalisation d'application à votre actif,
c'est Votre revue disponible dans les kiosques.

mak fan BP 33, 94471 - BOISSY Cédex Tél. : 45 99 02 10

SOMMAIRE DU N° 4 (64 pages)
actuellement dans les kiosques

MAC WORLD
DESKTOP PUBLISHING
INFOS PRESSE
MACROS D'EXCEL
INTRODUCTION A 4è DIMENSION
BASEGEST (4èD)
WRITER PLUS
TUTORIAL : ASSEMBLEUR 68000
LE_LISP
ALGORITHMES
L'HOMME DU MOIS (UN GRAPHISTE)
RÉCRÉATIONS ALGORITHMIQUES
CALCULS EN MULTI-PRÉCISION

bulletin d'abonnement ***(Bulletin à recopier ou à photocopier)***

à retourner à : **mak fan BP 33, 94471 - BOISSY Cédex**

❑ **Oui, je m'abonne pour 1 an** [8 numéros] à partir du N°.....
au prix de **260 F** au lieu de 280 F

❑ Abonnement de soutien : 350 F (8 numéros)

Frais de port inclus - Étranger : 45 F le numéro.

VENTE AU NUMÉRO : ❑ N° 1 ❑ N° 2 au prix unitaire de 30 F (40 F pour l'Étranger).
❑ N° 3 au prix unitaire de 35 F (45 F pour l'Étranger).

ci-joint un chèque de.....FF à l'ordre de **mak fan**

Nom : .....Prénom : ..... Adresse : .....
..... Code postal : .....Ville : ..... Profession :

Ont collaboré à ce numéro

Alexandre Avrane – Jean-Luc Bazanegue
Alain Bohec – Serge Cattan
Claude-Lucien Cauet
Michel de Crémoux
Alexandre Duback
Jean-Michel Gourévitch
Olivier Herz – Le Pham Hiep
Nicolas Martin – Philippe Mathieu
Gérard Michel – Patrice Neveu
Christian Piard – Frédéric Rosay
Hervé Thiriez – Julien Thomas

Directeur de la publication rédacteur en chef

Hervé Thiriez

Rédacteurs

Alexandre Avrane – Olivier Herz

Siège social

Éditions MEV
12, rue d'Anjou
78000 Versailles
☎ (1) 39.51.24.43

Publicité

Éditions MEV

Diffusion

N.M.P.P.

Impression

Berger-Levrault
18, rue des Glacis
54000 Nancy
☎ 83.35.61.44

Photos de couverture

CP & JLB

Photogravure

Graphotec
21, chemin de la Tour
92350 Le Plessis-Robinson
☎ (1) 46.30.44.49

Pom's est une revue indépendante non rattachée à Apple Computer, Inc. ni à Apple Computer France S.A.R.L. Apple, le logo Apple, Mac et le logo Macintosh sont des marques déposées d'Apple Computer, Inc.

IBM est une marque déposée de International Business Machine. PC et AT sont des marques déposée de la Société IBM.

Éditorial

Une fois n'est pas coutume, voici en couverture – et sur fond de loisirs estivaux – une illustration du fonctionnement de deux programmes que vous pourrez trouver dans ce numéro : Schémateur, *volumineux, dans le domaine de la création graphique et* Peeks/Pokes longs *qui en fait voir de toutes les couleurs – ou presque – sur l'Apple IIGS et derrière le logo "Pom's".*

Pom's poursuit sa lancée dans l'exploration des voies communicantes en vous proposant trois disquettes éditées en dehors des pages de la revue :

InterPom's *– le programme qui, depuis le numéro 28, relie vos Apple // et/ou Macintosh par téléphone via Minitel – a reçu de nombreuses améliorations dans la version 2, version capable de communiquer de la même manière avec les micros IBM® et compatibles.*

CLV_Pom's *vous est proposé pour répondre aux besoins de nombre de lecteurs abonnés à CalvaCom. Que se soit pour Apple // ou Macintosh,* CLV_Pom's *se présente sous la forme d'une application autonome et est compatible 'toutes configurations'.*

Enfin, un éditeur vidéotex *en CP/M, pour les Apple // munis d'une carte 80 colonnes ainsi que d'une carte Z80.*

Communiquer n'a pas été facile pour tous nos lecteurs ; vous êtes nombreux à avoir attendu longtemps – parfois un mois – la réception de votre câble Apple/Minitel décrit dans les numéros 27 et 28. En effet, nous avions envisagé la distribution de ce type de câble comme un service aux lecteurs qui n'avaient pas la possibilité d'effectuer la réalisation eux-même, or, la parution d'articles dithyrambiques dans d'autres pages que celles de Pom's nous a valu un afflux de commandes auquel nous ne nous attendions pas. Tout est désormais rentré dans l'ordre et nous prions ceux d'entre-vous qui ont attendu de bien vouloir nous excuser de ce retard regrettable.

Pour revenir au numéro 31 proprement dit, et sans aller jusqu'à décrire chaque article (il y en a beaucoup et ce serait trop long pour un éditorial !), on peut signaler la présence de trois bancs d'essais dont deux 'hard' pour l'Apple // : la carte vocale Valentine *et le 'disque-RAM'* Speedisk. *Le banc d'essai 'soft' est celui de* Writer +, *traitement de textes pour Macintosh.*

Hervé Thiriez

Peeks et Pokes longs sur Apple IIGS

Christian Piard

L'Applesoft semble bien perdu devant 254 des 255 bancs de 64Ko du IIGS. En effet, comment accéder depuis le Basic à la ROM en page 255, à la page Super Haute Résolution en page 225 (adresses 14 753 792 à 14 778 367) par exemple ?

La petite routine présentée ici dote le Basic de PEEK et POKE longs, c'est-à-dire accédant à l'ensemble de la mémoire. Deux programmes de démonstration en illustrent les possibilités.

Mode d'emploi

La routine s'installe à l'adresse $300, mais ne demande qu'à être assemblée ailleurs en cas de besoin.

L'initialisation se fait en mode direct par :

```
- PIC.POC
```

et dans un programme par :

```
PRINT CHR$(4)"-PIC.POC"
```

Voici la syntaxe des commandes disponibles :

& PEEK page, adresse, var%

Page est le numéro en décimal du banc où la lecture se fera.

Adresse est l'adresse en décimal dans ladite page.

Var% est la variable entière qui recevra la valeur de l'octet souhaité.

Par exemple :

```
& PEEK 255, 65535, GF%
```

affectera à GF% la valeur de la dernière adresse mémoire.

```
& PEEK 1-1,MLI,A1%(2)
```

affectera à la 3ème variable du tableau A1% la valeur 76 si on a définit MLI=48896 ($BF00) et que ProDOS est actif.

Page supérieure à 255 et **a**dresse supérieure à 65535 conduisent à un «Illegal quantity error» mérité. Si la variable réceptrice n'est pas entière (%), l'utilisateur sera gratifié du «Type mismatch error».

& POKE page, adresse, valeur

Même principe pour **P**age et **A**dresse ; **V**aleur sera comprise entre 0 et 255 sous peine de message d'erreur.

Des exemples

Le programme LIT.MESS.GS est une illustration du PEEK : il lit et affiche les messages stockés en page 255 (ROM), particulièrement ceux affichés par le tableau de bord.

Cette zone est visée par un pointeur dont l'adresse est donnée en $E100C0/E100C2. Sous réserve de bien respecter les conventions "poids fort à 1, poids fort à 0", il suffit de changer cette adresse et le GS affichera votre table de message en français par exemple.

Le programme GSCOLOR ne fait qu'un dégradé de couleurs en page Super Haute Résolution, semblable à celui de la couverture de ce numéro. Il prend son temps (9 minutes en mode rapide...) mais illustre le fonctionnement du &POKE par 64000 points à l'écran. On peut rappeler succinctement la structure d'une page SHGR.

La SHGR

Une telle page se situe dans la page $E1 (225) entre $2000 et $9FFF.

de $9E00 à $9FFF

16 palettes de 16 couleurs chacune composée de 2 octets ou 32 bits :

4 bits pour le vert
4 bits pour le bleu
4 bits à 0
4 bits pour le rouge

On obtient un jaune vif avec : F00F c'est-à-dire seize 16èmes de vert, zéro 16ème de bleu et seize 16èmes de rouge.

de $9D00 à $9DC7

200 octets (1 par ligne à l'écran) dont les 4 bits de poids faible donnent le numéro de la palette qui sera utilisée par la ligne concernée.

Le bit 7 sera à 0 si la ligne comprend 320 point, c'est le cas dans notre exemple.

Le bit 6 à 1 provoquerait une interruption en fin de ligne pour changer une couleur de la palette avant affichage de la ligne suivante par exemple.

de $2000 à $9CFF

32000 octets pour les 64000 points à l'écran (200 lignes de 320 pixels). Chaque point est représenté par un demi octet qui indique, de 0 à 15 le numéro de sa couleur dans la palette réservée pour la ligne considérée.

Source PIC.POC.S
Assembleur ProCODE

```
0
1   *--------------------------------
2   * PEEK & POKE LONGS SUR APPLE IIGS
3   *--------------------------------
4
5           DSK  PIC.POC
6
7   AMPERV  =    $3F5       ; vecteur &
8   FRNUM   =    $DD67      ; calcule une formule
9   CHKCOM  =    $DEBE      ; virgule ?
10  GETADR  =    $E752      ; FAC -> entier
11  GETBYTC =    $E6F5      ; formule sur un octet
12  PTRGET  =    $DFE3      ; recherche d'une variable
13  MISMATCH =   $DD76      ; message type mismatch
14  VARNAM  =    $81        ; nom de la variable
15  VARPNT  =    $83        ; pointeur sur sa valeur
16
17          ORG  $300
18
19          LDA  AMPERV+1   ; vecteur déjà installé ?
20          CMP  #<DEBUT
21          BNE  INIT
22          LDA  AMPERV+2
23          CMP  #>DEBUT
24          BEQ  INITOK
25  INIT    LDA  AMPERV+1   ; non, on sauvegarde l'éventuel
26          STA  ANCVEC     ; ancien vecteur
27          LDA  AMPERV+2
28          STA  ANCVEC+1
29          LDA  #$4C       ; et on installe le nouveau
30          STA  AMPERV
31          LDA  #<DEBUT
32          STA  AMPERV+1
33          LDA  #>DEBUT
34          STA  AMPERV+2
35  INITOK  RTS
36
37  DEBUT   CMP  #$B9       ; est-ce POKE
38          BEQ  POKE
39          CMP  #$E2       ; est-ce PEEK
40          BNE  AUTRE
41
42  PEEK    LDX  #$FF       ; c'est PEEK
43          STX  DP_PEEK    ; on positionne un drapeau
44          BMI  POKE1
45
46  POKE    LDX  #0         ; c'est POKE
47          STX  DP_PEEK
48  POKE1   JSR  GETBYTC    ; 1er paramètre = 1 octet
49          STX  N_PAGE     ; c'est le n° de page de 64Ko
50          STX  N_PAGE1
51
52          JSR  CHKCOM     ; virgule ?
53
54          JSR  FRNUM      ; 2ème paramètre sur deux octets
55          JSR  GETADR     ; c'est l'adresse
56          LDX  $50        ; on prépare l'adresse du Peek
57          STX  ADRESSE    ; ou du Poke
58          STX  ADRESSE1
59          STA  ADRESSE+1
60          STA  ADRESSE1+1
61          BIT  DP_PEEK    ; est-ce Peek ?
62          BMI  S_PEEK
63
64          JSR  GETBYTC    ; 3ème paramètre : octet à poker
65          STX  A_POKER
66
67          JMP  LPOKE      ; on poke
68
69  S_PEEK  JSR  CHKCOM     ; virgule ?
70          JSR  PTRGET     ; cherche pointeur sur variable
71          STA  VARPNT     ; le sauve
72          STY  VARPNT+1
73          LDA  VARNAM     ; la variable doit être entière
74          AND  VARNAM+1   ; C%, CA% ou D%(4) par exemple
75          BMI  ]1
76          JMP  MISMATCH   ; sinon Type mismatch error
77
78  ]1      JSR  LPEEK      ; on va chercher l'octet
79          LDY  #1         ; et on le stocke à l'adresse
80          STA  (VARPNT),Y ; de la variable choisie
81          DEY
82          TYA
83          STA  (VARPNT),Y
84          RTS
85
86  AUTRE   JMP  (ANCVEC)   ; saute à l'ancien vecteur
87
88  LPOKE   DFB  $18        ; CLC
89          DFB  $FB        ; XCE passage en 16 bits
90          DFB  $A9        ; LDA #donnée
91  A_POKER DFB  0
92          DFB  $8F        ; STA $000000
93  ADRESSE DFB  0,0        ;
94  N_PAGE  DFB  0          ;
95  FIN     DFB  $38        ; SEC Retour au 6502
96          DFB  $FB        ; XCE
97          DFB  $60        ; RTS
98
99  LPEEK   DFB  $18        ; CLC
100         DFB  $FB        ; XCE
101         DFB  $AF        ; LDA $000000
102 ADRESSE1 DFB 0,0
103 N_PAGE1 DFB  0
104         DFB  $80,$F5    ; BRA FIN
105
106 ANCVEC  DA   $FF58      ; RTS par défaut
107 DP_PEEK DS   1
```

]lgs

Programme LIT.MESS.GS

```
5  PRINT  CHR$ (4)"-PIC.POC
10  FOR I = 35374 TO 37261: &  PEEK 255,I,G
    %:G% = G% - 128 * (G% > 127): ON  NOT G
    % GOSUB 100: IF G% > 31 THEN  PRINT  CH
    R$ (G%);
20  NEXT
30  END
100  PRINT : RETURN
```

ProDOS

Programme GSCOLOR

Temps d'exécution : 9 mn…

```
1  PRINT  CHR$ (4)"-PIC.POC
10 U =  PEEK (49204): POKE 49204,0: REM
    Sauvegarde de la couleur des
    bords, puis mise au noir
15  POKE 49193,161
16 A$ = "E1/9 N 0<2000.9FFFZ N 5<9D60.9DC7Z
     N 0<9EC0.9EDFZ N 0/9 N D823G" : REM
    Efface l'ensemble SHGR, routine
    SHLAM
17  FOR I = 1 TO  LEN (A$): POKE 511 + I, A
    SC ( MID$ (A$,I,1)) + 128: NEXT : POKE
    72,0: CALL  - 144
20  REM   Construction des palettes
30  FOR I = 0 TO 30: &  POKE 225,40448 + I,
    240:I = I + 1: &  POKE 225,40448 + I,I
    / 2: NEXT
40  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 32,(30 - I) * 8:I = I + 1: &  POKE 22
    5,40448 + I + 32,15: NEXT
50  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 64,I / 2:I = I + 1: &  POKE 225,40448
     + I + 64,15: NEXT
60  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 96,15:I = I + 1: &  POKE 225,40448 +
    96,(31 - I) / 2: NEXT
70  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 128,15 + I * 8:I = I + 1: &  POKE 225
    ,40448 + I + 128,0: NEXT
72  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 160,240 + (30 - I) / 2:I = I + 1: &
```

```
    POKE 225,40448 + I + 128,0: NEXT
73  FOR I = 0 TO 30: &  POKE 225,40448 + I,
    240: &  POKE 225,40448 + I + 192,240:I
    = I + 1: &  POKE 225,40448 + I,I / 2: &
      POKE 225,40448 + I + 192,I / 2: NEXT
74  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 32,(30 - I) * 8: &  POKE 225,40448 +
    I + 32 + 192,(30 - I) * 8:I = I + 1: &
     POKE 225,40448 + I + 32,15: &  POKE 22
    5,40448 + I + 32 + 192,15: NEXT
75  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 64,I / 2: &  POKE 225,40448 + I + 64
    + 192,I / 2:I = I + 1: &  POKE 225,4044
    8 + I + 64,15: &  POKE 225,40448 + I +
    64 + 192,15: NEXT
76  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 96,15: &  POKE 225,40448 + I + 96 + 1
    92,15:I = I + 1: &  POKE 225,40448 + I
    + 96,(31 - I) / 2: &  POKE 225,40448 +
    I + 96 + 192,(31 - I) / 2: NEXT
77  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 128,15 + I * 8: &  POKE 225,40448 + I
     + 128 + 192,15 + I * 8:I = I + 1: &  P
    OKE 225,40448 + I + 128,0: &  POKE 225,
    40448 + I + 128 + 192,0: NEXT
78  FOR I = 0 TO 30: &  POKE 225,40448 + I
    + 160,240 + (30 - I) / 2: &  POKE 225,4
    0448 + I + 160 + 192,240 + (30 - I) / 2
    :I = I + 1: &  POKE 225,40448 + I + 128
    ,0: &  POKE 225,40448 + I + 128 + 192,0
    : NEXT
100  REM  Installation des pointeurs
110  FOR I = 40192 TO 40192 + 176 STEP 16:
    FOR J = 0 TO 15: &  POKE 225,I + J,(I -
     40192) / 16: NEXT : NEXT
200  REM  Remplissage de la page
210  FOR A = 8192 TO 8192 + 5 * 2560 STEP 2
    560
220  FOR B = 0 TO 15
230  FOR C = 0 TO 159
240 AD = A + B * 160 + C: &  POKE 225,AD,B
    + B * 16
245  &  POKE 225,AD + 15360,B + B * 16
250  NEXT : NEXT : NEXT
999  PRINT  CHR$ (7): GET R$: POKE 49193,65
1000  POKE 49204,U: REM Restaure la
    couleur des bords
```

Récapitulation PIC.POC

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : `BSAVE PIC.POC,A$300,L$A4`

```
0300:AD F6 03 C9 2A D0 07 AD
0308:F7 03 C9 03 F0 1B AD F6
0310:03 8D A1 03 AD F7 03 8D
0318:A2 03 A9 4C 8D F5 03 A9
0320:2A 8D F6 03 A9 03 8D F7
0328:03 60 C9 B9 F0 0B C9 E2
0330:D0 59 A2 FF 8E A3 03 30
0338:05 A2 00 8E A3 03 20 F5
0340:E6 8E 95 03 8E 9E 03 20
0348:BE DE 20 67 DD 20 52 E7
0350:A6 50 8E 93 03 8E 9C 03
0358:8D 94 03 8D 9D 03 2C A3
0360:03 30 09 20 F5 E6 8E 91
0368:03 4C 8E 03 20 BE DE 20
0370:E3 DF 85 83 84 84 A5 81
0378:25 82 30 03 4C 76 DD 20
0380:99 03 A0 01 91 83 88 98
0388:91 83 60 6C A1 03 18 FB
0390:A9 00 8F 00 00 00 38 FB
0398:60 18 FB AF 00 00 00 80
03A0:F5 58 FF 00
```

ProDOS

Le Basic et l'heure

En Basic sous ProDOS, il existe un moyen simple d'obtenir l'heure dans le but de mesurer un temps d'exécution par exemple. Nul besoin d'appeler une commande externe ou d'écrire la mini-routine qui appellerait GET_TIME et remettrait à l'heure les octets $BF90 à $BF93.

Il suffit d'utiliser la routine de fermeture des fichiers du Basic.Interpreter 1.1 qui se chargera du travail :

faire CLOSE ; l'octet 49043 donne alors l'heure, et le 49042 les minutes.

Par exemple, le programme :

```
10 HOME
20 PRINT CHR$(4)"CLOSE
30 VTAB 10: PRINT PEEK(4
   9043)" heures "PEEK (
   4904 2)" minutes
40 GOTO 20
```

affiche l'heure en permanence.

Bien sûr, il faut une carte horloge dans l'appareil ; celle intégrée au IIGS ne sera activée qu'avec ProDOS 1.2 et suivants.

Un conseil ?
Un problème ?
Un souhait ?...

...24 heures sur 24, laissez votre message sur CalvaCom.

Le code d'abonné de votre revue, c'est "emp11".

Graphes et fonctions

Claude-Lucien CAUET

Ce programme se compose de deux parties :

• le programme *SCHEMATEUR* permet de dessiner des figures en haute résolution (rectangles, cercles, ellipses...) avec des textes incorporés (y compris un jeu de caractères mathématiques). Par ailleurs, il est possible de remplir les surfaces avec des motifs variés, de déplacer l'image (ou une partie) dans toutes les directions : centrage, mixage, image géante de trois écrans. On peut enfin travailler simultanément avec deux images superposées, les mettre en réserve (jusqu'à cinq en mémoire auxiliaire), inverser la couleur... Bref, un programme ambitieux.

• le programme *FONCTION* permet de tracer le graphe de fonctions mathématiques de tous types, ou leur dérivée, en choisissant à loisir l'échelle et la fenêtre de visualisation. Les tracés sont continus ou pointillés, et leur création peut être accélérée. Trois écrans graphiques sont disponibles, dont deux superposables. Enfin, une fonction est acceptée même dans un domaine où elle n'est pas définie.

Ces deux programmes communiquent sans problème leurs images graphiques, qui peuvent être imprimées directement sur ImageWriter grâce à l'utilitaire *GRAVURE*.

Configuration

Un minimum de 64Ko de mémoire est nécessaire ; par ailleurs, une paire de 'paddles' (ou un joystick) est recommandée. Le programme est initialement conçu pour un Apple //e ou //c mais doit pouvoir tourner sans modification sur un][+.

L'essentiel des modules sont écrits en Applesoft et donc facilement modifiables. Plusieurs fichiers en assembleur complètent l'ensemble pour assurer les fonctions délicates ou qui nécessitent une grande vitesse.

GRAPHISME fonctionne sous DOS 3.3 ; son adaptation à ProDOS nécessiterait de revoir les fonctions de gestion des images en mémoire (actuellement conservées dans la carte langage et l'éventuelle mémoire auxiliaire de la carte 80 colonnes étendues).

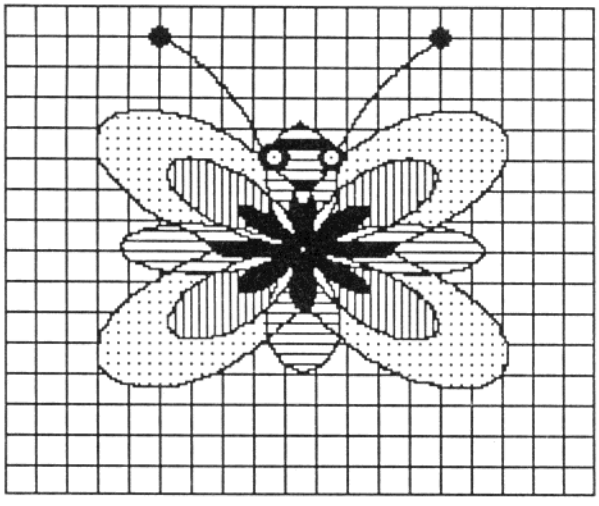
Le démarrage de GRAPHISME s'effectue par un simple :

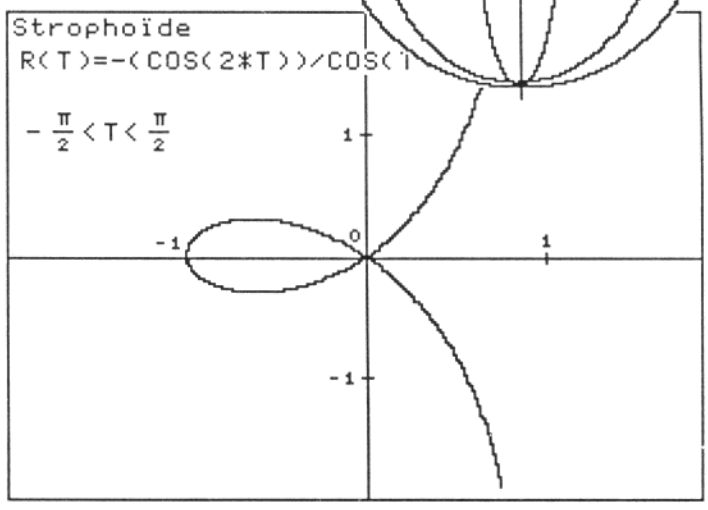
```
RUN GRAPHISME
```

qui amène sur un premier menu proposant l'accès aux deux principaux programmes, ainsi qu'une option de configuration : existence de paddles et nombre de lecteurs de disquettes.

Utilisation

En lançant le programme, on se trouve placé en mode graphique et un point lumineux apparaît au centre de l'écran.





La touche **Q** permet d'accéder au menu principal permettant la lecture/écriture d'une image graphique, le catalogue d'une disquette et de quitter SCHEMATEUR.

Ctrl-Q permet d'obtenir à l'écran un rappel des fonctions disponibles.

Déplacement du curseur

Le curseur graphique se déplace sans tracer grâce aux flèches du clavier, par saut de douze positions. En appuyant simultanément sur (ou le bouton adéquat du joystick), le déplacement s'effectue point par point, donc beaucoup plus lentement. À l'inverse, l'appui de permet de se déplacer par saut de quart d'écran.

La touche **TAB (Ctrl-I)** permet d'activer (ou de désactiver) le mode joystick. Le déplacement peut alors être effectué beaucoup plus rapidement vers la position voulue, mais avec moins de précision. Il sera donc commode d'utiliser alternativement ces deux modes.

Commandes globales

DEL efface l'écran graphique.

K définit une couleur : 0 à 7 si vous utilisez réellement la couleur, ou *N*, *B*, *G* pour noir, blanc et gris respectivement avec un moniteur monochrome.

I inverse totalement la couleur de l'écran.

W échange le dessin courant avec un dessin stocké en carte langage.

U superpose le deuxième dessin au dessin courant.

Commandes de positionnement et de tracé

X marque la position du point, comme étant le début du tracé. Par suite, le dernier point du tracé deviendra le nouveau point de départ.

Z marque de manière permanente le point courant comme étant le point de départ de tous les tracés suivants. Utile pour tracer des cercles concentriques ou des droites concourantes.

ESPACE trace un segment de droite entre le point de départ défini et le point actuel.

H trace un segment horizontal entre le point de départ et la verticale passant par le point courant.

V idem mais trace un segment vertical.

R trace un rectangle dont les points opposés sont le point de départ et le point courant.

C trace un cercle centré sur le point de départ et dont la circonférence passe par le point mobile.

D idem mais trace un disque plein.

A trace lentement un cercle dans le sens des aiguilles d'une montre, interruptible par une touche quelconque afin d'obtenir un arc de cercle. Le tracé est définitivement arrêté par RETURN.

E idem mais trace lentement une ellipse centrée sur le point de départ et dont les demi-axes sont donnés par la position du point courant.

Commandes de remplissage

S remplit une surface de forme quelconque, définie par la position du point courant et délimitée par des limites pleines (pas de pointillé). Le remplissage peut être interrompu par une touche quelconque.

P précise le mode de remplissage :
C (remplissage complet),
V (lignes verticales),
H (lignes horizontales),
P (points régulièrement espacés),
Q (quadrillage),
D (hachures penchées à droite),
G (idem à gauche),
L (losange).

Le pas peut également être défini, ainsi que l'épaisseur des traits.

L indique si le remplissage est limité par des points allumés ou éteints, ainsi que le mode de superposition (par la commande U).

On peut gommer par les touches **G** (gomme horizontale) ou **Y** (gomme verticale), jusqu'à la fin de ce mode par RETURN.

Commandes de texte

On entre en mode texte par la touche **T**, et un curseur apparaît. Celui-ci se déplace avec les flèches, ainsi qu'avec les touches .

On peut superposer deux caractères en revenant en arrière après le premier caractère (accent circonflexe, tréma).
L'appui simultané de fournit un jeu de caractères mathématiques et grecs.

DEL efface un caractère. ESC donne un curseur mort : il n'avance pas (utile pour le pivotement),

Ctrl-A agrandit le caractère, **Ctrl-Z** le réduira (s'il a été agrandit).

Enfin, **Ctrl-W** fait pivoter le caractère d'un quart de tour. On sort de ce mode texte par RETURN.

Déplacements du dessin

La touche **M** active le mode déplacement : les flèches permettent alors de déplacer le dessin, de huit points en huit points.

L'appui simultanée de la touche permet d'enrouler le dessin : horizontalement, il *roule* sur une largeur égale à deux écrans ; verticalement, sur une hauteur d'un écran et demi. Soit un total de trois écrans. La touche **W** reste bien sûr opérationnelle pour observer la seconde partie de l'image horizontale. On ressort de

ce mode par Return.

Définition de fenêtre

F permet de définir le coin d'une fenêtre dont le coin opposé a été défini par la touche **X**.

Plusieurs options sont alors possibles :

- effacer la fenêtre (touche **G**) ;
- la remplir uniformément (**R**) ;
- effacer l'extérieur (**D**) ou
- encadrer la fenêtre (**E**).

La touche **M** permet de déplacer la fenêtre en superposition sur le dessin en cours ; la barre d'espace permet ensuite de figer le résultat. Dans ce mode, il est recommandé d'utiliser la touche .

Stockage d'images en mémoire

La touche **B** échange le dessin courant avec un dessin stocké en mémoire.

Les fonctions suivantes sont disponibles si l'on dispose d'une carte 80 colonnes étendues (128Ko) :

O permet d'observer les dessins stockés (jusqu'à 5) ; Return permet de sortir de ce mode d'observation.

N échange le dessin courant avec le dernier dessin visualisé par O.

J permet de fixer le nombre d'images stockées.

Le programme FONCTION

Trois types de définitions de fonction sont disponibles :

- fonction cartésienne : y = f (x)
- fonction polaire : R = f (T) où T est la variable angulaire en radians, et R le rayon vecteur correspondant
- fonction paramétrique : x = f (t) et y = g (t) où x et y sont fonctions d'une variable t.

Il est possible d'incorporer un paramètre M dans les équations, afin de tracer une famille de courbes sans ne faisant varier que ce paramètre.

Programmation des fonctions

Les symboles suivants sont disponibles :

+, –, *, / et ^ (élévation à la puissance) ; **ABS**, **INT**, **SQR**, **LOG**, **EXP**, **SIN**, **COS**, **TAN**, **ATN** sont semblables à leur définition en Applesoft. **FNLD** (logarithme décimal), **FNSH**, **FNCH** et **FNTH** (sinus, cosinus, tangente hyperboliques) sont par ailleurs disponibles.

N'oubliez pas d'utiliser les parenthèses selon les règles habituelles de l'algèbre.

Par la suite, il est seulement nécessaire de définir les bornes du graphe (en utilisant éventuellement les valeurs par défaut proposées).

Options disponibles au menu

P permet d'obtenir la courbe en pointillé

D pour tracer la dérivée de la fonction

S évite de dessiner les axes

C encadre le graphe et RETURN visualise l'écran.

Ensuite, les touches suivantes déclenchent l'opération :

1 trace le graphe après avoir effacé l'écran,
2 évite l'effacement,
3 autorise la modification des valeurs,
4 permet de changer de fonction,
5 sauve le graphe sur disque (par exemple pour le reprendre sous SCHEMATEUR),
6 charge un graphe du disque,
7 donne le catalogue,
8 retourne au menu.

Options disponibles au graphe

Pendant le cycle de création du graphe, les touches disponibles sont :

Return interrompt le tracé,
C poursuit le tracé,
ESC saute dix points de calcul et permet ainsi un tracé plus rapide,
TAB revient en arrière de vingt points (corrige l'effet d'ESC).

Une fois le tracé terminé, on dispose de :

Return revient au menu initial,
W échange l'écran courant avec un deuxième écran mémorisé,
U superpose ces deux écrans,
B échange l'écran courant avec un écran en mémoire secondaire
T inscrit du texte sur l'écran: pratiquement toutes les sous-options disponibles avec SCHEMATEUR sont disponibles ici. De plus Ctrl-F affiche l'équation sous la position du curseur.

Le programme GRAVURE

Le programme GRAVURE permet d'imprimer, sur ImageWriter, les images graphiques créées par SCHEMATEUR ou FONCTION.

Deux hauteurs d'image sont disponibles, ainsi que l'inversion des couleurs. On peut également jouer avec un jeu de déformations horizontales et le positionnement de la feuille de papier et de la tête d'écriture.

Fichiers sur la disquette Pom's

Un grand nombre de modules forment l'ensemble de GRAPHISME.

Seuls les sources qui n'ont jamais été publiés se trouvent sur la disquette.

GRAPHISME initialisation du logiciel, gestion de la configuration

CONF.C module de configuration (manuel de la carte 80 colonnes)
HEL.C module de nettoyage de la mémoire auxiliaire
T.HEL source de HEL.C
MENU programme général de menu (ne doit pas être verrouillé car il peut s'auto-modifier)
SCHEMATEUR le programme
SHEM.C module assembleur d'édition des graphiques
T.SHEM source de SHEM.C
CARAC.SH table des jeux de caractères
CHARGE.OBJ module de chargement rapide (Pom's 4 modifié par Pom's 19)
FONCTION
FONC.C module assembleur de tracé de courbes
T.FONC source de FONC.C (adapté de G.Mauffrey dans Pom's 11)
ERREUR module classique de gestion d'erreur en Applesoft
GRAVURE
GRAV.C module d'impression
T.GRAV source de GRAV.C (adapté de l'ImageWriter ToolKit)
EXEMPLE.PIC image graphique exemple illustrant le jeu alterné de caractères

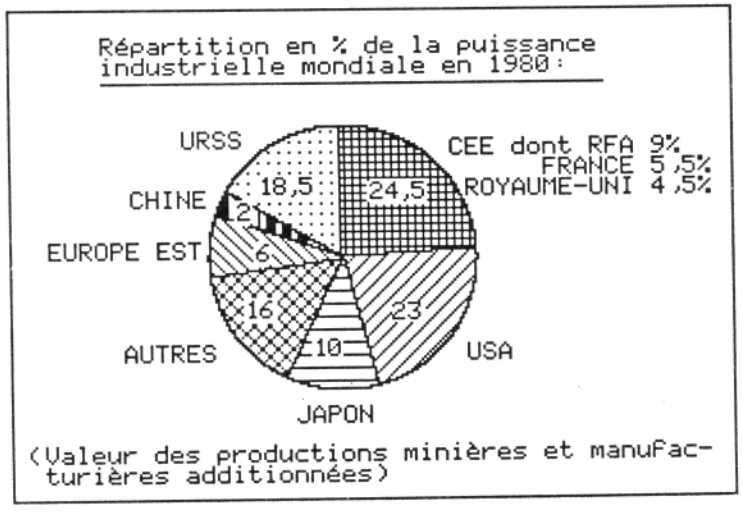
Conclusion

GRAPHISME se veut être un logiciel complet, à vocation quasi-professionnelle ; il devrait rendre de grands services à tous ceux qui manipulent fréquemment à la fois des graphiques et des fonctions mathématiques, et ont toujours regretté de ne pouvoir les manipuler simultanément.

Programme FONCTION

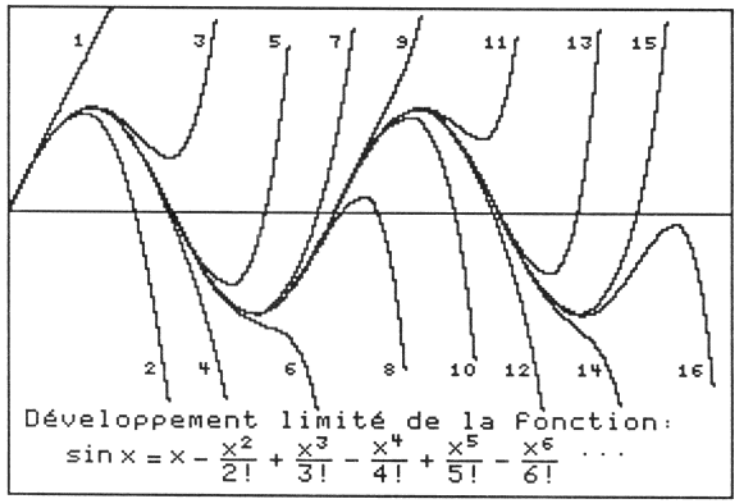
DOS 3.3
]I+
//e
//e+
//c
]Igs

```
10  DEF  FN F(T) = X: REM  ^ 2: REM 1600
    - X ^ 2): REM  REM ***********************
    ************************************
    ************************************
    ************************************
    ************************************
    ********************************
20  DEF  FN G(T) = T: REM : REM ): REM T
    : REM  ^ 2): REM  * T: REM 2: REM ***
    ************************************
    ************************************
    ************************************
    ************************************
    ********************************
30  FOR I = 1 TO 3: READ IN$(I): NEXT
40 D$ =  CHR$ (4)
50  IF  PEEK (103) + 256 *  PEEK (104) = 24577 GOTO 80
60  POKE 103,1: POKE 104,96: POKE 24576,0
70  PRINT D$"RUN FONCTION"
80  IF  PEEK (236) GOTO 140
90 AL = 6000: POKE 233, INT (AL / 256): POKE 232,AL -
   256 * ( INT (AL / 256))
100  PRINT D$"BLOAD FONC.C"
110  PRINT D$"BLOAD CARAC.SH,A"AL
120  PRINT D$"BLOAD ERREUR,A931"
130  POKE 236,1
140 A1 = 776:A2 = 769:A3 = 921:A4 = 853:A5 = 878
150 SC = 1: SCALE= SC:RO = 0: ROT= RO: HCOLOR= 3
160 VP = 7:VN = 10:VF = 12:VM = 17:VO = 18:NP = 280:X1 =  - 14:X
   2 = 14:Y1 =  - 9.6:Y2 = 9.6:TD =  - 10:TF = 10
170 S = 0:AX = 1:IM = 0:CO = 0:MM = 0:M = 1:WW = 0:NL = 1
180 PL = 0: IF  PEEK (214) THEN PL = 1
190  ONERR  GOTO 2470
200  GOSUB 2000: GOTO 1010
210  REM ** CALCUL DES COORDONNEES**
220 T4 = T: GOTO 250
230 W = W + 1: IF W > 2 * WW + 1 THEN W = 0:WW = 1 - WW
240  IF WW = 0 THEN T = T + INC: POP : GOTO 1610
250  IF  PEEK (49152) > 127 THEN SS =  PEEK (49152) - 128: POKE
   49168,0: GOTO 400
260  ON REP GOTO 270,320,360
270 X = X1 + (T + 1) / EX
280 Y =  FN F(X)
290  IF  NOT DE GOTO 310
300 X = X + (X2 - X1) / 10000:YA =  FN F(X):Y = (YA - Y) * 10000
    / (X2 - X1)
310 I = T:J = (Y - Y1) * EY - 1: RETURN
320 Z =  FN F(T)
330  IF DE THEN T = T + (T2 - T1) / 10000:ZA =  FN F(T):Z = (ZA
   - Z) * 10000 / (T2 - T1)
340 X = Z *  COS (T):I = EX * (X - X1): IF I < 0 OR I > 280 THEN
     POP : GOTO 1630
350 Y = Z *  SIN (T):J = EY * (Y - Y1): RETURN
360 X =  FN F(T):Y =  FN G(T)
370  IF DE THEN T = T + (T2 - T1) / 10000:XA =  FN F(T):YA =  FN
    G(T):Y = (YA - Y) / (XA - X)
380 I = EX * (X - X1): IF I < 0 OR I > 280 THEN  POP : GOTO 1630
390 J = EY * (Y - Y1) - 1: RETURN
400  IF SS = 13 THEN T5 = T - INC: POP :: GOTO 1700
410  IF SS = 27 THEN T = T + 10 * INC: GOTO 430
420  IF SS = 9 THEN T = T - 20 * INC
430  IF T > T2 THEN T = T2
440  IF T < T1 THEN T = T1
450  POP : GOTO 1610
```




```
460  REM  ** SAISIE DES DONNEES **
470  VTAB VT: HTAB HT: INPUT "";V$: IF V$ = "" THEN  VTAB VT: HT
   AB HT: PRINT V: RETURN
480 V =  VAL (V$): RETURN
490  REM  ** TABLEAU DONNEES **
500  VTAB VP + 1: IF REP = 2 THEN  PRINT "ANGLE ->": GOTO 520
510  PRINT "PARAMETRE ->"
520  VTAB VP + 1: HTAB 14: PRINT "T minima:"TD: HTAB 14: PRINT "
   T maxima:"TF
530  VTAB VN: PRINT "NOMBRE DE POINTS:"NP
540  VTAB VF: PRINT "================FENETRE================";
    VTAB VM: CALL  - 958: FOR K = 1 TO 40: HTAB K: PRINT "=";: N
    EXT
550  VTAB VF + 1: PRINT "X minima:"X1;: HTAB 20: PRINT "Y minima
   :"Y1: PRINT "X maxima:"X2;: HTAB 20: PRINT "Y maxima:"Y2
560  RETURN
570  REM  ** ENTREE DONNEES **
580 VT = VP + 1:HT = 23:V = TD: GOSUB 470:TD = V:VT = VP + 2:HT
   = 23:V = TF: GOSUB 470:TF = V:T1 = TD:T2 = TF
590 VT = VN:HT = 18:V = NP: GOSUB 470:NP = V: IF REP = 1 THEN IN
   C = 280 / NP: GOTO 610
600 INC = (T2 - T1) / NP
610  POKE 33,19:VT = VF + 1:HT = 10:V = X1: GOSUB 470:X1 = V:VT
   = VF + 2:HT = 10:V = X2: GOSUB 470:X2 = V:CX = X2 - X1: IF CX
    <  = 0 GOTO 610
620 EX = 279 / CX:UX = 1: POKE 33,40
630  IF CX > 40 THEN CX = CX / 10:UX = UX * 10: GOTO 630
640  IF CX < 4 THEN CX = CX * 10:UX = UX / 10: GOTO 640
650 VT = VF + 1:HT = 29:V = Y1: GOSUB 470:Y1 = V:YM = Y1 + (X2 -
    X1) * 192 / 280: VTAB VF + 4: HTAB 1: CALL  - 868: VTAB VF +
    3: CALL  - 868: PRINT "Pour la meme échelle:Y maxima=";0.1 *
     INT (10 * YM + 0.5)
660 VT = VF + 2:HT = 29:V = Y2: GOSUB 470:Y2 = V:CY = Y2 - Y1: I
   F CY <  = 0 GOTO 650
670 EY = 191 / CY:UY = 1
680  IF CY > 40 THEN CY = CY / 10:UY = UY * 10: GOTO 680
690  IF CY < 4 THEN CY = CY * 10:UY = UY / 10: GOTO 690
700  HTAB 1: VTAB VF + 3: CALL  - 868: PRINT "Unité en abscisse:
   ";UX: PRINT "Unité en ordonnée:";UY
710  RETURN
```

```
720  REM  ** TABLEAU OPTIONS **
730  VTAB VM: CALL  - 958: FOR K = 1 TO 40: HTAB K: PRIN
   T "=";: NEXT
740  VTAB VO: PRINT "1- EFFACER ET TRACER": PRINT "2- TR
   ACER SANS EFFACER": PRINT "3- MODIFIER LES VALEURS"
750  VTAB 21: HTAB 1: PRINT "4- CHANGER DE FONCTION": PR
   INT "5- ENREGISTRER       6- CHARGER": PRINT "7- IMPRI
   MER          8- CATALOGUE": PRINT "9- QUITTER LE PROGR
   AMME";
760  RETURN
770  REM  ** PARAMETRE **
780  VTAB VP: PRINT "SI VOUS VOULEZ UTILISER UN PARAMETR
   E M  TAPEZ M (SINON UNE AUTRE TOUCHE):";: GET PR$: IF
    PR$ <  > "M" AND PR$ <  > "m" THEN MM = 0: GOTO 795
790 MM = 1
795  VTAB VP: HTAB 1: CALL  - 958: RETURN
800  GOSUB 855: VTAB VP: HTAB 1: PRINT "1- FAMILLE DE CO
   URBES": PRINT "2- PARAMETRE A LA DEMANDE": PRINT "CHO
   IX:";: GET PR$:PR =  VAL (PR$) - 1: IF PR < 0 OR PR >
    1 GOTO 800
810  IF PR = 0 GOTO 830
820  VTAB VP: HTAB 1: CALL  - 958: INPUT "PREMIERE VALEU
   R DE 'M':";M: GOTO 850
830  GOSUB 855: VTAB VP: HTAB 1: PRINT "M minima:"M1: VT
   AB VP: HTAB 16: PRINT "M maxima:"M2: VTAB VP: HTAB 31
   : PRINT "Pas:"PM
840  POKE 33,15:VT = VP:HT = 10:V = M1: GOSUB 470:M1 = V: POKE 3
   3,30:VT = VP:HT = 25:V = M2: GOSUB 470:M2 = V: POKE 33,40:VT
   = VP:HT = 35:V = PM: GOSUB 470:PM = V
850  RETURN
855  FOR II = VP TO VP + 3: VTAB II: HTAB 1: CALL  - 868: NEXT :
    RETURN
860  REM  ** MESSAGE DEFILANT **
870  SPEED= 245:AX = 1:CA = 0:PO = 0:DE = 0:DF = 1:T$ = A$(DF):N
   $ = "": FOR J = 1 TO 20:N$ = N$ + " ": NEXT
880 M$ = N$ + T$ + " "
890  INVERSE : FOR K = 1 TO  LEN (M$): VTAB VM: HTAB 5: PRINT  M
   ID$ (M$,K,30)
900 TC =  PEEK (49152): IF TC > 127 THEN  POKE 49168,0: SPEED= 2
   55: NORMAL : GOTO 930
910  NEXT :DF = 1 + DF: IF DF = DM + 1 THEN DF = 1
920 T$ = A$(DF): GOTO 880
930  IF TC = 196 THEN DE = 1: GOTO 890
940  IF TC = 208 THEN PO = 1: GOTO 890
950  IF TC = 211 THEN AX = 0: GOTO 890
955  IF TC = 195 THEN CA = 1
960  IF TC = 141 THEN  POKE 49232,0: POKE 49234,0: POKE 49236,0:
    POKE 49239,0: POP : GOTO 1760
970  IF TC > 176 AND TC < 186 THEN R = TC - 176: GOTO 990
980  GOTO 890
990  FOR K = 1 TO 40: VTAB VM: HTAB K: PRINT "=": NEXT : RETURN
1000  REM  ** DEBUT ** MENU **
1010  HOME : VTAB 2: HTAB 4: PRINT "GRAPHES DE FONCTIONS": VTAB
   3: HTAB 4: FOR I = 1 TO 18: PRINT "_";: NEXT : PRINT "_"
1020  FOR I = 1 TO 3: GOSUB 1100: NEXT
1030 I = 1: INVERSE : GOSUB 1100: NORMAL
1040  VTAB 20: PRINT "FLECHES -> CHOISIR   RETURN -> VALIDER"
1050  VTAB 21: GET CAR$:CH =  ASC (CAR$)
1060  IF CH = 10 OR CH = 8 THEN  GOSUB 1100:I = I + 1: INVERSE :
    GOSUB 1100: NORMAL
1070  IF CH = 11 OR CH = 21 THEN  GOSUB 1100:I = I - 1: INVERSE
   : GOSUB 1100: NORMAL
1080  IF CH = 13 THEN  HOME : GOTO 1130
1090  GOTO 1050
1100  IF I = 4 THEN I = 1
1110  IF I = 0 THEN I = 3
1120  VTAB I + 6: HTAB 4: PRINT IN$(I): RETURN
1130 REP = I: ON I GOTO 1150,1180,1210
1140  REM  ** ENTREE DES FONCTIONS **
1150  HOME : HTAB 9: INVERSE : PRINT " FONCTION CARTESIENNE": NO
   RMAL : VTAB 3: PRINT "Y(X)=": GOSUB 780
1160  HTAB 1: VTAB 3: PRINT "Y(X)=";: CALL A1: IF MM THEN  GOSUB
    800
1170  GOSUB 530: GOSUB 590: GOTO 1240
1180  HOME : HTAB 12: INVERSE : PRINT " FONCTION POLAIRE ": NORM
   AL : PRINT "(La variable T est l'angle en radians)": VTAB 3:
   PRINT "R(T)=": GOSUB 780
1190  HTAB 1: VTAB 3: PRINT "R(T)=";: CALL A1: IF MM THEN  GOSUB
    800
1200  GOTO 1230
1210  INVERSE : HOME : HTAB 9: PRINT " FONCTION PARAMETRIQUE ":
   NORMAL : PRINT "(La variable T est le paramètre courant)": VT
   AB 3: PRINT "X(T)=": VTAB 5: PRINT "Y(T)=": GOSUB 780: HTAB 1
   : VTAB 3: PRINT "X(T)=";: CALL A1
1220  VTAB 5: PRINT "Y(T)=";: CALL A2: IF MM THEN  GOSUB 800
1230  GOSUB 500: GOSUB 580
1240  GOSUB 730
1250  GOSUB 870: ON R GOTO 1350,1260,1265,1300,1890,1890,1290,19
   50,1320
1260  POKE 49232,0: POKE 49234,0: POKE 49236,0: POKE 49239,0: GO
   TO 1360
1265  IF MM = 0 GOTO 1270
1266  GOSUB 800: IF REP = 1 GOTO 1170
1267  GOTO 1230
```



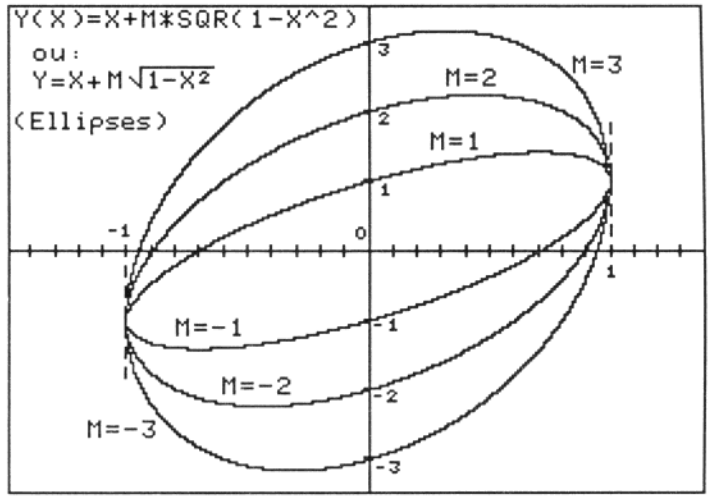

```
1270  IF REP = 1 THEN  GOSUB 590: GOTO 1240
1280  GOSUB 580: GOTO 1240
1290  PRINT D$"RUN GRAVURE,D1"
1300  HOME : GOTO 1010
1310  REM  ** FIN **
1320  HOME : PRINT
1330  PRINT D$"RUN MENU,D1"
1340  REM  ** DEBUT DU GRAPHE **
1350  HGR : HCOLOR= 3: POKE 49234,0
1360  IF AX = 0 GOTO 1565
1370  REM **TRACE DES AXES**
1390 I0 =  INT ( - X1 * EX)
1400 J0 =  INT ( - Y1 * EY)
1410  IF J0 < 0 OR J0 > 191 THEN 1490
1420  HPLOT 0,191 - J0 TO 279,191 - J0
1430  IF J0 < 2 THEN C = J0:D = 2: GOTO 1460
1440  IF J0 > 189 THEN C = 2:D = 191 - J0: GOTO 1460
1450 C = 2:D = 2
1460  FOR I =  INT (X1 / UX) TO  INT (X2 / UX): IF I0 + I * EX *
    UX < 0 OR I0 + I * EX * UX > 279 THEN 1480
1470  HPLOT I0 + I * EX * UX,191 + C - J0 TO I0 + I * EX * UX,19
   1 - D - J0
1480  NEXT
1490  IF I0 < 0 OR I0 > 279 THEN 1570
1500  HPLOT I0,0 TO I0,191
1510  IF I0 < 2 THEN A = I0:B = 2: GOTO 1540
1520  IF I0 > 277 THEN A = 2:B = 279 - I0: GOTO 1540
1530 A = 2:B = 2
1540  FOR J =  INT (Y1 / UY) TO  INT (Y2 / UY): IF J0 + J * EY *
    UY > 191 OR J0 + J * EY * UY < 0 THEN 1560
1550  HPLOT I0 - A,191 - J0 - J * EY * UY TO I0 + B,191 - J0 - J
   * EY * UY
1560  NEXT
1565  IF CA = 1 THEN  HPLOT 0,0 TO 279,0 TO 279,191 TO 0,191 TO
   0,0
1570  REM  **TRACE DE LA FONCTION**
1580  IF MM = 1 AND PR = 0 THEN M = M1
1590  IF REP = 1 THEN T1 = 1:T2 = 279
1600 T = T1
1610  GOSUB 220
1620  IF J >  = 0 AND J <  = 191 THEN  HPLOT I,191 - J: GOTO 169
   0
1630  XDRAW 98 AT 10,10: XDRAW 98 AT 10,10
1640 T = T + INC: IF T < T2 GOTO 1610
1650  GOTO 1700
1660  GOSUB 220
1670  IF J < 0 OR J > 191 GOTO 1640
1675  IF PO THEN  HPLOT I,191 - J: GOTO 1690
1680  HPLOT  TO I,191 - J
1690 T = T + INC: IF T < T2 GOTO 1660
1700  IF MM = 0 GOTO 1750
1710  IF  NOT PR GOTO 1740
1720  POKE 49235,0: VTAB 21: HTAB 1: CALL  - 958: VTAB 22: INPUT
    "NOUVEAU PARAMETRE(SINON RETURN):";PR$: IF PR$ = "" THEN  GO
   SUB 750: POKE 49234,0: GOTO 1750
1730 M =  VAL (PR$): GOSUB 750: POKE 49234,0:T1 = TD: GOTO 1590
1740  IF T >  = T2 THEN T1 = TD:M = M + PM: IF M =  < M2 GOTO 15
   90
1750  PRINT  CHR$ (7);
1760  IF  PEEK (49152) < 128 GOTO 1760
1770 RR =  PEEK (49152) - 128: POKE 49168,0: IF RR = 87 THEN  CA
   LL A3: GOTO 1760
1780  IF RR = 85 THEN  CALL A4: GOTO 1760
1790  IF RR = 66 THEN  CALL A5: GOTO 1760
1800  IF RR = 84 THEN  GOSUB 2190: GOTO 1760
1810  IF RR = 13 THEN  POKE 49233,0: GOTO 1250
1820  IF RR = 67 THEN T = T5: GOTO 1610
1830  GOTO 1760
```

```
1840  REM  ** ENREGISTRER-CHARGER **
1850  IF  NOT PL THEN  RETURN
1860  VTAB VO: HTAB 1: CALL  - 958: PRIN
   T : PRINT "LECTEUR N°:";: INVERSE : P
   RINT NL;: NORMAL : CALL  - 1008: GET
   CAR$: IF CAR$ =  CHR$ (13) THEN  RETU
   RN
1870  IF CAR$ <  > "1" AND CAR$ <  > "2"
    GOTO 1850
1880 NL =  VAL (CAR$): INVERSE : PRINT N
   L: NORMAL : RETURN
1890  GOSUB 1850
1900  HTAB 1: VTAB VO + 3: INPUT "NOM DE
    L'IMAGE:";IM$: IF IM$ = "" GOTO 1940
1910 PG =  ASC ( LEFT$ (IM$,1)): IF PG <
    65 OR PG > 90 GOTO 1900
1920  IF R = 5 THEN  PRINT D$"BSAVE"IM$"
   ,A$2000,L$2000,D";NL: GOTO 1940
1930  PRINT D$"BLOAD"IM$",A$2000,D"NL
1940  GOTO 1240
1950  POKE 34,VP - 1: HOME : GOSUB 1850:
    PRINT
1960  PRINT D$"CATALOG,D"NL: INVERSE : P
   RINT "FIN": NORMAL
1970  GET CAR$: HOME : POKE 34,0: IF REP
    = 1 THEN  GOSUB 530: GOTO 1240
1980  GOSUB 500: GOTO 1240
1990  REM **FONCTIONS SPECIALES**
2000  DEF  FN SH(X) = ( EXP (X) -  EXP (
   - X)) / 2
2010  DEF  FN CH(X) = ( EXP (X) +  EXP (
   - X)) / 2
2020  DEF  FN TH(X) =  -  EXP ( - X) /
   EXP (X) +  EXP ( - X) * 2 + 1
2030  DEF  FN LD(X) =  LOG (X) /  LOG (1
   0)
2040 DM = 3: REM   **NB DE MESSAGES **
2050 A$(1) = "TAPEZ VOTRE OPTION DE 1 A
   9...'P' POUR COURBE EN POINTILLE...'D
   ' POUR DERIVEE...'S' SANS AXES...'C'
   CADRE...'RETURN' VOIR L'ECRAN"
2060 A$(2) = "UNE SONNETTE ANNONCERA LA
    FIN DU GRAPHE...ON PEUT L'INTERROMPRE
    PAR 'RETURN' ET LE CONTINUER PAR 'C'
   ...'ESC' SAUTE 10 POINTS...'TAB' REVI
   ENT DE 20 POINTS"
2070 A$(3) = "QUAND LE GRAPHE EST INTERR
   OMPU  ' T ' MODE TEXTE...' W ' PASSE
   D'UN ECRAN A UN AUTRE...'U' FUSIONNE
   LES DEUX ECRANS...'B' BANQUE ...'RETU
   RN' RAMENE A CETTE PAGE"
2080 PI = 3.1416:E =  EXP (1)
2090  RETURN
2100  REM  ** TEXTE **
2110 VX = 7:VY = 8
2120  IF  PEEK (49250) > 127 THEN VX = 1
   :VY = 1
2130  IF  PEEK (49249) > 127 THEN VX = 7
   0:VY = 48
2140  IF R = 21 THEN X% = X% + VX: IF X%
    > 279 THEN X% = 279
2150  IF R = 8 THEN X% = X% - VX: IF X%
   < 0 THEN X% = 0
2160  IF R = 10 THEN Y% = Y% + VY: IF Y%
    > 191 THEN Y% = 191
2170  IF R = 11 THEN Y% = Y% - VY: IF Y%
   < 0 THEN Y% = 0
2180  RETURN
2190 PT = 1:CT = 98:KK = 0:CR = CT:S = C
   T:X% = 5:Y% = 5: XDRAW CT AT X%,Y%
2200 C =  PEEK (49152) - 128: IF C < 0 G
   OTO 2200
2210  POKE 49168,0:S = C - 29: IF C = 27
    THEN TT = 1 - TT: XDRAW CT AT X%,Y%:
   CT = 34 * (TT = 1) + CR * (TT = 0): X
   DRAW CT AT X%,Y%: GOTO 2200
2230  IF C <  > 32 OR TT = 0 GOTO 2250
2240 KK = 1: GOTO 2200
2250  IF C < 32 OR C = 127 GOTO 2330
2260  IF  PEEK (49249) > 127 AND C > 47
    AND C < 91 THEN S = S + 80
2270  XDRAW CT AT X%,Y%: DRAW S AT X%,Y%
2280  IF TT = 0 THEN CT = CR:KK = 0: GOT
   O 2310
2290 CT = S
2300  GOTO 2200
2310 X% = X% + 7
2320  XDRAW CT AT X%,Y%: GOTO 2200
2330  IF KK = 1 THEN KK = 0: DRAW CT AT
   X%,Y%: GOTO 2350
2340  XDRAW CT AT X%,Y%
2350  IF C = 13 THEN RO = 0: ROT= RO:CT
   = CR:TT = 0:KK = 0: RETURN
2360  IF C = 23 THEN RO = (RO = 56) * 0
   + (RO < 56) * (RO + 8): ROT= RO
2370  IF C = 127 THEN  DRAW CT AT X%,Y%:
    XDRAW CT AT X%,Y%
2380  IF C <  > 6 GOTO 2450
2390 LD = 1280: GOSUB 2410: IF REP = 3 T
   HEN LD = 1536:X% = X% - 7 * FF:Y% = Y
   % + 8: GOSUB 2410
2400  GOTO 2450
2410 FF = 0
2420 S =  PEEK (LD + FF) - 157: IF S = 3
    THEN  RETURN
2430  XDRAW S AT X%,Y%:X% = X% + 7:FF =
   FF + 1: IF X% > 277 THEN  RETURN
2440  GOTO 2420
2450 R = C: GOSUB 2110: GOTO 2320
2460  REM  ** TRAITEMENT D'ERREUR **
2470  CALL 931
2480  POP
2490  XDRAW 34 AT 10,10
2500 ER =  PEEK (222)
2510  IF ER <  > 53 AND ER <  > 133 GOTO
    2540
2520  XDRAW 34 AT 10,10
2530 T = T4: GOTO 1640
2540  IF ER = 6 THEN  PRINT : PRINT "PAS
    D'IMAGE DE CE NOM (RETURN)": GET CAR
   $
2550  IF ER = 16 THEN  POKE 49233,0: VTA
   B VM + 1: CALL  - 958: PRINT "ERREUR
   DE SYNTAXE": PRINT "VERIFIER L'ECRITU
   RE DE LA FONCTION": PRINT "RETURN";:
   GET CAR$
2560  GOTO 1240
2570  DATA  FONCTION CARTESIENNE,FONCTIO
   N POLAIRE,FONCTION PARAMETRIQUE
```

Programme SCHEMATEUR

```
5  HIMEM: 36863
10  REM  ** SCHEMATEUR **
20  REM  ** M.CAUET - NOV 86 **
30  IF  PEEK (103) + 256 *  PEEK (104) =
    24577 GOTO 2240
40  REM  ** IMPLANTATION DU PROGRAMME AU
   -DESSUS DE HGR2 **
50  POKE 103,1: POKE 104,96: POKE 24576,
   0
60  PRINT  CHR$ (4)"RUN SCHEMATEUR"
70  REM  ** COORDONNEES DU POINT **
80  IF ME AND R = 9 THEN MN = 1 - MN
90  IF MN = 0 GOTO 110
100 X% =  PDL (0) / 255 * 279:Y% =  PDL
   (1) / 255 * 191: RETURN
110 VX = 12:VY = 12
120  IF  PEEK (49250) > 127 THEN VX = 1:
   VY = 1
130  IF  PEEK (49249) > 127 THEN VX = 70
   :VY = 48
140  IF R = 21 THEN X% = X% + VX: IF X%
   > 279 THEN X% = 279
150  IF R = 8 THEN X% = X% - VX: IF X% <
    0 THEN X% = 0
160  IF R = 10 THEN Y% = Y% + VY: IF Y%
   > 191 THEN Y% = 191
170  IF R = 11 THEN Y% = Y% - VY: IF Y%
   < 0 THEN Y% = 0
180  RETURN
190  REM  ** ANALYSE DE LA COMMANDE **
200  IF DP AND  NOT DH THEN X1% = X%:Y1%
   = Y%
210 DH = 0
220  IF  PEEK (49152) > 127 THEN R =  PE
   EK (49152) - 128: POKE 49168,0: IF MN
    OR  NOT MN AND (R > 31 OR R = 13 OR
   R = 17) GOTO 250
230  GOSUB 80: IF  NOT PT THEN  XDRAW CU
    AT X0%,Y0%
240 X0% = X%:Y0% = Y%: XDRAW CU AT X0%,Y
   0%:PT = 0:R = 0: GOTO 220
250  FOR I = 1 TO ZC: IF R = R(I) THEN
   ON I GOSUB 290,310,1030,350,400,420,4
   40,490,440,550,570,690,950,2160,1070,
   1210,1250,1580,1580,1580,1580,1920,19
   20,1090,1130,980,810,1150,510: GOTO 2
   00
260  NEXT : IF R > 96 AND R < 123 THEN
   POKE 49235,0: HOME : VTAB 22: PRINT "
   PASSEZ EN MAJUSCULES S.V.P. ET RETURN
   .": GET CAR$: POKE 49234,0:R = R - 32
   : GOTO 250
270  GOTO 200
280  REM  ** Q **
290  POP : RETURN
300  REM  ** DEL **
310  POKE 49235,0: HOME : VTAB 23: PRINT
    "PERTE DU DESSIN (O/N)?:";: GET CAR$
   : IF CAR$ <  > "O" AND CAR$ <  > "o"
   GOTO 330
320  HGR :PT = 1:B = 0:CL = 3: HCOLOR= C
   L: POKE AA + 8,B
330  POKE 49234,0: RETURN
340  REM  ** L **
350  POKE 49235,0: HOME : VTAB 21: PRINT
    "LIMITE ET FUSION PAR POINTS ";: IF
   B = 0 THEN  PRINT "ALLUMES.": GOTO 37
   0
360  PRINT "ETEINTS."
370  PRINT "TAPEZ C POUR CHANGER": GET C
   AR$: IF CAR$ = "C" THEN B = 1 - B: PO
   KE AA + 8,B
380  POKE 49234,0: RETURN
390  REM  ** Z **
400  GOSUB 420:DP = 0: RETURN
410  REM  ** X **
420  HPLOT X0%,Y0%:X1% = X0%:Y1% = Y0%:D
   P = 1:PT = 1: RETURN
430  REM  ** H-V **
440  IF R = 86 GOTO 460
450  HPLOT X%,Y1% TO X1%,Y1%:X1% = X%: G
   OTO 470
460  HPLOT X1%,Y% TO X1%,Y1%:Y1% = Y%
470 DH = 1: RETURN
480  REM  ** ESPACE **
490  HPLOT X%,Y% TO X1%,Y1%:PT = 1: RETU
   RN
500  REM  ** M **
510  XDRAW CU AT X%,Y%
520  CALL AG
530  XDRAW CU AT X%,Y%: RETURN
540  REM  ** R **
550  HPLOT X%,Y% TO X1%,Y% TO X1%,Y1% TO
    X%,Y1% TO X%,Y%:PT = 1: RETURN
560  REM  ** F **
570 PT = 1: POKE 49235,0: HOME : VTAB 22
   : PRINT "          G- GOMME    R- REM
   PLIT": PRINT "          D- DECOUPE  E
   - ENCADRE": PRINT "          M- MOBIL
   E";: VTAB 21: HTAB 1: PRINT "FENETRE:
    CHOIX:";: GET CAR$
580  POKE 49234,0: IF CAR$ =  CHR$ (13)
   THEN  RETURN
590 ST =  - CL + 3 + 8 * (CL > 3): IF CA
   R$ = "G" OR CAR$ = "D" OR CAR$ = "M"
   THEN  HCOLOR= ST
600  IF CAR$ = "G" OR CAR$ = "R" THEN  G
   OSUB 670: HCOLOR= CL: RETURN
610  IF CAR$ = "D" OR CAR$ = "E" THEN  G
   OSUB 640: HCOLOR= CL: RETURN
620  IF CAR$ = "M" THEN  POKE 49235,0: H
   OME : VTAB 23: PRINT "UN INSTANT S.V.
   P.": CALL AP: GOSUB 640: CALL AE: GOS
   UB 670: HCOLOR= CL: POKE 49234,0: CAL
   L AV: RETURN
630  GOTO 570
640  IF X1% > X% THEN WW = X1%:X1% = X%:
   X% = WW
650  IF Y1% > Y% THEN WW = Y1%:Y1% = Y%:
   Y% = WW
660  FOR I = 0 TO X1%: HPLOT I,0 TO I,19
   1: NEXT : FOR I = X1% TO X%: HPLOT I,
   0 TO I,Y1%: HPLOT I,Y% TO I,191: NEXT
    : FOR I = X% TO 279: HPLOT I,0 TO I,
   191: NEXT : RETURN
670  FOR I = Y% TO Y1% STEP 2 * (Y1% > Y
   %) - 1: HPLOT X%,I TO X1%,I: NEXT :PT
    = 1: RETURN
680  REM  ** K **
690 CC = 78 * (CL = 0) + 66 * (CL = 3) +
    71 * (CL = 1):CL$ =  CHR$ (CC): IF C
   C = 0 THEN CL$ =  STR$ (CL)
700  POKE 49235,0: HOME : VTAB 21: PRINT
    "COULEUR (0à7,ou B blanc,N noir,G gr
   is):"CL$;: CALL  - 1008: GET CL$
710 CB = (CL$ = "N") + 2 * (CL$ = "G") +
    4 * (CL$ = "B") - 1: IF CB =  - 1 AN
   D CL$ > "/" AND CL$ < "8" THEN CB =
   VAL (CL$)
720  IF CB =  - 1 GOTO 740
730 CL = CB
740  HCOLOR= CL: POKE 49234,0: RETURN
750  REM  ** P **
760  INVERSE : PRINT DI;: FOR I = 1 TO
   LEN ( STR$ (DI)): CALL  - 1008: NEXT
   : GET R1$: IF R1$ =  CHR$ (17) THEN C
   H = 17: GOTO 870
```

```
770  IF R1$ < "0" OR R1$ > "9" THEN DF = DI: NORMAL : PRINT
     "": RETURN
780  PRINT R1$;:D1 =  VAL (R1$): GET R2$: IF R2$ =  CHR$ (1
     7) THEN CH = 17: GOTO 870
790  IF R2$ < "0" OR R2$ > "9" THEN  PRINT " ":DF = D1: NOR
     MAL : RETURN
800  PRINT R2$:DF = D1 * 10 +  VAL (R2$): NORMAL : RETURN
810  POKE 49235,0: HOME : VTAB 21: PRINT "MODE: C V H P Q G
     D L   (AIDE: 'Ctrl Q')":I = MD
815  HTAB 1: VTAB 23: PRINT "FLECHES -> CHOISIR   RETURN ->
     VALIDER"
820  IF I = 8 THEN I = 0
830  IF I =  - 1 THEN I = 7
840  VTAB 21: HTAB 7 + 2 * I: GET CAR$:CH =  ASC (CAR$)
850  IF CH = 21 OR CH = 10 THEN I = I + 1: GOTO 820
860  IF CH = 8 OR CH = 11 THEN I = I - 1: GOTO 830
870  IF CH = 17 THEN  TEXT : HOME : NORMAL : PRINT IN$(24):
     PRINT : FOR I = 25 TO 32: PRINT IN$(I): NEXT : PRINT :
    PRINT IN$(33): PRINT : PRINT IN$(34): GET CAR$: POKE 492
     32,0: GOTO 810
875  IF CH <  > 13 GOTO 840
880 MD = I: POKE AA + 1,MD
890  IF I = 0 GOTO 930
900  HTAB 1: VTAB 22: CALL  - 958: PRINT "PAS:";:DI = PA: G
    OSUB 760:PA = DF: IF PA <  = 0 GOTO 900
910  POKE AA + 2,PA
920  PRINT "EPAISSEUR:";:DI = EP: GOSUB 760:EP = DF: POKE AA,EP
     - 1
930  POKE 49234,0: RETURN
940  REM  ** O **
950  IF MA * NI = 0 THEN  RETURN
960  CALL AO: RETURN
970  REM  ** B **
980  HOME : IF MA = 0 THEN  XDRAW CU AT X%,Y%: CALL AE:PT = 1: R
    ETURN
990  IF NI > 4 THEN  POKE 49235,0: HOME : VTAB 21: PRINT "SATURA
    TION": GET CAR$: GOTO 1010
1000  XDRAW CU AT X%,Y%:NI = NI + 1: POKE AA + 3,NI: POKE AA + 1
     5,NI * 32: CALL AB:PT = 1:B = 0:CL = 3: HCOLOR= CL: POKE AA +
      8,B: HGR
1010  POKE 49234,0: RETURN
1020  REM   ** N **
1030  IF MA = 0 THEN  RETURN
1040 NU =  PEEK (AA + 16): XDRAW CU AT X%,Y%: POKE AA + 15,NU *
     32: CALL AB:PT = 1:B = 0:CL = 3: HCOLOR= CL: POKE AA + 8,B: C
    ALL AE
1050  POKE 49234,0: RETURN
1060  REM    ** W **
1070  XDRAW CU AT X%,Y%: CALL AF:PT = 1: RETURN
1080  REM  ** U **
1090  POKE 49235,0: HOME : VTAB 23: PRINT "SUPERPOSITION CONFIRM
    EE? (O/N)";: GET CAR$: IF CAR$ <  > "O" AND CAR$ <  > "o" GOT
    O 1110
1100  XDRAW CU AT X%,Y%: CALL AU
1110  POKE 49234,0:PT = 1: RETURN
1120  REM  ** I **
1130  CALL AI:CL =  - CL + 3 + 8 * (CL > 3):B = 1 - B: HCOLOR= C
    L: POKE AA + 8,B: RETURN
1140  REM  ** J **
1150  IF MA = 0 THEN  RETURN
1160  POKE 49235,0: HOME : VTAB 23: PRINT "NOMBRE D'IMAGES EN BA
    NQUE: "NI;: CALL  - 1008: GET NI$: IF NI$ < "0" OR NI$ > "5"
     GOTO 1190
1170  PRINT NI$:NI =  VAL (NI$)
1180  POKE AA + 3,NI
1190  POKE 49234,0: RETURN
1200  REM  ** S **
1210 XH =  INT (X% / 256):XB = X% - XH * 256
1220  POKE AA + 4,XB: POKE AA + 12,XB: POKE AA + 5,XH: POKE AA +
     13,XH: POKE AA + 6,Y%: POKE AA + 7,Y%: POKE AA + 9,0
1230 PT = 1: XDRAW CU AT X%,Y%: CALL AS: RETURN
1240  REM  ** T **
1250 PT = 1:CR = CT:S = CT: XDRAW CU AT X%,Y%: XDRAW CT AT X%,Y%
1260 C =  PEEK (49152) - 128: IF C < 0 GOTO 1260
1270  POKE 49168,0:S = C - 29: IF C = 27 THEN TT = 1 - TT: XDRAW
     CT AT X%,Y%:CT = 34 * (TT = 1) + CR * (TT = 0): XDRAW CT AT
    X%,Y%: GOTO 1260
1280  IF C = 17 THEN  TEXT : HOME : FOR I = 41 TO 51: PRINT IN$(
    I): NEXT : GET CAR$: POKE 49232,0: GOTO 1260
1284  IF ME AND C = 9 THEN MN = 1 - MN
1290  IF C <  > 32 OR TT = 0 GOTO 1310
1292  IF MN = 0 GOTO 1300
1293  DRAW CT AT X%,Y%: GOSUB 100: XDRAW CT AT X%,Y%
1294  IF  PEEK (49152) > 127 GOTO 1260
1296 XX% = X%:YY% = Y%: GOSUB 100: IF X% = XX% AND Y% = YY% GOTO
     1294
1297  XDRAW CT AT XX%,YY%: XDRAW CT AT X%,Y%: GOTO 1294
1300 KK = 1: GOTO 1260
1310  IF C < 32 OR C = 127 GOTO 1430
1320  IF  PEEK (49249) > 127 AND C > 47 AND C < 91 THEN S = S +
     80
1330  XDRAW CT AT X%,Y%: DRAW S AT X%,Y%
1340  IF TT = 0 THEN CT = CR:KK = 0: GOTO 1370
1350 CT = S: IF MN GOTO 1294
```




```
1360  GOTO 1260
1370 X% = X% + 7
1380  IF X% > 277 THEN Y% = Y% + 8:X% = X% - 273: IF X% > 8 THEN
     X% = X% - 7
1390  IF X% < 2 THEN Y% = Y% - 8:X% = X% + 273: IF X% < 271 THEN
     X% = X% + 7
1400  IF Y% < 4 THEN Y% = 186
1410  IF Y% > 186 THEN Y% = 4
1420  XDRAW CT AT X%,Y%: GOTO 1260
1430  IF KK = 1 THEN KK = 0: DRAW CT AT X%,Y%: GOTO 1450
1440  XDRAW CT AT X%,Y%
1450  IF C = 13 THEN RO = 0: ROT= RO:CT = CR:TT = 0:SC = 1: SCAL
    E= SC:KK = 0: RETURN
1460  IF (C = 1 AND SC < 51) OR (C = 26 AND SC > 1) THEN SC = SC
     + (C = 1) - (C = 26): SCALE= SC
1470  IF C = 23 THEN RO = (RO = 56) * 0 + (RO < 56) * (RO + 8):
    ROT= RO
1480  IF C = 127 THEN  DRAW CT AT X%,Y%: XDRAW CT AT X%,Y%:C = 8
1485  IF TT = 1 AND MN = 1 THEN  XDRAW CT AT X%,Y%: GOTO 1294
1490 VX = 7:VY = 8
1500  IF  PEEK (49250) > 127 THEN VX = 1:VY = 1
1510  IF  PEEK (49249) > 127 THEN VX = 70:VY = 48
1520  IF C = 21 THEN X% = X% + VX
1530  IF C = 8 THEN X% = X% - VX
1540  IF C = 10 THEN Y% = Y% + VY
1550  IF C = 11 THEN Y% = Y% - VY
1560  GOTO 1380
1570  REM  ** C,D,A,E **
1580 PT = 1:RX = X0% - X1%:RY = Y1% - Y0%:RA =  INT ( SQR (RX ^
     2 + RY ^ 2) + .5)
1590  IF RA = 0 THEN  RETURN
1600  XDRAW CU AT X0%,Y0%
1610  IF R = 65 OR R = 69 GOTO 1750
1620 XH =  INT (X1% / 256):XB = X1% - XH * 256: POKE AA + 4,XB:
    POKE AA + 5,XH: POKE AA + 6,Y1%
1630  IF RA > 255 THEN  RETURN
1640  IF R = 68 GOTO 1660
1650  POKE AA + 9,RA: CALL AC: RETURN
1660 RM = RA: FOR RA = 1 TO RM: GOSUB 1650: IF  PEEK (49152) > 1
    27 THEN  POKE 49168,0: RETURN
1662  NEXT
1670  IF RA < 64 GOTO 1730
1680  FOR I =  - RA TO  - 64: GOSUB 1710: NEXT
1690  FOR I = 64 TO RA: GOSUB 1710: NEXT
1700 RA = 64: GOTO 1730
1710  IF Y1% + I >  - 1 AND Y1% + I < 192 AND X1% > 1 AND X1% <
    192 THEN  HPLOT X1% - 1,Y1% + I TO X1% + 1,Y1% + I
1720  RETURN
1730  FOR I =  - RA TO RA: IF Y1% + I >  - 1 AND Y1% + I < 192 T
    HEN  HPLOT X1%,Y1% + I
1740  NEXT : RETURN
1750 D = .02
1760  IF R = 65 THEN X = X0%:Y = Y0%:EX = 1:EY = 1: GOTO 1810
1770  IF RX = 0 OR RY = 0 THEN  RETURN
1780  IF  ABS (RX) >  =  ABS (RY) THEN X = X1% + RX:Y = Y1%: GOT
    O 1800
1790 X = X1%:Y = Y1% - RY
1800 EX =  ABS (RX / RY):EY =  ABS (RY / RX)
1810  HPLOT X,Y
1820 Y = Y + EY * D * (X - X1%):X = X - EX * D * (Y - Y1%)
1830  IF X < 0 OR X > 279 OR Y < 0 OR Y > 191 GOTO 1820
1840  HPLOT  TO X,Y
1850  IF  PEEK (49152) > 127 THEN TH =  PEEK (49152) - 128: POKE
     49168,0: GOTO 1870
1860  GOTO 1820
1870  IF TH = 13 GOTO 1900
1880  GET CAR$: IF CAR$ =  CHR$ (13) GOTO 1900
1890  GOTO 1820
```

```
1900 X% = X:Y% = Y: RETURN
1910  REM  ** G,Y **
1920 XO% = X%:YO% = Y%: DRAW CG AT XO%,Y
   O%
1930  IF MN THEN MR = 1:MN = 0
1940  GOTO 1970
1950  IF  PEEK (49152) > 127 THEN R =  P
   EEK (49152) - 128: POKE 49168,0
1960  GOSUB 80: XDRAW CG AT XO%,YO%:XO%
   = X%:YO% = Y%: DRAW CG AT XO%,YO%
1970  IF R = 89 THEN  XDRAW CG AT XO%,YO
   %: ROT= 16: DRAW CG AT XO%,YO%
1980  IF R = 71 THEN  XDRAW CG AT XO%,YO
   %: ROT= 0: DRAW CG AT XO%,YO%
1990  IF R = 13 THEN  XDRAW CG AT XO%,YO
   %: DRAW CU AT XO%,YO%: ROT= 0:MN = MR
   :MR = 0: RETURN
2000 R = 0: GOTO 1950
2010  REM  ** CHARGER,ENREGISTRER,CATALO
   G **
2020  IF  NOT PL THEN  RETURN
2030  HTAB 1: VTAB 14: PRINT "LECTEUR N°
   :";: INVERSE : PRINT NL;: NORMAL : CA
   LL  - 1008: GET CAR$: IF CAR$ =  CHR$
    (13) THEN  RETURN
2040  IF CAR$ <  > "1" AND CAR$ <  > "2"
    GOTO 2020
2050 NL =  VAL (CAR$): INVERSE : PRINT N
   L: NORMAL : RETURN
2060  GOSUB 2020
2070  PRINT : PRINT D$"CATALOG,D"NL: PRI
   NT : INPUT "NOM DE L'IMAGE:";IM$: IF
   IM$ = "" GOTO 2110
2080 PG =  ASC ( LEFT$ (IM$,1)): IF PG <
    65 OR PG > 90 THEN  VTAB 9: GOTO 207
   0
2090  IF I = 2 THEN  XDRAW CU AT X%,Y%:
   PRINT D$"BSAVE"IM$",A$2000,L$2000,D";
   NL: GOTO 2110
2100  PRINT D$"BRUN CHARGE.OBJ,A$9000"
2105  & IM$,8192
2110  RETURN
2120  HOME : GOSUB 2020: HOME : PRINT
2130  PRINT D$"CATALOG,D"NL: INVERSE : P
   RINT "FIN": NORMAL
2140  GET CAR$: HOME : RETURN
2150  REM  ** INFORMATIONS **
2160  TEXT : HOME : FOR I = 1 TO 22
2170  IF (I = 4 OR I = 5) AND ME = 0 GOT
   O 2200
2180  IF I = 21 AND MA = 0 THEN I = 25::
    PRINT "B -> ECHANGE BIS": GOTO 2200
2190  PRINT IN$(I)
2200  NEXT
2210  PRINT IN$(23);
2220  GET CAR$: POKE 49232,0: POKE 49239
   ,0: POKE 49236,0: RETURN
2230  REM  ** DEBUT **
2240  HOME : VTAB 8: HTAB 12: INVERSE :
   PRINT " SCHEMATEUR ": NORMAL : VTAB 1
   4: PRINT "TOUCHE 'Q'  POUR ACCEDER AU
    MENU": VTAB 19: PRINT "TOUCHE 'Ctrl
   Q'  POUR AIDE-MEMOIRE"
2250 ZC = 29:IN = 51:N = 6
2260 ME = 0: IF  PEEK (207) THEN ME = 1
2270 PL = 0: IF  PEEK (214) THEN PL = 1
2280 MA = 0: IF  PEEK (215) THEN MA = 1
2290 AA = 2817:AL = 6000:AE = AA + 17
2300  DIM R(ZC),IN$(IN)
2310 CU = 1:CG = 2:CT = 98:KK = 0:SC = 1
   :RO = 0:CR = 1:TT = 0
2320 DE = 5:CL = 3:MN = 0:MR = 0:X% = 13
   9:Y% = 95:XO% = X%:YO% = Y%:PT = 0:DP
    = 0:ST = 1:NL = 1:B = 0:DH = 0:NI =
   0:VX = 12:VY = 12:MD = 0:PA = 3:EP =
   1
2330 C = 0:S = 0:PG = 0:RX = 0:RY = 0:RA
    = 0:D = 0:EX = 1:EY = 1:TH = 0:R = 0
   :X1% = 0:Y1% = 0:XH = 0:XB = 0:CH = 0
   :X = 0:Y = 0:NU = 0:CE = 0
2340 IM$ = "":CAR$ = "":CL$ = "":D$ =  C
   HR$ (4)
2350  FOR I = 1 TO IN: READ IN$(I): NEXT

2360  FOR I = 1 TO ZC: READ R(I): NEXT
2370  IF  PEEK (235) GOTO 2420
2380  POKE 233, INT (AL / 256): POKE 232
   ,AL - 256 * ( INT (AL / 256))
2390  PRINT D$"BRUN CHARGE.OBJ,A$9000"
2400  & "CARAC.SH",AL: & "SCHEM.C",AA
2410  POKE 235,1
2420 AF = AE + 3:AB = AE + 6:AO = AE + 9
   :AI = AE + 12:AU = AE + 15:AC = AE +
   18:AG = AE + 21:AV = AE + 24:AP = AE
   + 27:AW = AE + 30:AS = AE + 33
2430  POKE AA + 16,1: POKE AA + 3,NI: PO
   KE AA + 8,B
2440  POKE AA + 1,MD: POKE AA + 2,PA: PO
   KE AA,EP - 1
2450  ONERR  GOTO 2710
2460  POKE 49232,2: POKE 49239,0: POKE 4
   9236,0: POKE 49234,0
2470  HCOLOR= CL: ROT= RO: SCALE= 1
2480  DRAW CU AT XO%,YO%
2490  GOSUB 200: TEXT : HOME
2500  REM  ** MENU **
2510  FOR I = 1 TO N: GOSUB 2670: NEXT
2520 I = 1: INVERSE : GOSUB 2670: NORMAL

2530  VTAB 22: PRINT "FLECHES -> CHOISIR
      RETURN -> VALIDER"
2540  VTAB 23: GET CAR$:CH =  ASC (CAR$)
2550  IF CH = 10 OR CH = 8 THEN  GOSUB 2
   670:I = I + 1: INVERSE : GOSUB 2670:
   NORMAL
2560  IF CH = 11 OR CH = 21 THEN  GOSUB
   2670:I = I - 1: INVERSE : GOSUB 2670:
    NORMAL
2570  IF CH = 13 THEN  HOME : GOTO 2590
2580  GOTO 2540
2590  ON I GOTO 2600,2610,2620,2660,2630
   ,2640
2600  POKE 49232,0: POKE 49239,0: POKE 4
   9236,0: GOTO 2490
2610  GOSUB 2060: GOTO 2600
2620  GOSUB 2060:PT = 1: GOTO 2600
2630  GOSUB 2120: GOTO 2510
2640  XDRAW 1 AT X%,Y%: TEXT : HOME : PR
   INT
2650  PRINT D$"RUN MENU,D1": REM  ** FI
   N **
2660  PRINT : PRINT D$"RUN GRAVURE,D1"
2670  IF I = N + 1 THEN I = 1
2680  IF I = 0 THEN I = N
2690  VTAB I + DE: HTAB 4: PRINT IN$(34
   + I): RETURN
2700  REM  ** ERREUR **
2710 ER =  PEEK (222): IF ER = 6 THEN  T
   EXT : HOME : VTAB 10: PRINT "FICHIER
   ABSENT": GET CAR$
2720  HOME : GOTO 2510
2730  DATA  LES FLECHES DEPLACENT LE POI
   NT,POMME BLANCHE -> RAPIDE,POMME NOIR
   E ->LENT,Tab -> MANETTES, Tab -> FLEC
   HES,Del -> EFFACE
2740  DATA  K -> COULEUR  I -> INVE
   RSION,X -> POINT DE DEPART,Z -> DEPAR
   T PERMANENT,BARRE ESPACE -> RELIE DEU
   X POINTS,H -> RELIE HORIZONTALEMENT,V
    -> RELIE VERTICALEMENT
2750  DATA  R -> RECTANGLE,C -> CERC
   LE  D -> DISQUE,F -> FENETRES
      M -> MOUVEMENT,A -> ARC DE CERCLE
      E -> ELLIPSE,G -> GOMME HORIZONTA
   LE  Y ->VERTICALE
2760  DATA  S -> REMPLIT UNE SURFACE,
   P -> MODE REMPLISSAGE  L -> LIMITES,
   W -> ECHANGE DESSINS  U -> UNION,B
   -> METTRE EN BANQUE  O -> OBSERVER,N
    -> REMPLACER EN BANQUE  J -> NOMBRES
   ,T -> TEXTE  Ctrl Q -> INFORMATIO
   NS
2770  DATA  MODE DE REMPLISSAGE DES SUR
   FACES,C -> REMPLISSAGE COMPLET,V -> L
   IGNES VERTICALES,H -> LIGNES HORIZONT
   ALES,P -> POINTILLES,Q -> QUADRILLAGE
   ,D -> HACHURES PENCHEES A DROTE,G ->
   HACHURES PENCHEES A GAUCHE,L -> LOSAN
   GES
2780  DATA  L'OPTION 'PAS' DONNE L'ECART
    ENTRE LES  MOTIFS,L'OPTION 'EPAISSEU
   R' DONNE L'EPAISSEUR  DES TRAITS
2790  DATA  RETOUR A L'IMAGE,ENREGISTRER
    L'IMAGE,CHARGER UNE IMAGE,IMPRIMER,C
   ATALOGUE,QUITTER LE PROGRAMME
2800  DATA  VOUS ETES EN MODE TEXTE,T
   OUT LE CLAVIER AZERTY EST OPERATIONNE
   L,LES FLECHES DEPLACENT LE CURSEUR,De
   l -> EFFACE,Ctrl A -> AGRANDIT,Ctrl Z
    -> RAPETISSE,Ctrl W -> FAIT TOURNER
2810  DATA  -,Esc -> AFFICHE CARACTERE
   OU SHAPE,DEPLACEMENT COMME EN MODE GR
   APHIQUE,BARRE ESPACE -> MARQUE
2820  DATA  81,127,78,76,90,88,86,32,72
   ,82,70,75,79,17,87,83,84,67,68,65,69,
   71,89,85,73,66,80,74,77
```

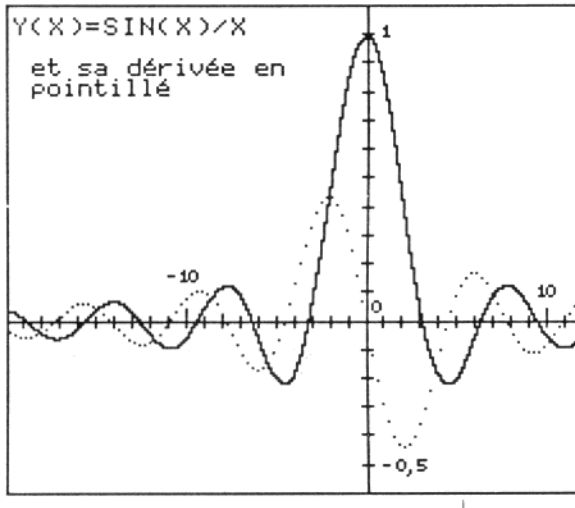
Programme GRAVURE

```
5  HIMEM: 36863
10  REM  *** GRAVURE ***
20 D$ =  CHR$ (4):DM = 1
30  ONERR  GOTO 890
40  IF  PEEK (103) + 256 *  PEEK (104) =
    24577 GOTO 70
50  POKE 103,1: POKE 104,96: POKE 24576,
   0
60  PRINT D$"RUN GRAVURE"
70  HOME : VTAB 10: HTAB 6: INVERSE : PR
   INT " PROGRAMME  GRAVURE ": NORMAL :
   VTAB 15: HTAB 6: PRINT "SORTIE SUR IM
   PRIMANTE": HTAB 11: PRINT "IMAGEWRITE
   R"
80  DIM C(8),D(14)
90  FOR I = 1 TO 8: READ C(I): NEXT : FO
   R I = 1 TO 14: READ D(I): NEXT
100  FOR I = 0 TO 3: READ FR$(I): NEXT :
    FOR I = 1 TO 3: READ IN$(I): NEXT
110 IH = 48:IB = 48:MB = 48:MH = 48
120 NL = 1:PL = 0: IF  PEEK (214) THEN P
   L = 1
130 AG = 2049:AC = 206: REM  **ADRESSES
    DU PROGRAMME COUT ET DU CARACTERE EN
   VOYE**
140 AP = AG + 6: REM  *ADRESSE DU PROGR
   AMME GRAV.C**
150  IF  PEEK (237) GOTO 180
160  PRINT : PRINT D$"BLOAD GRAV.C,A"AG
170  POKE 237,1
180  HOME : VTAB 9: PRINT "RETURN POUR I
   MAGE EN COURS": PRINT "AUTRE TOUCHE P
   OUR NOUVELLE IMAGE";: GET CAR$: IF CA
   R$ <  >  CHR$ (13) GOTO 210
200  POKE 49232,0: POKE 49234,0: POKE 49
   236,0: POKE 49239,0: GOTO 270
210  IF  NOT PL GOTO 250
220  HTAB 1: VTAB 13: PRINT "LECTEUR N°:
   ";: INVERSE : PRINT NL;: NORMAL : CAL
   L  - 1008: GET CAR$: PRINT CAR$: IF C
   AR$ =  CHR$ (13) GOTO 250
230  IF CAR$ < "1" OR CAR$ > "2" GOTO 22
   0
240 NL =  VAL (CAR$)
250  PRINT : PRINT D$"CATALOG,D"NL: PRIN
   T : INPUT "NOM DE L'IMAGE:";IM$: IF I
   M$ = "" GOTO 180
260  HGR : POKE 49234,0: PRINT : PRINT D
   $"BRUN CHARGE.OBJ,A$9000": & IM$,8192
270  GET CAR$: TEXT
280  HOME : PRINT "L'IMPRIMANTE EST-ELLE
    PRETE(O/N)?:";: INVERSE : PRINT "O";
   : NORMAL : CALL  - 1008: GET CAR$: PR
   INT CAR$: IF CAR$ = "N" OR CAR$ = "n"
    GOTO 760
290  VTAB 3: PRINT "NOUVELLES DISPOSITIO
   NS(O/N)?:";: INVERSE : PRINT "O";: NO
   RMAL : CALL  - 1008: GET CAR$: PRINT
   CAR$: IF CAR$ <  > "N" AND CAR$ <  >
    "n" GOTO 310
300  GOTO 580
310 IC = 0: HOME : VTAB 3: PRINT "INVERS
   ER NORMALEMENT LA COULEUR(O/N)?:";: I
   NVERSE : PRINT "O";: NORMAL : CALL  -
    1008: GET CAR$: PRINT CAR$: IF CAR$
   = "O" OR CAR$ = "o" OR CAR$ =  CHR$ (
   13) THEN IC = 1
320  HOME : VTAB 2: PRINT "L'image peut
   s'imprimer en deux hauteursHauteur si
   mple(6.7cm) ou double(13.4cm)"
330 HI = 1: VTAB 5: PRINT "IMAGE DE HAUT
   EUR SIMPLE(O/N)?:";: INVERSE : PRINT
   "O";: NORMAL : CALL  - 1008: GET CAR$
   : PRINT CAR$: IF CAR$ = "O" OR CAR$ =
    "o" OR CAR$ =  CHR$ (13) THEN HI = 0
340 DM = IC + 2 * HI
350  IF DM > 1 AND D(1) = 4.4 THEN  FOR
   I = 1 TO 8:D(I) = 2 * D(I): NEXT
360  IF DM < 2 AND D(1) = 8.8 THEN  FOR
   I = 1 TO 8:D(I) = D(I) / 2: NEXT
370  HOME : VTAB 3: PRINT FR$(DM): VTAB
   5: PRINT "DEFORMATION HORIZONTALE(O/N
   )?:";: INVERSE : PRINT "N";: NORMAL :
    CALL  - 1008: GET CAR$: PRINT CAR$
380  IF CAR$ = "N" OR CAR$ = "n" OR CAR$
    =  CHR$ (13) THEN DO = 14:D = D(8):D
   I = 2: GOTO 480
390 II = 14: IF DM > 1 THEN II = 8
400  VTAB 7: PRINT "LARGEUR DE L'IMAGE:"
   : FOR I = 1 TO II: PRINT  CHR$ (64 +
   I)"- ("D(I)" cm)": NEXT : VTAB 15: HT
   AB 12: INVERSE : PRINT "NORMAL": NORM
```

```
AL
410  IF DM > 1 GOTO 460
420  VTAB 23: PRINT "CHOIX(A à N):";: GE
   T CAR$:DI =  ASC (CAR$) - 64: IF DI <
    1 OR DI > 14 GOTO 420
430 DO = 15: IF DI > 6 THEN DO = 14
440 D = D(DI): IF DI > 6 THEN DI = DI -
   6
450  GOTO 480
460  VTAB 17: PRINT "CHOIX(A à H):";: GE
   T CAR$:DI =  ASC (CAR$) - 64: IF DI <
    1 OR DI > 8 GOTO 460
470 DO = 15:D = D(DI)
480  HTAB 1: VTAB 5: CALL  - 958:CI = C(
   DI): PRINT : PRINT "LARGEUR DE L'IMAG
   E: "D" cm":H = 6.7 * (DM < 2) + 13.4
   * (DM > 1): PRINT "HAUTEUR DE L'IMAGE
   : "H" cm"
490  VTAB 10: HTAB 1: PRINT "POSITIONNEM
   ENT DU COIN SUPERIEUR GAUCHE DE L'ECR
   AN CONTENANT L'IMAGE:"
500  VTAB 13: HTAB 1: PRINT "MARGE HAUTE
    (DE 0 A 24 cm):";: INVERSE : PRINT "
   0";: NORMAL : CALL  - 1008: INPUT "";
   Y$: IF Y$ = "" THEN  VTAB 13: HTAB 29
   : PRINT "0":Y = 0: GOTO 520
510 Y =  VAL (Y$): IF Y < 0 OR Y > 24 GO
   TO 500
520 IL = 3.85 * Y:IH =  INT (IL / 10):IB
    =  INT (IL - 10 * IH) + 48:IH = IH +
   48
530  VTAB 15: HTAB 1: PRINT "(10.1 pour
   raccorder une image simple)"
540  VTAB 14: HTAB 1: PRINT "MARGE GAUCH
   E (DE 0.3 A 16.5cm):";: INVERSE : PRI
   NT "0.3";: NORMAL : CALL  - 1008: CAL
   L  - 1008: CALL  - 1008: INPUT "";X$:
    IF X$ = "" THEN  VTAB 14: HTAB 32: P
   RINT "0.3":X = 0.3: GOTO 560
550 X =  VAL (X$): IF X < 0.3 OR X > 16.
   5 GOTO 540
560 MG = 3.6 * (X - 0.3):MH =  INT (MG /
    10):MB =  INT (MG - 10 * MH) + 48:MH
   = MH + 48
570  VTAB 18: PRINT "EST-CE EXACT(O/N)?:
   ";: INVERSE : PRINT "O";: NORMAL : CA
   LL  - 1008: GET CAR$: PRINT CAR$: IF
   CAR$ = "N" OR CAR$ = "n" GOTO 760
580  PRINT : PRINT D$"PREI"
590  PRINT  CHR$ (27);"n"
600  POKE 6,1: POKE 7,DM
610  POKE AG + 15,CI
620  POKE AC,27: CALL AG: POKE AC,114: C
   ALL AG
630  POKE AC,27: CALL AG: POKE AC,84: CA
   LL AG: POKE AC,50: CALL AG: POKE AC,5
   5: CALL AG: POKE AC,31: CALL AG: POKE
    AC,49: CALL AG
640  POKE AC,27: CALL AG: POKE AC,102: C
   ALL AG
650  IF IH = 48 AND IB = 48 GOTO 680
660  POKE AC,27: CALL AG: POKE AC,84: CA
   LL AG: POKE AC,IH: CALL AG: POKE AC,I
   B: CALL AG
670  POKE AC,31: CALL AG: POKE AC,63: CA
   LL AG
680  POKE AC,27: CALL AG: POKE AC,110: C
   ALL AG
690  POKE AC,27: CALL AG: POKE AC,76: CA
   LL AG: POKE AC,48: CALL AG: POKE AC,M
   H: CALL AG: POKE AC,MB: CALL AG
700  POKE AC,DO: CALL AG
710  POKE 49239,0: POKE 49234,0: POKE 49
   232,0
720  POKE 6,1: POKE 7,DM
730  CALL AP
740  PRINT : PRINT D$"PREO"
750  POKE 49233,0
760  HOME : FOR I = 1 TO 3: GOSUB 770: N
   EXT
761 I = 1: INVERSE : GOSUB 770: NORMAL
762  VTAB 22: PRINT "FLECHES -> CHOISIR
    RETURN -> VALIDER"
763  VTAB 23: GET CAR$:CH =  ASC (CAR$)
764  IF CH = 10 OR CH = 8 THEN  GOSUB 77
   0:I = I + 1: INVERSE : GOSUB 770: NOR
   MAL
765  IF CH = 11 OR CH = 21 THEN  GOSUB 7
   70:I = I - 1: INVERSE : GOSUB 770: NO
   RMAL
766  IF CH = 13 THEN  HOME : GOTO 775
767  GOTO 763
770  IF I = 4 OR I = 0 THEN I =  ABS (I
   - 3)
772  VTAB I + 8: HTAB 4: PRINT IN$(I): R
   ETURN
775  ON I GOTO 790,790,780
780  PRINT : PRINT D$"RUN MENU,D1"
790  HOME : VTAB 10: PRINT "REVENIR AU N
   IVEAU PRECEDENT (O/N):";: INVERSE : P
   RINT "N";: NORMAL : CALL  - 1008: GET
    RM$: PRINT RM$: IF RM$ = "O" OR RM$
   = "o" THEN  GOSUB 820
800  IF I = 1 THEN  HOME : GOTO 290
810  GOTO 210
820  PRINT : PRINT D$"PREI"
830  POKE AC,27: CALL AG: POKE AC,114: C
   ALL AG
840  POKE AC,27: CALL AG: POKE AC,84: CA
   LL AG: POKE AC,55: CALL AG: POKE AC,5
   4: CALL AG
850  POKE AC,31: CALL AG: POKE AC,53: CA
   LL AG
855  POKE AC,27: CALL AG: POKE AC,84: CA
   LL AG: POKE AC,48: CALL AG: POKE AC,4
   9: CALL AG: POKE AC,31: CALL AG: POKE
    AC,49: CALL AG
860  POKE AC,27: CALL AG: POKE AC,102: C
   ALL AG
870  PRINT : PRINT D$"PREO"
880  RETURN
890 ER =  PEEK (222)
900  IF ER = 6 THEN  TEXT : HOME : VTAB
   10: PRINT "PAS D'IMAGE DE CE NOM.": P
   RINT "TAPEZ UNE TOUCHE.": GET CAR$: H
   OME : GOTO 760
910  GOTO 760
920  DATA  80,112,81,113,101,69,78,110
930  DATA  4.4,5,5.3,5.8,6.6,7.5,8.8,10,
   10.6,11.6,13.2,15,17.6,20
940  DATA  HAUTEUR SIMPLE - PAS D'INVERS
   ION,HAUTEUR SIMPLE - INVERSION NORMAL
   E,HAUTEUR DOUBLE - PAS D'INVERSION,HA
   UTEUR DOUBLE - INVERSION NORMALE
950  DATA  IMPRESSION DE LA MEME IMAGE,A
   UTRE IMAGE,QUITTER LE PROGRAMME
300  IF I = N + 1 THEN I = ND
310  IF I = ND - 1 THEN I = N
320  VTAB I + DE - 1: PRINT I;"-";R$(I):
    RETURN
330  REM  ** CONFIGURATION **
340  HOME : VTAB 10: PRINT "CONFIGURATIO
   N ACTUELLE:"
350  VTAB 12: HTAB 1: PRINT "PAS DE MANE
   TTES": IF  PEEK (207) THEN  VTAB 12:
   HTAB 1: CALL  - 868: PRINT "MANETTES"
360  VTAB 13: HTAB 1: PRINT "1 LECTEUR":
    IF  PEEK (214) THEN  VTAB 13: HTAB 1
   : PRINT "2 LECTEURS"
370  VTAB 14: HTAB 1: PRINT "PAS DE MEMO
   IRE AUXILIAIRE": IF  PEEK (215) THEN
    VTAB 14: HTAB 1: CALL  - 868: PRINT
   "MEMOIRE AUXILIAIRE"
380  VTAB 18: HTAB 1: PRINT "RETURN POUR
    CONFIRMER.": PRINT "'C' POUR MODIFIE
   R": GET CAR$: IF CAR$ <  > "C" AND CA
   R$ <  > "c" GOTO 150
390  HOME : VTAB 10: PRINT "AVEZ-VOUS DE
   S MANETTES?(O/N):";: GET ME$: PRINT M
   E$:ME = ((ME$ = "N") + (ME$ = "O") *
   2) - 1: IF ME =  - 1 GOTO 390
400  VTAB 12: PRINT "AVEZ-VOUS DEUX LECT
   EURS?(O/N):";: GET PL$: PRINT PL$:PL
   = ((PL$ = "N") + (PL$ = "O") * 2) - 1
   : IF PL =  - 1 GOTO 400
410  VTAB 14: PRINT "MEMOIRE AUXILIAIRE?
   (O/N):";: GET MA$: PRINT MA$:MA = ((M
   A$ = "N") + (MA$ = "O") * 2) - 1: IF
   MA =  - 1 GOTO 410
420  POKE 24613,ME + 48: POKE 24620,PL +
    48: POKE 24627,MA + 48
430  PRINT  CHR$ (13); CHR$ (4)"SAVE MEN
   U"
440  PRINT  CHR$ (13); CHR$ (4)"RUN MENU
   "
```

Programme MENU

```
10  REM  ** PROGRAMME MENU **
20  POKE 207,1: POKE 214,1: POKE 215,1:
   REM  *MANETTES,LECTEURS,MEMOIRE AUXIL
   IAIRE*
30 N = 3: REM     ** NOMBRE DE PROGRAMMES
    **
40  REM   ** TITRES DU MENU: **
50  DATA     PROGRAMME SCHEMATEUR,GRAPHES
    DE FONCTIONS,IMPRESSION SUR IMAGEWRI
   TER
60 N = N + 3
70  DATA  CONFIGURATION,CATALOGUE DE LA
   DISQUETTE,QUITTER
80  REM   ** NOMS DES PROGRAMMES: **
90  DATA     SCHEMATEUR,FONCTION,GRAVURE
100  REM   ** DISPOSITION DU MENU: **
110 ND = 1:DE = 12:D$ =  CHR$ (4)
120  DIM R$(N),F$(N)
130  FOR I = 1 TO N: READ R$(I): NEXT
140  FOR I = 1 TO N - 3: READ F$(I): NEX
   T
150  HOME : VTAB 4: PRINT "M.CAUET": PRI
   NT "OCTOBRE 1986"
160  GOSUB 220
170  IF I = N THEN  HOME : END
180  IF I = N - 1 THEN  PRINT : PRINT D$
   "CATALOG": PRINT : INVERSE : PRINT "R
   ETURN": NORMAL : GET CAR$: GOTO 150
190  IF I = N - 2 THEN  GOTO 340
200  PRINT : PRINT D$"RUN"F$(I)
210  REM  ** AFFICHAGE MENU **
220  FOR I = ND TO N: GOSUB 300: NEXT :I
   = ND: INVERSE : GOSUB 300: NORMAL
230  VTAB 22: PRINT "FLECHES POUR CHOISI
   R-RETURN POUR VALIDER"
240  VTAB 23: GET ME$:ME =  ASC (ME$)
250  IF ME = 10 OR ME = 21 THEN  GOSUB 3
   00:I = I + 1: INVERSE : GOSUB 300: NO
   RMAL
260  IF ME = 11 OR ME = 8 THEN  GOSUB 30
   0:I = I - 1: INVERSE : GOSUB 300: NOR
   MAL
270  IF ME = 13 THEN  HOME : RETURN
280  IF ME > 48 AND ME < 49 + N THEN I =
    ME - 48: HOME : RETURN
290  GOTO 240
```



Programme GRAPHISME

```
10  REM  ** HELLO **
20  HOME
30  HGR2
40  HGR
50  TEXT
60  REM  ** NETTOIE MEV COMMUTEE **
70  PRINT  CHR$ (4)"BLOAD HEL.C"
80  CALL 768
90  REM   ** POINTEURS PROG.MACHINE:SCHE
   M/CARAC - FONC - GRAV **
100  FOR I = 235 TO 237: POKE I,0: NEXT
110  REM  ** PLACE LES PROGRAMMES AU-DES
   SUS DE HGR2 **
120  POKE 103,1: POKE 104,96: POKE 24576
   ,0
130  PRINT  CHR$ (13); CHR$ (4)"RUN MENU
   "
140  END
```

Récapitulation CARAC.SH

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE CARAC.SH, A$4000,L$6C1

```
4000:8D 00 1C 01 1E 01 23 01
4008:25 01 2C 01 30 01 3C 01
4010:49 01 54 01 60 01 64 01
4018:6C 01 74 01 81 01 89 01
4020:8D 01 92 01 95 01 9C 01
4028:AA 01 B2 01 BC 01 C7 01
4030:D1 01 DD 01 E7 01 F0 01
4038:FB 01 06 02 0A 02 0F 02
4040:18 02 20 02 29 02 33 02
4048:41 02 4D 02 59 02 64 02
4050:70 02 7D 02 87 02 92 02
4058:9E 02 A7 02 AF 02 BC 02
4060:C5 02 D1 02 DD 02 E9 02
4068:F3 02 FF 02 0C 03 17 03
4070:20 03 2B 03 36 03 42 03
4078:4E 03 57 03 62 03 67 03
4080:72 03 83 03 88 03 8F 03
4088:95 03 A0 03 AB 03 B3 03
4090:BE 03 C9 03 D2 03 DD 03
4098:E7 03 EF 03 F8 03 04 04
40A0:0C 04 15 04 1E 04 28 04
40A8:33 04 3E 04 46 04 50 04
40B0:5A 04 63 04 6C 04 76 04
40B8:82 04 8D 04 98 04 A5 04
40C0:B0 04 BD 04 C2 04 DA 04
40C8:E2 04 E8 04 F1 04 F9 04
40D0:00 05 0A 05 12 05 18 05
40D8:20 05 29 05 35 05 44 05
40E0:4D 05 61 05 6C 05 7A 05
40E8:86 05 93 05 A6 05 B5 05
40F0:C2 05 CB 05 D9 05 E7 05
40F8:F8 05 09 06 18 06 24 06
4100:2D 06 39 06 44 06 4E 06
4108:56 06 62 06 6D 06 76 06
4110:80 06 88 06 92 06 9D 06
4118:A8 06 B5 06 04 00 1B 2D
4120:2D 04 00 03 00 40 43 33
4128:36 B6 04 00 C0 6C 26 00
4130:40 43 3F 1E 36 6F 1E F6
4138:0D 2D 25 00 E7 0C 25 15
4140:F5 AB 15 1F 15 3F 77 29
4148:00 0C 0C DC 3B 2E 96 17
4150:4D 2E 24 00 60 1C BF AE
4158:17 76 65 1C 0D 16 07 00
4160:08 24 05 00 92 1C 1C 24
4168:0C 0C 06 00 92 0C 0C 24
4170:1C 1C 06 00 3C 1C 4C 6E
4178:1E 16 3F 17 0D 0D DE 07
4180:00 20 8D 3A 3F 77 31 05
4188:00 89 F6 04 00 3F 4C 11
4190:35 00 12 05 00 0C 0C D6
4198:DA 1E 06 00 0C 25 1C 3F
41A0:17 36 2E 1E 0E 2D 0C 24
41A8:07 00 24 BC 96 31 17 2D
41B0:04 00 65 E4 3F 17 95 BA
41B8:2E 2D 25 00 25 0C 3C 3F
41C0:B7 92 15 2D 0C 24 00 3A
41C8:27 0C 0C 0C 36 36 F5 3E
41D0:00 38 27 2C 2D F5 AA 36
41D8:1E 3F 1C 04 00 AD F6 3F
41E0:1C 24 25 0C 0C 35 00 0C
41E8:0C 3C 3F 77 92 36 05 00
41F0:E7 64 2D 15 BE 15 F6 3F
41F8:1C 2C 00 E7 64 2D 15 36
4200:77 1E 17 3F 04 00 08 16
4208:06 00 08 16 BE 05 00 91
4210:E2 1C 1C 0C 0C 0C 06 00
4218:38 67 89 B5 3F 3F 04 00
4220:93 62 0C 0C 1C 1C 1C 06
4228:00 0C 0C 1C 3F 17 95 0A
4230:16 05 00 43 2B 2D 38 70
4238:12 36 3E 3F 1C 45 2B 25
4240:00 3A 37 6E 09 24 67 E4
4248:1C 1E 1E 2E 00 3F 24 2C
4250:2D 15 BE 0E BE 3F 27 2C
4258:00 89 F2 3F 1C 24 24 0C
4260:2D 15 06 00 09 36 1E 3F
4268:27 24 24 2C 2D 15 3E 00
4270:39 B7 3A 24 24 24 2D 2D
4278:96 92 3F 04 00 39 B7 1A
4280:24 24 24 2D 2D 06 00 11
4288:35 3E 3F 1C 24 24 0C 2D
4290:35 00 2B 2D 24 FC 1B 36
4298:36 36 4D 21 24 00 52 3A
42A0:67 24 24 3C 0D 06 00 9B
42A8:72 2D 0C 24 24 3C 00 73
42B0:0E 15 DF 23 24 24 6C 09
42B8:1E 1E 06 00 89 12 3F 3F
42C0:24 24 24 05 00 E0 1C 36
42C8:36 36 4D 21 24 24 BC 06
42D0:00 0E 56 24 24 24 DF 33
42D8:2E 1E 36 2E 00 92 E7 24
42E0:24 0C 2D 15 36 36 17 05
42E8:00 65 3C 38 3F 36 2E 1E
42F0:36 05 00 AA 15 1F 3F 20
42F8:24 64 2D 15 36 36 00 77
4300:15 15 DF 23 24 24 2C 2D
4308:15 BE 06 00 E7 64 2D 15
4310:97 15 F6 3F 1C 04 00 24
4318:1F 28 2D F5 92 33 2E 00
4320:92 E7 24 24 6C 09 36 36
4328:BE 05 00 92 1C 1C 24 24
4330:4D 31 36 BE 06 00 F6 1E
4338:24 24 24 4D 31 36 BE 35
4340:07 00 0C 0C FC 1B 76 16
4348:17 6E 09 E4 04 00 1C 1C
4350:6C 09 F6 D6 36 05 00 0C
4358:0C 3C 3F 77 92 17 2E 2D
4360:25 00 38 60 15 04 00 4B
4368:41 3B 3F 1E AE 2D F5 1E
4370:04 00 40 43 39 3F 1E 0E
4378:2D 15 1E 3F 47 B3 2A 2D
4380:0C 04 00 40 1C 1E 04 00
4388:92 DA 2B 2D 2D 25 00 40
4390:E3 4F 91 04 00 43 2B AD
4398:36 3E 3F 1C 0C 2D 04 00
43A0:43 2B AD 36 1E 3F 27 24
43A8:24 24 00 41 3F BF 36 15
43B0:2D 25 00 41 3B BF 36 15
43B8:2D 25 24 24 24 00 1A 2D
43C0:25 1C 3F 1E 36 15 2D 25
43C8:00 65 E0 3F 1E 36 F5 36
43D0:04 00 8A 3F 1C 2C 28 AD
43D8:36 F6 3F 04 00 28 15 36
43E0:FE 1B 24 24 E5 24 00 18
43E8:0C B0 36 36 2F 21 00 58
43F0:0C B0 36 36 1E E7 04 00
43F8:41 1E 1E 15 0E DF 23 2C
4400:1C 24 24 00 24 3C 96 0A
4408:36 2F 21 00 36 55 24 24
4410:FF 37 36 26 00 38 37 36
4418:6E 09 24 E4 04 00 38 1E
4420:36 15 2D 0C 24 1C 04 00
4428:9A 2D 45 23 1C 3F 1E 36
4430:36 04 00 8A 3F 47 23 0C
4438:2D 15 36 36 04 00 41 3F
4440:1E 1C 36 36 04 00 41 3F
4448:F7 15 2D 15 1E 3F 27 00
4450:08 3F 67 34 12 36 15 65
4458:04 00 41 36 36 47 F3 E7
4460:24 24 00 41 36 1E 1E 1C
4468:1C 24 04 00 41 36 36 E7
4470:34 1A 27 24 24 00 41 1E
4478:1E 1E 1E 4D E1 1C 18 1C
4480:04 00 41 36 36 1E 3F 4C
4488:38 E7 24 04 00 1B 28 2D
4490:F5 1E 1E 1E 2D 2D 04 00
4498:1A 2D 25 1C 44 F3 3E 1E
44A0:36 15 2D 25 00 40 E3 9E
44A8:36 AE 65 35 44 23 24 00
44B0:1A 2D 25 1C 27 1C 16 1E
44B8:36 15 2D 25 00 40 C3 0D
44C0:04 00 92 4A 24 24 24 3C
44C8:36 36 36 3E 24 24 24 3C
44D0:36 36 36 3E 24 24 24 24
44D8:00 1E 45 23 E4 F7 36 0E
44E0:04 00 2B E5 24 BC 04 00
44E8:39 3F 0C 45 2B E0 BF 04
44F0:00 E7 4D 1C 45 23 3F 27
44F8:00 24 24 1F 36 6D 04 00
4500:1B 2D 45 E3 5F 27 2C 2D
4508:04 00 3F 68 E1 3F 64 2D
4510:04 00 24 45 23 3F 27 00
4518:E7 4D 1C E7 0C AD 04 00
4520:1B 2D 45 23 E4 F7 0E 25
4528:00 9B 2D 0E 15 40 2B E0
4530:47 BB 3F 04 00 1B 15 15
4538:15 15 24 24 24 24 2D 2D
4540:2D 2D 04 00 92 2A E5 23
4548:24 24 2C 25 00 1B 45 63
4550:96 AB 0C 2D 2D 75 0C 0C
4558:1C 47 73 1B 5F 3F 3F 04
4560:00 89 92 3F 4D 43 23 24
4568:24 3C 27 00 36 F6 1E 27
4570:40 43 43 43 09 24 0C 0C
4578:25 00 40 43 43 43 FB 2B
4580:6D 6D 1C 16 04 00 2A 28
4588:B0 12 1C 1F 1E 24 24 10
4590:15 04 00 28 0E 1E 15 1E
4598:3F 27 24 0C 96 D2 8F 44
45A0:43 43 43 11 04 00 45 2B
45A8:F8 1B 15 16 1E 15 A5 5A
45B0:41 63 1C 04 00 40 43 2B
45B8:F5 15 36 3F F7 15 2D 0C
45C0:04 00 28 FD F3 15 F5 AB
45C8:2D 25 00 B8 15 65 1C D6
45D0:33 76 1F 40 43 43 23 04
45D8:00 40 AB 15 DF 63 96 D8
45E0:4D 49 1E 3F 3F 04 00 24
45E8:95 A8 F6 3F 1C 64 56 16
45F0:26 40 43 43 43 43 23 00
45F8:1B 2C 28 15 15 1E 1E E7
4600:4D 49 73 65 24 1C F7 04
4608:00 40 43 2B 3D 1B 77 15
```

```
4610:15 1E 1E 1E 2D 2D 04 00
4618:45 23 B5 12 27 1F 5E 67
4620:40 23 27 00 31 76 DF 63
4628:0C 0C 20 24 00 2A 28 30
4630:36 DF 40 E3 36 36 36 04
4638:00 1B A8 36 1E 2D 45 2B
4640:20 1C 04 00 D8 36 AE 45
4648:AB 45 23 24 04 00 28 F5
4650:36 FE 24 24 27 00 27 45
4658:2B 38 3F 96 09 F1 1E 2E
4660:25 00 D2 2D 45 23 1C 3F
4668:1E 36 36 26 00 36 3E 1C
4670:24 45 2B 2D 04 00 3F 36
4678:0E 2D 20 24 E4 F7 26 00
4680:2D F8 13 1E 0D 36 27 00
4688:E3 0D AD 36 36 DF 40 23
4690:04 00 38 1E 2E 36 2F 09
4698:E5 2C E4 04 00 2C 28 F8
46A0:1B 0E 96 1E 4D E1 04 00
46A8:34 32 36 26 08 45 23 FC
46B0:1B 36 15 04 00 60 0C 3F
46B8:16 1A 1E 2D AD 1E 3F 04
46C0:00
```

Récapitulation SCHEM.C

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE SOURIS.C, A$B01,L$9CE

```
0B01:EA EA EA EA EA EA EA
0B08:EA EA EA EA EA EA EA EA
0B10:EA EA 4C 55 0E 4C FE 0D
0B18:4C 3F 0D 4C 54 0D 4C A4
0B20:0D 4C D8 0D 4C 2C 14 4C
0B28:F9 0E 4C A1 0E 4C 80 0E
0B30:4C 42 0E 20 F1 0B 20 FD
0B38:0B 20 12 0C D0 03 4C 33
0B40:0B 20 06 0C 20 F1 0B 20
0B48:06 0C 20 12 0C D0 03 4C
0B50:44 0B 20 FD 0B AD 05 0B
0B58:8D 0B 0B AD 06 0B 8D 0C
0B60:0B 20 5E 0C 20 FD 0B 20
0B68:12 0C D0 27 AD 02 0B D0
0B70:1F CE 07 0B 20 12 0C D0
0B78:14 20 FD 0B 20 12 0C D0
0B80:09 20 06 0C EE 07 0B 4C
0B88:33 0B 20 06 0C EE 07 0B
0B90:4C 61 0B EE 07 0B AD 0B
0B98:0B 8D 05 0B AD 0C 0B 8D
0BA0:06 0B AD 0A 0B D0 23 AD
0BA8:07 0B CD 08 0B 90 24 A9
0BB0:BF 38 ED 07 0B 8D 07 0B
0BB8:8D 0A 0B AD 0D 0B 8D 05
0BC0:0B AD 0E 0B 8D 06 0B 4C
0BC8:33 0B 20 4E 0C CD 08 0B
0BD0:B0 01 60 20 12 0C F0 0B
0BD8:20 FD 0B 20 12 0C D0 F8
0BE0:4C EE 0B 20 06 0C 20 12
0BE8:0C F0 F8 20 FD 0B 4C 55
0BF0:0B CE 07 0B 20 12 0C F0
0BF8:F8 EE 07 0B 60 EE 05 0B
0C00:D0 03 EE 06 0B 60 AD 05
0C08:0B D0 03 CE 06 0B CE 05
0C10:0B 60 AD 00 C0 10 03 68
0C18:68 60 20 4E 0C C9 00 90
0C20:2A C9 C0 B0 26 AE 05 0B
0C28:AC 06 0B F0 07 E0 18 B0
0C30:1A 4C 38 0C E0 00 90 13
0C38:20 13 F4 A4 E5 B1 26 29
0C40:7F 25 30 F0 02 A9 01 4D
0C48:09 0B 60 A9 01 60 AD 0A
0C50:0B F0 07 A9 BF 38 ED 07
0C58:0B 60 AD 07 0B 60 AD 02
0C60:0B D0 03 4C E2 0C C9 01
0C68:F0 1F C9 02 F0 21 C9 03
0C70:F0 23 C9 04 F0 2A C9 05
0C78:F0 31 C9 06 F0 4E C9 07
0C80:D0 06 20 AB 0C 4C CC 0C
0C88:60 20 EF 0C F0 54 60 20
0C90:05 0D F0 4E 60 20 05 0D
0C98:D0 05 20 EF 0C F0 43 60
0CA0:20 05 0D F0 3D 20 EF 0C
0CA8:F0 38 60 20 05 0D 8D 0F
0CB0:0B 20 EF 0C AE 01 0B E8
0CB8:CD 0F 0B F0 25 CE 0F 0B
0CC0:D0 06 AC 03 0B 8C 0F 0B
0CC8:CA D0 ED 60 20 05 0D 8D
0CD0:0F 0B AD 03 0B 18 69 01
0CD8:38 ED 0F 0B 8D 0F 0B 4C
0CE0:B1 0C AE 05 0B AC 06 0B
0CE8:20 4E 0C 20 57 F4 60 AD
0CF0:06 0B D0 06 AD 05 0B 4C
0CF8:08 0D A9 FF 20 08 0D 6D
0D00:05 0B 4C 08 0D 20 4E 0C
0D08:38 4C 0F 0D ED 03 0B CD
0D10:03 0B F0 09 B0 F6 CD 01
0D18:0B B0 02 A2 00 60 A9 00
0D20:85 3C 85 42 A9 F8 85 3E
0D28:60 20 1E 0D A9 40 85 43
0D30:AD 10 0B 85 3D 18 69 1F
0D38:85 3F 18 20 11 C3 60 20
0D40:1E 0D A9 20 85 3D A9 3F
0D48:85 3F AD 10 0B 85 43 38
0D50:20 11 C3 60 A9 20 8D 10
0D58:0B 20 29 0D A9 01 8D 11
0D60:0B 8D 55 C0 20 92 0D AD
0D68:04 0B C9 02 90 F3 EE 11
0D70:0B AD 11 0B 20 8B 0D 8D
0D78:10 0B 20 29 0D 20 92 0D
0D80:AD 11 0B CD 04 0B 90 E6
0D88:4C 54 0D A2 05 0A CA D0
0D90:FC 60 AD 00 C0 10 FB C9
0D98:8D D0 05 68 68 8D 54 C0
0DA0:8D 10 C0 60 A9 00 85 08
0DA8:A9 20 85 09 A2 20 A0 00
0DB0:B1 08 49 FF 91 08 88 D0
0DB8:F7 E6 07 E6 09 CA D0 EE
0DC0:60 AD 83 C0 AD 83 C0 A9
0DC8:00 85 08 85 06 A9 E0 85
0DD0:07 A9 20 85 09 A2 20 60
0DD8:20 C1 0D A0 00 B1 08 48
0DE0:AD 09 0B F0 06 68 31 06
0DE8:4C EE 0D 68 11 06 91 08
0DF0:88 D0 EA E6 07 E6 09 CA
0DF8:D0 E1 AD 81 C0 60 20 C1
0E00:0D 20 65 0E A9 D0 85 07
0E08:A9 00 85 06 A2 10 AD 83
0E10:C0 AD 83 C0 A0 00 B1 06
0E18:8D 0F 0B AD 8B C0 AD 8B
0E20:C0 B1 06 8D 0A 0B AD 0F
0E28:0B 91 06 AD 83 C0 AD 83
0E30:C0 AD 0A 0B 91 06 88 D0
0E38:DD E6 07 CA D0 D6 AD 81
0E40:C0 60 A9 00 85 08 85 06
0E48:A9 20 85 07 A9 40 85 09
0E50:A2 20 4C DB 0D A9 00 85
0E58:08 85 06 A9 40 85 07 A9
0E60:20 85 09 A2 20 A0 00 B1
0E68:08 8D 0F 0B B1 06 91 08
0E70:AD 0F 0B 91 06 88 D0 EF
0E78:E6 07 E6 09 CA D0 E6 60
0E80:A9 00 85 08 85 06 A9 40
0E88:85 07 A9 20 85 09 A2 20
0E90:A0 00 B1 08 91 06 88 D0
0E98:F9 E6 07 E6 09 CA D0 F0
0EA0:60 20 55 0E 8D 55 C0 EA
0EA8:EA 8D 54 C0 EA EA AD 00
0EB0:C0 10 F1 C9 8D D0 07 20
0EB8:55 0E 8D 10 C0 60 C9 A0
0EC0:D0 09 20 42 0E 8D 10 C0
0EC8:4C A4 0E C9 8B D0 09 20
0ED0:CC 11 8D 10 C0 4C A4 0E
0ED8:C9 8A D0 09 20 7A 12 8D
0EE0:10 C0 4C A4 0E C9 95 D0
0EE8:03 20 44 10 C9 88 D0 03
0EF0:20 53 0F 8D 10 C0 4C A4
0EF8:0E 8D 10 C0 AD 00 C0 10
0F00:FB C9 8B D0 06 20 CC 11
0F08:4C F9 0E C9 8A D0 06 20
0F10:7A 12 4C F9 0E C9 95 D0
0F18:06 20 44 10 4C F9 0E C9
0F20:88 D0 06 20 53 0F 4C F9
0F28:0E C9 D7 D0 06 20 FE 0D
0F30:4C F9 0E 8D 10 C0 60 E0
0F38:00 20 00 E0 80 20 80 E0
0F40:28 20 28 E0 A8 20 A8 E0
0F48:50 20 50 E0 D0 20 D0 D0
0F50:00 D0 00 A2 00 20 2F 11
0F58:20 AA 0F A2 04 20 2F 11
0F60:20 AA 0F A2 08 20 2F 11
0F68:20 AA 0F A2 0C 20 2F 11
0F70:20 AA 0F A2 10 20 2F 11
0F78:20 AA 0F A2 14 20 2F 11
0F80:20 AA 0F AD 62 C0 10 01
0F88:60 A2 18 20 2F 11 20 F6
0F90:0F A5 06 18 69 28 90 0A
0F98:E6 07 A6 07 E0 DF F0 09
0FA0:E6 09 85 06 85 08 4C 8E
0FA8:0F 60 A2 20 A0 00 B1 06
0FB0:8D 0F 0B C8 B1 06 88 91
0FB8:06 C8 C0 27 90 F5 AD 62
0FC0:C0 10 08 AD 0F 0B 91 06
0FC8:4C EE 0F AD 83 C0 AD 83
0FD0:C0 A0 00 B1 08 A0 27 91
0FD8:06 A0 00 C8 B1 08 88 91
0FE0:08 C8 C0 27 90 F5 AD 0F
0FE8:0B 91 08 AD 81 C0 E6 07
0FF0:E6 09 CA D0 B7 60 AD 8B
0FF8:C0 AD 8B C0 A0 00 B1 06
1000:8D 0F 0B C8 B1 06 88 91
1008:06 C8 C0 27 90 F5 AD 83
1010:C0 AD 83 C0 A0 00 B1 08
1018:8D 0A 0B AD 8B C0 AD 8B
1020:C0 AD 0A 0B A0 27 91 06
1028:A0 00 AD 83 C0 AD 83 C0
1030:C8 B1 08 88 91 08 C8 C0
1038:27 90 F5 AD 0F 0B 91 08
1040:AD 81 C0 60 A2 00 20 2F
```

```
1048:11 20 9B 10 A2 04 20 2F
1050:11 20 9B 10 A2 08 20 2F
1058:11 20 9B 10 A2 0C 20 2F
1060:11 20 9B 10 A2 10 20 2F
1068:11 20 9B 10 A2 14 20 2F
1070:11 20 9B 10 AD 62 C0 10
1078:01 60 A2 18 20 2F 11 20
1080:E5 10 A5 06 18 69 28 90
1088:0A E6 07 A6 07 E0 DF F0
1090:09 E6 09 85 06 85 08 4C
1098:7F 10 60 A2 20 AD 62 C0
10A0:10 0A A0 27 B1 06 8D 0F
10A8:0B 4C CF 10 AD 83 C0 AD
10B0:83 C0 A0 27 B1 08 8D 0F
10B8:0B 88 B1 08 C8 91 08 88
10C0:D0 F7 A0 27 B1 06 A0 00
10C8:91 08 AD 81 C0 A0 27 88
10D0:B1 06 C8 91 06 88 D0 F7
10D8:AD 0F 0B 91 06 E6 07 E6
10E0:09 CA D0 B9 60 AD 83 C0
10E8:AD 83 C0 A0 27 B1 08 8D
10F0:0F 0B 88 B1 08 C8 91 08
10F8:88 D0 F7 AD 8B C0 AD 8B
1100:C0 A0 27 B1 06 8D 0A 0B
1108:AD 83 C0 AD 83 C0 AD 0A
1110:0B A0 00 91 08 AD 8B C0
1118:AD 8B C0 A0 27 88 B1 06
1120:C8 91 06 88 D0 F7 AD 0F
1128:0B 91 06 AD 81 C0 60 BD
1130:37 0F 85 09 BD 38 0F 85
1138:08 BD 39 0F 85 07 BD 3A
1140:0F 85 06 60 20 00 1E BF
1148:1F 80 20 00 20 28 23 80
1150:1F A8 20 28 20 50 23 A8
1158:1F D0 20 50 23 D0 DD C0
1160:DE 00 DC C0 D0 00 1E BF
1168:E0 00 1E BF DF 80 E0 00
1170:E0 28 E3 80 DF A8 E0 28
1178:E0 50 E3 A8 DF D0 E0 50
1180:E3 D0 DD C0 23 D0 1E BF
1188:40 50 3F D0 23 A8 20 50
1190:40 28 3F A8 23 80 20 28
1198:40 00 3F 80 20 00 D0 00
11A0:D0 00 D1 40 DD C0 1E BF
11A8:E3 D0 1E BF 00 50 FF D0
11B0:E3 A8 E0 50 00 28 FF A8
11B8:E3 80 E0 28 00 00 FF 80
11C0:E0 00 D0 00 23 D0 1E BF
11C8:20 00 1E BF A2 00 20 28
11D0:13 20 BA 13 A2 04 20 28
11D8:13 20 55 13 A2 08 20 28
11E0:13 20 9B 13 A2 0C 20 28
11E8:13 20 55 13 A2 10 20 28
11F0:13 20 9B 13 A2 14 20 28
11F8:13 20 55 13 AD 62 C0 10
1200:09 A2 80 20 28 13 20 C5
1208:13 60 AD 8B C0 AD 8B C0
1210:A2 18 20 28 13 20 C5 13
1218:A2 1C 20 28 13 20 EB 13
1220:A2 20 20 28 13 20 15 14
1228:AD 83 C0 AD 83 C0 A2 24
1230:20 28 13 20 BA 13 A2 28
1238:20 28 13 20 55 13 A2 2C
1240:20 28 13 20 9B 13 A2 30
1248:20 28 13 20 55 13 A2 34
1250:20 28 13 20 9B 13 A2 38
1258:20 28 13 20 55 13 A2 3C
1260:20 28 13 20 C5 13 A2 1C
1268:20 28 13 20 EB 13 A2 20
1270:20 28 13 20 15 14 AD 81
1278:C0 60 A2 40 20 28 13 20
1280:BA 13 A2 44 20 28 13 20
1288:78 13 A2 48 20 28 13 20
1290:9B 13 A2 4C 20 28 13 20
1298:78 13 A2 50 20 28 13 20
12A0:9B 13 A2 54 20 28 13 20
12A8:78 13 AD 62 C0 10 09 A2
12B0:84 20 28 13 20 C5 13 60
12B8:AD 8B C0 AD 8B C0 A2 58
12C0:20 28 13 20 C5 13 A2 5C
12C8:20 28 13 20 02 14 A2 60
12D0:20 28 13 20 15 14 AD 83
12D8:C0 AD 83 C0 A2 64 20 28
12E0:13 20 BA 13 A2 68 20 28
12E8:13 20 78 13 A2 6C 20 28
12F0:13 20 9B 13 A2 70 20 28
12F8:13 20 78 13 A2 74 20 28
1300:13 20 9B 13 A2 78 20 28
1308:13 20 78 13 A2 7C 20 28
1310:13 20 C5 13 A2 5C 20 28
1318:13 20 02 14 A2 60 20 28
1320:13 20 15 14 AD 81 C0 60
1328:BD 44 11 85 09 BD 45 11
1330:85 08 BD 46 11 85 07 BD
1338:47 11 85 06 60 A0 00 B1
1340:08 91 06 C8 C0 28 90 F7
1348:60 A0 00 B1 06 91 08 C8
1350:C0 28 90 F7 60 A9 08 8D
1358:0F 0B 20 65 13 E6 07 CE
1360:0F 0B D0 F6 60 A2 04 E6
1368:09 20 3D 13 CA F0 08 E6
1370:07 20 49 13 4C 67 13 60
1378:A9 08 8D 0F 0B 20 88 13
1380:C6 07 CE 0F 0B D0 F6 60
1388:A2 04 C6 09 20 3D 13 CA
1390:F0 08 C6 07 20 49 13 4C
1398:8A 13 60 A9 08 8D 0F 0B
13A0:20 3D 13 CE 0F 0B F0 11
13A8:A5 07 18 69 04 85 07 A5
13B0:09 18 69 04 85 09 4C A0
13B8:13 60 A2 08 20 3D 13 20
13C0:D0 13 4C BC 13 A2 08 20
13C8:49 13 20 D0 13 4C C7 13
13D0:CA F0 15 A5 09 18 69 04
13D8:85 09 A5 06 18 69 28 90
13E0:02 E6 07 85 06 4C EA 13
13E8:68 68 60 A2 0E A0 FF B1
13F0:06 91 08 88 D0 F9 B1 06
13F8:91 08 C6 07 C6 09 CA D0
1400:EC 60 A2 0E A0 00 B1 06
1408:91 08 C8 D0 F9 E6 07 E6
1410:09 CA D0 F0 60 A0 00 B1
1418:06 91 08 C8 D0 F9 E6 07
1420:E6 09 B1 06 91 08 C8 C0
1428:40 D0 F7 60 AD 0A 0B 8D
1430:0C 0B A9 00 8D 0B 0B 8D
1438:0D 0B 8D 0E 0B 20 6A 14
1440:18 AD 0D 0B 6D 0C 0B 8D
1448:0D 0B 90 06 EE 0E 0B 20
1450:6A 14 38 AD 0B 0B ED 0E
1458:0B 8D 0B 0B B0 E2 CE 0C
1460:0B 20 6A 14 AD 0C 0B D0
1468:D7 60 AD 07 0B 18 6D 0E
1470:0B B0 07 C9 C0 B0 03 20
1478:83 14 38 AD 07 0B ED 0E
1480:0B 90 4B 48 AD 05 0B 18
1488:6D 0C 0B B0 12 AC 06 0B
1490:F0 04 C9 18 B0 19 AA 68
1498:48 20 57 F4 4C AF 14 AC
14A0:06 0B D0 0B C9 18 B0 07
14A8:AA C8 68 48 20 57 F4 38
14B0:AD 05 0B ED 0C 0B 90 0B
14B8:AC 06 0B AA 68 20 57 F4
14C0:4C CE 14 AA 68 AC 06 0B
14C8:F0 04 88 20 57 F4 60
```

Récapitulation GRAV.C

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE GRAV.C, A$801,L$25F

```
0801:A5 CE 20 ED FD 60 20
0808:93 09 A9 1B 20 81 09 A9
0810:6E 20 81 09 A5 07 29 08
0818:F0 03 4C 3C 0A A5 07 6A
0820:6A 6A 29 C0 85 07 A9 20
0828:90 02 A9 40 85 1B A0 00
0830:84 1A 84 09 A9 0D 20 81
0838:09 A9 1B 20 81 09 A9 5A
0840:20 81 09 A9 00 20 81 09
0848:A9 20 20 81 09 A9 1B 20
0850:81 09 A9 54 20 81 09 A9
0858:31 20 81 09 A9 36 20 81
0860:09 A9 1B 20 81 09 A9 3E
0868:20 81 09 A9 FF 85 19 A6
0870:1B A4 1A D0 0C 8A 29 1F
0878:D0 07 66 07 B0 62 38 26
0880:07 8A 48 98 48 A9 1B 20
0888:81 09 A9 47 20 81 09 A9
0890:30 20 81 09 24 07 10 0D
0898:A9 35 20 81 09 A9 36 20
08A0:81 09 38 B0 0A A9 32 20
08A8:81 09 A9 38 20 81 09 A9
08B0:30 20 81 09 A9 01 85 08
08B8:E6 19 A4 19 C0 28 90 0E
08C0:A9 0D 20 81 09 A9 0A 20
08C8:81 09 68 68 10 9D 68 AA
08D0:85 1A 68 85 1B 48 8A 48
08D8:A9 00 85 06 A2 08 D0 06
08E0:B0 74 30 D0 10 E8 A4 19
08E8:B1 1A 25 08 C9 01 66 06
08F0:CA 24 07 10 08 A5 09 49
08F8:01 85 09 D0 E9 18 A5 1B
0900:69 04 A0 1C 84 09 24 09
0908:D0 25 06 1A 90 1D 69 E0
0910:18 A0 04 84 09 24 09 F0
0918:14 A5 1A 69 50 49 F0 F0
0920:02 49 F0 85 1A A5 1B 29
0928:E0 90 02 69 E0 66 1A 85
0930:1B A9 00 85 09 E0 00 D0
0938:AD A5 06 24 07 70 02 49
0940:FF 20 81 09 24 07 10 08
0948:A5 09 49 01 85 09 D0 E9
0950:06 08 30 8E 10 8E 85 07
0958:A9 1B 20 81 09 A9 41 20
0960:81 09 A9 0D 20 81 09 A9
0968:0A 20 81 09 A9 1B 20 81
0970:09 A9 44 20 81 09 A9 00
0978:20 81 09 A9 20 20 EA 03
```

```
0980:60 A4 FC A6 1C 6C FA 00
0988:99 80 C0 B9 84 C0 29 80
0990:F0 F9 60 A5 06 0A 0A 0A
0998:0A 85 FC A9 00 85 1B A5
09A0:06 29 07 09 C0 85 1C A0
09A8:05 B1 1B C9 38 D0 3D A0
09B0:07 B1 1B C9 18 D0 35 A0
09B8:0B B1 1B C9 01 F0 02 D0
09C0:2B A0 0D B1 1B 85 FA A5
09C8:1C 85 FB 20 81 09 A0 0F
09D0:B1 1B 85 FA A0 0C B1 1B
09D8:C9 31 D0 0F A9 09 20 81
09E0:09 A9 5A 20 81 09 A9 0D
09E8:20 81 09 60 A0 0A B1 1B
09F0:C9 20 D0 17 C8 B1 1B C9
09F8:58 D0 10 C8 B1 1B C9 FF
0A00:D0 09 A9 81 85 FA A9 91
0A08:85 FB 60 AD 68 FA C9 C0
0A10:D0 18 A0 0B B1 1B C9 08
0A18:D0 10 C8 B1 1B C9 48 D0
0A20:09 A9 26 85 FA A9 92 85
0A28:FB 60 4C 02 0A 48 AD F1
0A30:C0 49 10 29 70 D0 F7 68
0A38:8D F0 C0 60 A9 59 85 FE
0A40:A9 92 85 FF A0 00 B1 FE
0A48:C9 FF F0 13 20 81 09 18
0A50:A5 FE 69 01 85 FE A5 FF
0A58:69 00 85 FF 38 B0 E5 60
```

Récapitulation CHARGE.OBJ

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE CHARGE.OBJ, A$9000,L$22B

```
9000:A9 00 8D F6 03 A9 9B 8D
9008:F7 03 A9 4C 8D F5 03 A2
9010:00 BD 2B 90 9D 00 9B E8
9018:D0 F7 BD 2B 91 9D 00 9C
9020:E8 D0 F7 A9 9A 8D 01 9D
9028:4C D4 A7 A5 1E 8D AC AA
9030:A5 1F 8D AD AA A0 00 8C
9038:EB B7 20 B7 00 C9 22 D0
9040:23 A9 06 85 C1 20 B1 00
9048:C9 22 F0 0C 09 80 99 75
9050:AA C8 C0 1E 90 EF B0 34
9058:A9 EF 85 C1 20 B1 00 C0
9060:00 D0 4C 60 A5 B8 48 A5
9068:B9 48 20 7B DD 24 11 30
9070:08 A2 A3 20 F5 9C 4C 12
9078:D4 68 85 B9 68 85 B8 20
9080:E3 DF 85 1E 84 1F A0 00
9088:B1 1E D0 05 A2 0B 4C 0A
9090:9C 8D AB AA C8 B1 1E 48
9098:C8 B1 1E 85 1F 68 85 1E
90A0:A0 00 B1 1E 09 80 99 75
90A8:AA C8 CC AB AA D0 F3 A9
90B0:A0 99 75 AA C8 C0 1E D0
90B8:F8 8D B0 AA 20 F5 9C 20
90C0:B7 00 F0 12 20 BE DE 20
90C8:67 DD 20 52 E7 A5 51 C9
90D0:02 90 B9 4E B0 AA A9 01
90D8:8D F4 B7 A9 10 8D ED B7
90E0:A9 11 8D EC B7 20 E6 9C
90E8:CE ED B7 F0 46 20 CB 9C
90F0:A0 EB 8C AE AA AD AE AA
90F8:18 69 23 A8 8D AE AA C9
9100:03 F0 E5 A2 00 B9 BB B3
9108:F0 29 DD 75 AA D0 E6 C8
9110:E8 E0 1E D0 F0 B9 9C B3
9118:29 04 D0 04 A2 0D D0 15
9120:AC AE AA B9 B9 B3 8D ED
9128:B7 B9 B8 B3 30 C7 8D EC
9130:B7 D0 0E A2 06 8E 5C AA
9138:20 F5 9C 20 93 FE 4C D5
9140:A6 8D AF AA 20 CB 9C A2
9148:0C A5 1E A4 1F 20 EE 9C
9150:AD AF AA F0 03 20 EA 9C
9158:20 BB 9C AD AF AA F0 40
9160:2C B0 AA 30 0A A5 50 8D
9168:BB B4 A5 51 8D BC B4 38
9170:AD BB B4 E9 04 85 1E 8D
9178:F0 B7 AD BC B4 E9 00 85
9180:1F 8D F1 B7 A9 04 8D DA
9188:9C 18 6D BD B4 8D AB AA
9190:A9 00 8D E2 9C 8D AF AA
9198:AD C9 B3 F0 22 20 D9 9C
91A0:AE AE AA EE F1 B7 E6 1F
91A8:E8 E0 FE B0 1D BD BD B3
91B0:D0 A6 20 EA 9C 20 BB 9C
91B8:A9 00 85 48 8D DA 9C AD
91C0:AB AA 8D E2 9C 20 D9 9C
91C8:F0 56 AD BC B3 F0 E3 20
91D0:BB 9C AD BD B3 8D ED B7
91D8:E6 1F 20 E6 9C AD BC B3
91E0:8D EC B7 4C 19 9C BD BB
91E8:B3 8D EC B7 E8 BD BB B3
91F0:8D ED B7 8E AE AA A9 B7
91F8:A0 E8 20 B5 B7 90 11 A2
9200:08 4C 0A 9C A0 00 B9 BB
9208:B4 91 1E C8 C0 00 D0 F6
9210:60 A0 B3 D0 02 A0 B4 A9
9218:BB 8D F0 B7 8C F1 B7 60
9220:AD AC AA 85 1E AD AD AA
9228:85 1F 60
```

Récapitulation ERREUR

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE ERREUR, A$3A3,L10

```
03A3:68 A8 68 A6 DF
03A8:9A 48 98 48 60
```

Récapitulation FONC.C

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE SOURIS.C, A$301,L$A2

```
0301:A9 F0 85 06 4C 0C 03
0308:A9 0B 85 06 A9 60 85 07
0310:A5 B8 48 A5 B9 48 20 2C
0318:D5 AE 00 02 F0 19 A2 00
0320:86 13 20 6D D5 A9 B2 91
0328:06 88 A9 3A 91 06 88 B9
0330:FC 01 91 06 88 D0 F8 68
0338:85 B9 68 85 B8 60 AD 83
0340:C0 AD 83 C0 A9 00 85 08
0348:85 06 A9 E0 85 07 A9 20
0350:85 09 A2 20 60 20 3E 03
0358:A0 00 B1 08 11 06 91 08
0360:88 D0 F7 E6 07 E6 09 CA
0368:D0 EE AD 81 C0 60 A9 00
0370:85 08 85 06 A9 40 85 07
0378:A9 20 85 09 A2 20 A0 00
0380:B1 08 8D 00 03 B1 06 91
0388:08 AD 00 03 91 06 88 D0
0390:EF E6 07 E6 09 CA D0 E6
0398:60 20 3E 03 20 7E 03 AD
03A0:81 C0 60
```

Récapitulation DHEL.C

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE HEL.C, A$300,L$36

```
0300:A9 D0 85 09 A9 00 85 08
0308:A2 10 AD 8B C0 AD 8B C0
0310:20 27 03 A9 D0 85 09 A9
0318:00 85 08 A2 30 AD 81 C0
0320:AD 81 C0 20 27 03 60 A9
0328:00 A0 00 91 08 88 D0 FB
0330:E6 09 CA D0 F4 60
```

La qualité de ce programme et son volume le destinaient à être vendu par Pom's en "produits édités à part".
Nous avons préféré vous faire profiter de ses nombreuses possibilités en l'insérant dans ces pages ; la place nous manquant pour les sources, seuls les fichiers nécessaires au fonctionnement sont listés ici.
Les lecteurs qui souhaitent approfondir la programmation ou adapter le logiciel à leurs besoins trouveront cette fois les sources sur la disquette d'accompagnement.

GOGO : les sauts du Basic

Patrice Neveu

Qui n'a jamais eu de problème à relire un programme complexe, constitué de nombreuses sous-routines et donc truffé de GOTOs et GOSUBs, voire de RUNs ? En effet, si beaucoup de Basics autorisent les sauts vers des routines précédées d'un nom, d'une étiquette, l'Applesoft n'accepte que les sauts spécifiés par numéros de ligne, d'où des listings souvent très hermétiques.

Le but de cet article est donc de modifier les commandes GOTO, GOSUB et RUN par l'intermédiaire de l'ampersand (&), afin d'obtenir une plus grande facilité pour réaliser et maintenir les programmes.

Fonctionnement

Bien que le listing soit entièrement commenté, il est nécessaire de présenter certains sous-programmes de la ROM Applesoft :

CHRGET : positionne TXTPTR sur le caractère suivant, puis va le lire. Lorsque c'est un espace, il recommence. On peut donc faire précéder le début du REM d'autant de blancs que l'on voudra, sans pour autant devoir les remettre dans l'étiquette qui suit le GOTO ou le GOSUB. D'autre part, CHRGET signale les fins d'instruction et donc aussi les fins de ligne de programme en mettant la retenue (carry) à 1.

DATA : se place à la fin de l'instruction en cours. Si on fait un JMP, il permet d'exécuter la prochaine instruction. Ainsi, on obtient COUCOU dans l'exemple suivant avec un saut & GOTO "TRUC :

```
100 PRINT "SALUT": REM TRU
    C
110 PRINT "COUCOU"
```

GOTO : va tout d'abord lire le numéro de ligne pointé par TXTPTR, le range en LINNUM, puis continue par GOTOBIS, qui recherche la ligne stockée en LINNUM. GOGO se branche donc automatiquement en GOTOBIS, après avoir stocké en LINNUM le numéro de la ligne suivant le REM voulu.

GOSUB : place d'abord sur la pile du 6502 la valeur de TXTPTR (2 octets) puis de CURLIN (2 octets) qui est le numéro de la ligne en cours d'exécution et enfin le type de commande ($B0 pour GOSUB). Puis GOSUB appelle GOTO et exécute la ligne par NEWSTT ($D7D2). Le RETURN reprendra les informations de la pile pour savoir où il faut reprendre le programme. POP les supprimera tout simplement, sans les utiliser.

Utilisation

Il suffit de modifier légèrement vos lignes de façon à répondre à la syntaxe suivante :

```
10  & GOSUB "ROUTINE"
20  & GOTO "FIN
100 REM ROUTINE
110 ...
120 RETURN
500 REM FIN DU PROGRAMME
```

Attention, après GOTO, GOSUB ou RUN, il est indispensable d'inscrire les étiquettes à l'intérieur des guillemets (ou, au minimum, de les faire précéder d'un guillemet) car l'Applesoft le décomposerait (par exemple CAT en C AT, AT étant un mot-clé). Toutefois, la routine offre de la souplesse à l'utilisation puisqu'il est possible d'appeler un :

```
REM Routine d'ouverture
```

par un :

```
& GOSUB Routine
```

ou même par un :

```
& GOSUB Rout.
```

Le retour d'un sous-programme se fait par le RETURN courant, ce qui permet aussi l'application de POP.

Bien entendu, ce qui est valable pour la commande GOSUB l'est également pour les GOTOs et les RUNs.

Ces trois commandes sont aussi utilisables depuis le clavier vers le programme directement.

Dans cas où la syntaxe ne serait pas respectée peut s'afficher l'un des message suivant :

- UNDEF'D STATEMENT ERROR (l'ampersand est omis)
- IL FAUT UN GOTO, GOSUB ou RUN (après l'ampersand)
- IL FAUT UN GUILLEMET (avant le label appelé)
- LE LABEL N'EXISTE PAS (aucun REM ne commence par le label)

Enfin, le programme source

T.GOGO se trouve sur la disquette Pom's au format TEXT, assemblable directement par Merlin/Big Mac.

Pour conclure ce mode d'emploi, deux remarques :

- Afin d'éviter des catastrophes avec le Compacteur de Sylvie Gallet, (Pom's 22), il est nécessaire de supprimer les lignes 287 et 288. En effet, le Compacteur détruit tout ce qui suit le REM, ce qui nuirait naturellement au bon fonctionnement de GOGO ;
- GOGO est opérationnel aussi bien sous DOS 3.3 que sous ProDOS.

Programme GOGO.DEMO

```
10  PRINT  CHR$ (4)"BRUN GOGO
1000  REM   INITIALISATION
1010  TEXT : NORMAL : HOME
1020  INVERSE : PRINT "  DES SAUTS A GOGO,
      PROGRAMME DE DEMO.  ": NORMAL : POKE
      34,2
1030  VTAB 10
1040  PRINT " 1) LISTING DU PROGRAMME
1050  PRINT : PRINT " 2) CATALOGUE DU DISK
1060  PRINT : PRINT " 3) QUITTER
2000  REM   CHOIX DU MENU
2010  GET A$:R =  VAL (A$): IF R = 0 THEN
      &  GOTO "CHOIX
2020  & R GOSUB "LIST","CAT","QUIT"
2025  GET R$
2030  &  RUN "INIT
3000  REM  LISTING
3010  HOME : LIST : RETURN
4000  REM  CATALOG
4010  HOME : PRINT
4020  IF  PEEK (48896) = 76 THEN  PRINT  C
     HR$ (4)"CAT": RETURN
4030  PRINT  CHR$ (4)"CATALOG": RETURN
5000  REM  QUIT
5010  END
```

Source T.GOGO

Assembleur Big Mac

En format TEXT sur la disquette Pom's pour récupération par tout assembleur

```
        ORG  $8000

*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*
*                    Commandes disponibles                          *
*                                                                   *
*  &GOTO "label"           : Saute après la ligne REM qui           *
*                            contient le label.                     *
*  &GOSUB "label"          : Va au sous-programme commençant        *
*                            après la ligne REM qui contient        *
*                            le label.                              *
*  &RUN "label"            : Execute le programme là où se          *
*                            situe le label.                        *
*                                                                   *
*  & VR GOTO  "11","12"    : Saute au label d'ordre VR              *
*  & VR GOSUB "11","12"    : Va au sous-programme d'ordre VR        *
*  & VR RUN   "11","12"    : Run indexé d'ordre VR                  *
*                                                                   *
*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*


LINNUM  =   $50       ; Numéro de la ligne où l'on va.
TXTTAB  =   $67       ; Adresse du début du texte Basic.
LOMEM   =   $69       ; Adresse de la fin du texte Basic.
ARYTAB  =   $6B       ; Fin des variables simples
HIMEM   =   $73       ; Plus haute adresse disponible, buffer ProDOS
CURLIN  =   $75       ; Numéro de la ligne en cours d'execution.
LASTVAR =   $81       ; Nom de la variable à rechercher (index)
VARPNT  =   $83       ; Adresse de la valeur de la variable (index)
CHRGET  =   $B1       ; Sous-routine de prise de caractère.
CHRGOT  =   $B7       ; Prend le caractère actuel.
TXTPTR  =   $B8       ; Pointe sur le caractère à prendre.

TEMP    =   $D7       ; Stockage temporaire
PGNUM   =   $CF       ; Pointeur dans la table des pages à protéger

BUFFER  =   $0280     ; Buffer pour copier l'instruction après &.
CHRGET1 =   $0300     ; Adresse de CHRGET0 relogée
RELOCER =   $0319     ; Message d'erreur lors du relogement
ERROR   =   $033F     ; Message d'erreur de commande
ERROR0  =   $035D     ; Message de manque de guillemet
ERROR2  =   $0375     ; Message de label inconnu
CMD     =   $038E     ; Tokens de GOTO GOSUB RUN
LENGTH  =   $039C     ; Longueur de la commande &
TXTPTR2 =   $039D     ; Sauvegarde de TXTPTR pour les GOSUB
COMMANDE =  $039F     ; GOTO GOSUB ou RUN
PLACE   =   $03A0     ; Emplacement du label dans la ligne
INDEX   =   $03A1
BASIC   =   $03D0     ; Retour au Basic.
AMPERV  =   $03F5     ; Vecteur de l'&.

GETBUFR =   $BEF5     ; Pour déplacer les buffers.
BITMAP  =   $BF58     ; Bitmap en ProDOS

CLEARC  =   $D66C     ; Mise à zero de la pile et des variables.
NEWSTT  =   $D7D2     ; Execute une nouvelle instruction Basic.
GOTOBIS =   $D955     ; Partie de GOTO ($D93E): pas de LINGET ($DA0C)
DATA    =   $D995     ; Met TXTPTR à la fin de l'instruction Basic.
STROUT  =   $DB3A     ; Routine de sortie de chaine de caractères.
GETADR  =   $E752     ; LINNUM (entier à 2 octets) <- FAC
MOVFM   =   $EAF9     ; FAC <- (Y,A)
TEXT    =   $FB39
HOME    =   $FC58

GOTO    =   $AB       ; Token de GOTO
RUN     =   $AC       ; Token de RUN
GOSUB   =   $B0       ; Token de GOSUB
REM     =   $B2       ; Token de REM

*-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-*
* Cette routine  s'adapte à l'environnement qu'elle trouve:          *
*                                                                    *
* Si le Système d'Exploitation du Disque est le DOS 3.3, elle        *
* reste en $9400, là où elle a été chargée.                          *
*                                                                    *
* Si le SED est le ProDOS elle abaisse les buffers de deux           *
* pages et s'installe entre eux et le Basic.System, puis             *
* protège ces pages.                                                 *
*                                                                    *
* Dans les deux cas, la page 3 est occupée par des routines          *
* des tables et des messages pour des problèmes de                   *
* relogeabilité. Elle est également protégée.                        *
*-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-*


* On se trouve sous ProDOS ?

DEBUT   LDA  $BF00     ; C'est le vecteur JMP du MLI
        CMP  £$4C
        BEQ  GBUFF
        JMP  AMPERSET

* On se place entre le Basic.System et les buffers de ProDOS.

GBUFF   LDA  £3        ; Descend les buffers du DOS de 3 pages
        JSR  GETBUFR   ; Pour qu'on puisse se protéger
        BCC  MOVER     ; Le réserve de place s'est faite sans pb

* Erreur lors d'un des déplacement

ERELOC  JSR  TEXT
        JSR  HOME
        LDA  £<RELOCER
        LDY  £>RELOCER
        JSR  STROUT
WAIT    JMP  WAIT

MOVER   EQU  *         ; Prépare le déménagement de la routine
        STA  PROGADR+2
        STA  PROTECT+1
        STA  AMPINIT+1
        TAX
        INX
        STX  PROTECT+2
        INX
        STX  PROTECT+3
```

```
        LDA  £0
        STA  PROGADR+1

MOVE    LDA  GOGO       ; Chargement de l'octet pour le
PROGADR STA  $FFFF      ; remettre à l'endroit désigné par ProDOS
        INC  MOVE+1     ; Passe au prochain octet
        BNE  MOVE0
        INC  MOVE+2
MOVE0   LDA  MOVE+2     ; Vérifie si on a terminé
        CMP  £>FIN
        BCC  MOVE1
        BNE  MOVE3
        LDA  MOVE+1
        CMP  £<FIN
        BCC  MOVE1
        BNE  MOVE3
MOVE1   INC  PROGADR+1
        BNE  MOVE
        INC  PROGADR+2
        BNE  MOVE
MOVE3   EQU  *

* On protège la page $03 et celles du programme relogé.

        LDA  £3         ; Il y a 4 pages à protéger
        STA  PGNUM      ; on se sert d'un compteur
NXTPG   LDY  PGNUM      ; Quel est le numéro de la page à protéger ?
        LDA  PROTECT,Y  ; Il est dans la table
        PHA             ; Sauvegarde temporaire
        LSR             ; On divise par huit pour déterminer l'octet
        LSR             ; Où se trouve le bit de la page recherchée
        LSR
        TAX             ; Dans X pour aller dans la table du ProDOS
        LDA  BITMAP,X   ; ( page 16 du pom's 22 )
        STA  TEMP       ; On le sauve pour usage ultérieur
        PLA             ; Reprise du n° de page
        AND  £7         ; Dont on ne garde que le bit de position
        TAY             ; Qui sert à chercher le masque
        LDA  BITMASK,Y  ; Pour mettre à 1 le bit de la page occupée
        BIT  TEMP       ; Verifie se c'est déja occupé
        BNE  ERELOC
        ORA  TEMP
        STA  BITMAP,X   ; Puis on met à jour la BitMap
        DEC  PGNUM      ; Et on passe à la prochaine page
        BPL  NXTPG

* Déplacement d'une sous-routine du programme pour rester
* relogeable en   environnement ProDOS

        LDY  £144       ; Nombre d'octets à mettre en $0300
MOVERTN LDA  CHRGET0,Y  ; Ce sont essentiellement des tables
        STA  CHRGET1,Y  ; et des messages, plus deux routines
        DEY             ; appelées par JSR et donc non-relogeables.
        CPY  £$FF
        BNE  MOVERTN

* Initialise le vecteur ampersand sous ProDOS

AMPINIT LDA  £$FF       ; L'ampersand ira au début du buffer donné
        LDY  £0         ; par ProDOS
        STA  AMPERV+2
        STY  AMPERV+1
        RTS

* Initialise le vecteur ampersand sous DOS 3.3

AMPERSET LDA  £>GOGO    ; Maintenant, l'& ira directement à GOtoGOsub.
        LDY  £<GOGO
        STA  AMPERV+2
        STY  AMPERV+1

        LDA  £>DEBUT    ; Fixe HIMEM sous DOS 3.3 pour éviter les
        LDY  £<DEBUT    ; variables intempéstives
        STA  HIMEM+1
        STY  HIMEM

* Déplacement d'une sous-routine du programme en DOS 3.3

        LDY  £144       ; Nombre d'octets à mettre en $0300
MOVERTN1 LDA  CHRGET0,Y ; Ce sont essentiellement des tables
        STA  CHRGET1,Y  ; et des messages, plus deux routines
        DEY             ; appelées par JSR et donc non-relogeables.
        CPY  £$FF
        BNE  MOVERTN1

        RTS             ; Retour au Basic.

*-----------------------------------------
*       Vérification de la syntaxe des commandes.
*-----------------------------------------

GOGO    EQU  *

        LDX  £1
        JSR  CHRGOT     ; Necessaire pour prendre le 1er caractère
        STA  BUFFER     ; suivant l'&. On stocke le résultat en BUFFER.
GETCMD  JSR  CHRGET     ; Boucle pour lire et stocker la suite de la
        BEQ  IDENTIFY   ; commande. A=0 si fin d'instruction.
        STA  BUFFER,X
        INX
        BNE  GETCMD

IDENTIFY EQU  *         ; Pour identifier la commande envoyée.
        DEX             ; On décremente la longueur de la commande de &
        STX  LENGTH     ; puis on la range pour usage ultérieur.
        LDX  £2         ; Il y a trois commandes possibles. On commence
IDENTIFY1 LDA CMD,X     ; par la dernière.
        CMP  BUFFER     ; Et on compare au token rangé dans le buffer.
        BNE  IDENTIFY2
        JMP  LEGAL      ; Si c'est autorisé, d'accord...
IDENTIFY2 EQU *
        DEX             ; Sinon, passe à la prochaine !
        BPL  IDENTIFY1  ; Et il n'y en a que 3...

        LDX  £0         ; On a peut etre un saut indexé
ISINDEX LDA  BUFFER,X   ; Alors sachant que la variable est au début,
        CMP  £GOTO      ; on recherche le GOTO,GOSUB,RUN
        BEQ  YAINDEX    ; C'est bien un saut indexé
        CMP  £GOSUB
        BEQ  YAINDEX
        CMP  £RUN
        BEQ  YAINDEX
        INX             ; Recherche d'un saut quelconque sur toute
        CPX  LENGTH     ; la longueur de la ligne
        BNE  ISINDEX

        LDY  £>ERROR    ; Si c'est l'inconnu, alors signale l'erreur
        LDA  £<ERROR    ; de commande.
        JSR  STROUT
        JMP  BASIC      ; Et retourne au basic.

* Lecture de l'index

YAINDEX EQU  *
        PHA
        TXA
        PHA
        LDY  £0         ; Lecture de la 1ère lettre de la variable
        LDA  BUFFER,Y
        STA  LASTVAR
        CPX  £1         ; Il y a une ou plusieur lettre dans le nom ?
        BNE  SNDLET     ; Va prendre la 2nd
        LDA  £0
        STA  LASTVAR+1
        BEQ  SRCHVAR
SNDLET  EQU  *
        INY
        LDA  BUFFER,Y
        STA  LASTVAR+1
SRCHVAR EQU  *
        LDA  LOMEM      ; On se place au début de la zone des
        LDX  LOMEM+1    ; variables simples afin de rechercher
        STA  VARPNT     ; notre index
        STX  VARPNT+1

CMPVAR  EQU  *          ; Et on compare la variable étudiée avec celle
        LDY  £0         ; qu'on a trouvé après l'ampersand
        LDA  (VARPNT),Y
        CMP  LASTVAR
        SEC
        BNE  NXTVAR
        INC  VARPNT
        CLC
        BNE  SRCHVAR1
        INC  VARPNT+1
SRCHVAR1 EQU *
        PHP
        LDA  (VARPNT),Y
        CMP  LASTVAR+1
        BEQ  VARFND
        PLP
NXTVAR  EQU  *
        LDA  VARPNT
        ADC  £6         ; On rajoute 6 ou 7 celon ce que la 1ère lettre
        STA  VARPNT     ; était convenable ou non
        BCC  VRFYEOV    ; Va vérifier si on a vu toutes les variables
        INC  VARPNT+1
VRFYEOV EQU  *
        LDA  VARPNT+1   ; A-t-on terminé la zone des variables simples
        CMP  ARYTAB+1
        BCC  CMPVAR
        BNE  NOVAR
        LDA  VARPNT
        CMP  ARYTAB
        BCC  CMPVAR

        LDA  £0         ; La variable n'existe pas, donc 0
        STA  INDEX
        BEQ  VARFOUND

VARFND  PLA

VARFOUND EQU *
        LDY  VARPNT+1   ; Pour indiquer à FAC où se trouve la valeur
        LDX  VARPNT     ; de l'index
        INX
```

```
        TXA
        BNE  PUTINFAC
        INY
PUTINFAC EQU  *
        JSR  MOVFM      ; Met la variable en FAC
        JSR  GETADR     ; Puis convertit FAC en entier dans LINNUM
NOVAR   EQU  *
        LDA  LINNUM
        STA  INDEX      ; On a un index

        PLA
        TAX
        PLA
        BNE  LEGAL1

LEGAL   EQU  *          ; C'est bon,
        LDX  #0
        STX  INDEX      ; Il n'y a pas d'index
LEGAL1  EQU  *
        STA  COMMANDE   ; Alors range le type de commande qu'on a eu.
        INX             ; Puis verifie l'existence du guillemet avant
        LDA  BUFFER,X   ; le label.
        CMP  #$22       ; $22 est le code de "
        BEQ  TESTINDX   ; S'il y est, on peut aller chercher le label
*                                       aprés le REM.
NOGUILL EQU  *
        LDY  #>ERROR0   ; Sinon, c'est raté, on doit tout arreter.
        LDA  #<ERROR0
        JSR  STROUT
        JMP  BASIC

*       Teste s'il      faut indexer et si oui alors allons-y

TESTINDX EQU  *
        LDY  INDEX      ; On n'indexe que si on doit lire aprés le 1er
        CPY  #2
        BCC  READY      ; Sinon, on est déja pret...

SEEKCOMA EQU  *
        LDA  BUFFER,X   ; On recherche les virgules car elles marquent
        CMP  #','       ; la fin d'un label à rechercher: c'est un
        BEQ  COMA       ; séparateur dans la liste de label
        INX             ; Mais il faut quand meme rester dans les
        CPX  LENGTH     ; limites de la ligne à étudier
        BNE  SEEKCOMA
        RTS             ; Si l'index > nombre de labels, alors on sort

COMA    INX             ; Aprés une virgule, on doit trouver un "
        LDA  BUFFER,X
        CMP  #$22
        BNE  NOGUILL    ; Sinon c'est une erreur
        DEY             ; Passe au prochain de la liste,
        CPY  #1         ; et si on est au bon, alors il faut y aller
        BNE  SEEKCOMA
READY   EQU  *
        INX             ; On est pret à rechercher le label
        STX  PLACE      ; Sauve le début du label

* Recherche d'une instruction REM dans le texte.

        LDA  TXTPTR     ; Regarde bien où on en est dans le texte.
        LDX  TXTPTR+1   ; puis sauve pour plus tard peut-etre.
        STA  TXTPTR2
        STX  TXTPTR2+1

        LDY  TXTTAB     ; Fixe le pointeur au début du programme.
        DEY             ; ( moins 1 )
        LDX  TXTTAB+1
        STY  TXTPTR
        STX  TXTPTR+1
SEEKREM EQU  *
        JSR  CHRGET1    ; Va lire les 2 octets qui donnent l'adresse
        JSR  CHRGET1    ; de la prochaine ligne.

        JSR  CHRGET1    ; Passe aussi les 2 octets du numéro de ligne.
        STA  LINNUM     ; et en passant, on stocke.
        JSR  CHRGET1
        STA  LINNUM+1

SEEKREM2 JSR  CHRGET1   ; Puis on lit le début de l'instruction.
        CMP  #REM       ; Est-ce un REM ?
        BEQ  READREM    ; Oui ! alors on va aller le lire.
READLINE JSR  CHRGET1   ; Non alors on passe à la prochaine instruction
        BNE  READLINE
        CMP  #00        ; Fin de ligne ?
        BEQ  SEEKREM    ; Oui, va sauter ce qu'il y a avant le texte.
        BNE  SEEKREM2   ; Non alors teste un REM.

FINLAB  EQU  *
        LDY  BUFFER,X
        CPY  #','
        BEQ  EXECLIN
        CPY  #$22
        BEQ  EXECLIN
        BNE  READLINE

READREM EQU  *
        LDX  PLACE      ; Le début du label

COMPREM JSR  CHRGET1    ; Il faut le comparer avec ce qui suit le REM
        BEQ  FINLAB     ; Fin du REM ?
        LDY  BUFFER,X
        CPY  #','       ; Dans une liste, une virgule ou des
        BEQ  EXECLIN    ; guillemets signifient une fin de label à
        CPY  #$22       ; rechercher.
        BEQ  EXECLIN
        CMP  BUFFER,X
        BNE  READLINE   ; Mais c'est différent, donc...
        INX             ; Si c'est identique, essaye plus loin !
        CPX  LENGTH     ; Mais ce n'est pas la peine d'en faire trop.
        BCC  COMPREM

* Execute cette ligne là

EXECLIN EQU  *
        LDA  COMMANDE   ; Selon ce que c'est, on n'agit pas pareil:
        CMP  #GOTO
        BEQ  EXECGOTO
        CMP  #RUN
        BEQ  EXECRUN

        LDA  #3         ; C'est un GOSUB qu'il faut executer,
        JSR  $D3D6      ; Ceci est le début de la routine GOSUB, mais
        LDA  TXTPTR2+1  ; la différence est qu'on ne va pas lire le n°
        PHA             ; de ligne qui suit puisqu'on l'a remplacé par
        LDA  TXTPTR2    ; un label.
        PHA
        LDA  CURLIN+1   ; Dans la pile, remarquez qu'on place:
        PHA             ; TXTPTR d'où on part,
        LDA  CURLIN     ; CURLIN qui s'execute encore,
        PHA             ; COMMANDE pour signaler que ce qui se trouve
        LDA  COMMANDE   ; dans la pile vient d'un GOSUB par exemple.
        PHA
        JSR  CHRGOT
        JSR  GOTOBIS    ; Ca y est au lieu d'aller en GOTO où on aurait
        JMP  NEWSTT     ; lu un n° de ligne, on ne le fait pas:
*                       ; on se branche aprés (GOTOBIS)
EXECRUN EQU  *
        JSR  CHRGOT     ; Est-on déja en fin de ligne ?
        BEQ  EXECRUN1   ; Si oui, alors fait le RUN...
EXECRUN2 JSR  CHRGET1   ; Sinon, il faut se placer en fin de ligne.
        BNE  EXECRUN2
EXECRUN1 EQU  *
        JSR  CLEARC     ; Mise à zero de la pile et des variables
        JMP  NEWSTT     ; Execution de la prochaine instruction...

EXECGOTO EQU  *
        JMP  DATA       ; C'est un GOTO, et comme on a déja fixé TXTPTR
*                       ; au but du saut, on execute...

* CHRGET qui controle la fin du texte.

CHRGET0 EQU  *          ; Avant de lire un caractère du prog., on doit
        LDA  TXTPTR+1   ; s'assurer qu'on est toujours dedans...
        CMP  LOMEM+1
        BCC  GOGET
        LDA  TXTPTR
        CMP  LOMEM
        BCS  PRNTEND
GOGET   JMP  CHRGET     ; ... et si c'est bon, on y va.

PRNTEND LDY  #>ERROR2   ; Si on sort, c'est que le label recherché
        LDA  #<ERROR2   ; n'a pas été trouvé. Alors on s'arrête.
        JSR  STROUT
        JMP  BASIC

* Messages divers

        HEX  8D8D8D8D8D8D8D8D8D8D8D8D
        ASC  "GOGO NE PEUT S'INSTALLER"
        HEX  8D00

        HEX  8D
        ASC  "IL FAUT GOTO, GOSUB OU RUN"
        HEX  8D0700
        HEX  8D
        ASC  "IL FAUT UN GUILLEMET"
        HEX  8D0700

        HEX  8D
        ASC  "LE LABEL N'EXISTE PAS"
        HEX  8D0700

* Commandes valables

        DFB  GOTO,GOSUB,RUN

* Masque pour le marquage de la BitMap

BITMASK HEX  8040201008040201

* Pages à protéger

PROTECT HEX  03FFFFFF

FIN     EQU  *
```

][+
//e
//e+
//c
][gs

DOS 3.3
ProDOS

Récapitulation GOGO

Après avoir saisi
ce code sous
moniteur, vous
le sauvegarderez
par :
BSAVE GOGO, A$8000,L$2F7

```
8000:AD 00 BF C9 4C F0 03 4C
8008:A1 80 A9 03 20 F5 BE 90
8010:10 20 39 FB 20 58 FC A9
8018:19 A0 03 20 3A DB 4C 1E
8020:80 8D 3D 80 8D F4 82 8D
8028:97 80 AA E8 8E F5 82 E8
8030:8E F6 82 A9 00 8D 3C 80
8038:AD C1 80 8D FF FF EE 39
8040:80 D0 03 EE 3A 80 AD 3A
8048:80 C9 82 90 0B D0 13 AD
8050:39 80 C9 F7 90 02 D0 0A
8058:EE 3C 80 D0 DB EE 3D 80
8060:D0 D6 A9 03 85 CF A4 CF
8068:B9 F3 82 48 4A 4A 4A AA
8070:BD 58 BF 85 D7 68 29 07
8078:A8 B9 EB 82 24 D7 D0 91
8080:05 D7 9D 58 BF C6 CF 10
8088:DD A0 90 B9 5A 82 99 00
8090:03 88 C0 FF D0 F5 A9 FF
8098:A0 00 8D F7 03 8C F6 03
80A0:60 A9 80 A0 C1 8D F7 03
80A8:8C F6 03 A9 80 A0 00 85
80B0:74 84 73 A0 90 B9 5A 82
80B8:99 00 03 88 C0 FF D0 F5
80C0:60 A2 01 20 B7 00 8D 80
80C8:02 20 B1 00 F0 06 9D 80
80D0:02 E8 D0 F5 CA 8E 9C 03
80D8:A2 02 BD 8E 03 CD 80 02
80E0:D0 03 4C 7C 81 CA 10 F2
80E8:A2 00 BD 80 02 C9 AB F0
80F0:18 C9 B0 F0 14 C9 AC F0
80F8:10 E8 EC 9C 03 D0 EB A0
8100:03 A9 3F 20 3A DB 4C D0
8108:03 48 8A 48 A0 00 B9 80
8110:02 85 81 E0 01 D0 06 A9
8118:00 85 82 F0 06 C8 B9 80
8120:02 85 82 A5 69 A6 6A 85
8128:83 86 84 A0 00 B1 83 C5
8130:81 38 D0 0F E6 83 18 D0
8138:02 E6 84 08 B1 83 C5 82
8140:F0 20 28 A5 83 69 06 85
8148:83 90 02 E6 84 A5 84 C5
8150:6C 90 D8 D0 1D A5 83 C5
8158:6B 90 D0 A9 00 8D A1 03
8160:F0 01 68 A4 84 A6 83 E8
8168:8A D0 01 C8 20 F9 EA 20
8170:52 E7 A5 50 8D A1 03 68
8178:AA 68 D0 05 A2 00 8E A1
8180:03 8D 9F 03 E8 BD 80 02
8188:C9 22 F0 0A A0 03 A9 5D
8190:20 3A DB 4C D0 03 AC A1
8198:03 C0 02 90 1B BD 80 02
81A0:C9 2C F0 07 E8 EC 9C 03
81A8:D0 F3 60 E8 BD 80 02 C9
81B0:22 D0 D9 88 C0 01 D0 E5
81B8:E8 8E A0 03 A5 B8 A6 B9
81C0:8D 9D 03 8E 9E 03 A4 67
81C8:88 A6 68 84 B8 86 B9 20
81D0:00 03 20 00 03 20 00 03
81D8:85 50 20 00 03 85 51 20
81E0:00 03 C9 B2 F0 18 20 00
81E8:03 D0 FB C9 00 F0 E0 D0
81F0:EE BC 80 02 C0 2C F0 24
81F8:C0 22 F0 20 D0 E8 AE A0
8200:03 20 00 03 F0 EB BC 80
8208:02 C0 2C F0 0F C0 22 F0
8210:0B DD 80 02 D0 D0 E8 EC
8218:9C 03 90 E5 AD 9F 03 C9
8220:AB F0 34 C9 AC F0 20 A9
8228:03 20 D6 D3 AD 9E 03 48
8230:AD 9D 03 48 A5 76 48 A5
8238:75 48 AD 9F 03 48 20 B7
8240:00 20 55 D9 4C D2 D7 20
8248:B7 00 F0 05 20 00 03 D0
8250:FB 20 6C D6 4C D2 D7 4C
8258:95 D9 A5 B9 C5 6A 90 06
8260:A5 B8 C5 69 B0 03 4C B1
8268:00 A0 03 A9 75 20 3A DB
8270:4C D0 03 8D 8D 8D 8D 8D
8278:8D 8D 8D 8D 8D 8D 8D C7
8280:CF C7 CF A0 CE C5 A0 D0
8288:C5 D5 D4 A0 D3 A7 C9 CE
8290:D3 D4 C1 CC CC C5 D2 8D
8298:00 8D C9 CC A0 C6 C1 D5
82A0:D4 A0 C7 CF D4 CF AC A0
82A8:C7 CF D3 D5 C2 A0 CF D5
82B0:A0 D2 D5 CE 8D 07 00 8D
82B8:C9 CC A0 C6 C1 D5 D4 A0
82C0:D5 CE A0 C7 D5 C9 CC CC
82C8:C5 CD C5 D4 8D 07 00 8D
82D0:CC C5 A0 CC C1 C2 C5 CC
82D8:A0 CE A7 C5 D8 C9 D3 D4
82E0:C5 A0 D0 C1 D3 8D 07 00
82E8:AB B0 AC 80 40 20 10 08
82F0:04 02 01 03 FF FF FF
```

Disquettes Pom's 31 Apple //

Voici la liste des fichiers que vous trouverez sur les disquettes Pom's Apple // correspondant à ce numéro.

La disquette 5'25 pouces 140Ko est formatée au recto en DOS 3.3 et au verso en ProDOS. Tous les fichiers ne tenant pas sur une face, nous les avons répartis en fonction du DOS utilisé. La face ProDOS ne *boote* pas (pour une question de copyright) ; il vous appartient donc d'y installer ProDOS et Basic.System.

La disquette 800Ko comprend tous les fichiers en format ProDOS. Il faudra convertir en DOS 3.3 les fichiers qui le nécessitent.

Liste des fichiers, face ProDOS

GOGO	BIN	Objet à exécuter
T.GOGO	TXT	Source Assembleur
GOGO.DEMO	BAS	Programme de démonstration
PIC.POC	BIN	Objet à exécuter
GSCOLOR	BAS	Programme de démonstration
PIC.POC.S	TXT	Source Assembleur
LIT.MESS.GS	BAS	Programme de démonstration
T.WRITE560	TXT	Source Assembleur
T.WRITE560.DEMO	TXT	Source Assembleur de démo
WRITE560.DEMO	BIN	Objet correspondant
ESSAI.PAS	BIN	Source Turbo-Pascal
DIAG.PAS	BIN	Source Turbo-Pascal
LEMOUSE3.0.C	BIN	Objet à exécuter
LEMOUSE3.0	TXT	Source Assembleur
START.MOUSE	BAS	Programme de lancement
STARTUP	BAS	Programme d'accueil
IM31	BIN	5 images générées
PAPILLON	BIN	avec le
NOEL	BIN	programme
ELLIPSES	BIN	Schémateur
GLOBE	BIN	de la face DOS 3.3

Liste des fichiers face DOS 3.3

A 002	*HELLO*	Programme d'accueil
A 008	*MENU*	Programme de lancement de
A 041	SCHEMATEUR	SCHEMATEUR
A 003	GRAPHISME	
B 002	HEL.C	
B 011	SCHEM.C	
B 008	CARAC.SH	
A 036	FONCTION	
B 002	FONC.C	
A 018	GRAVURE	
B 004	GRAV.C	
B 002	ERREUR	
B 004	CHARGE.OBJ	
T 003	T.HEL	4 sources non listés dans
T 062	T.SCHEM	la revue
T 006	T.FONC	
T 013	T.GRAV	
B 004	GOGO	Objet à exécuter
T 061	T.GOGO	Source Assembleur
A 003	*GOGO.DEMO*	Programme de démonstration
A 037	*LE.VOYAGEUR*	Programme à exécuter
T 002	TEXT	Fichier de démonstration
B 013	ESSAI.PAS	Source Turbo-Pascal
B 019	DIAG.PAS	Source Turbo-Pascal
B 034	IM31	Image HGR

Une nouvelle carte : Speedisk

Alexandre Duback

La carte dont nous vous proposons ici un banc d'essai est distribuée par la S.A. LOGMA, société qui travaille par ailleurs en régie pour la revue Pom's.

La carte

La carte Speedisk est un disque virtuel permanent : elle est un véritable volume ProDOS, sur lequel on peut démarrer et sur lequel on stocke fichiers et programmes. La mémoire de masse n'est plus un support magnétique traditionnel : la carte est constituée de circuits CMOS sans organes mécaniques en mouvement : les accès disque sont donc extrêmement rapides et la fiabilité est comparable à celle de l'unité centrale.

On accède à l'information en 0,2 ms soit 100 fois plus vite qu'avec un disque dur. Sur un Apple IIGS en mode rapide les chargements s'effectuent à raison de 100 blocs par seconde soit 51200 octets/seconde.

Les informations sont préservées pendant deux mois après l'arrêt de l'ordinateur, grâce à un accumulateur qui se recharge lorsque l'ordinateur est sous tension : il ne faut donc pas laisser l'appareil plus de deux mois hors tension sans sauvegarder le contenu de Speedisk sur disquette.

La capacité

Speedisk est proposée en quatre versions : 384 Ko et 1 Mo, avec ou sans horloge. La première version est extensible à 1 Mo.

Les cartes avec horloge compatible ProDOS n'ont d'intérêt que sur][+ et //e, la carte intégrée au IIGS remplissant fort bien son office. Notons d'ailleurs que les cartes Speedisk dotées d'une horloge sont incompatibles avec le IIGS.

ProDOS se refuse à reconnaître un support de masse et une carte horloge dans le même slot, aussi, pour utiliser l'extension horloge, il faut installer un programme (livré sur disquette) qui s'exécute lors du démarrage pour modifier ProDOS.

L'ordinateur

La carte fonctionne indifféremment sur Apple][+ (avec carte langage pour ProDOS), //e et IIGS. Les programmes sont métamorphosés surtout ceux ayant recours fréquemment à des fichiers : les programmes WPL d'AppleWriter par exemple, sont méconnaissables.

La carte étant un support disque comme un autre, est toujours adressée à 1 Mhz même en mode rapide (sur le IIGS bien sûr).

La vitesse

Mieux qu'un long discours, voici les temps d'exécution mesurés.

- Booter sur Basic.System ≈ **3 s**
- BRUN FILER ≈ **1/2 s**
- BRUN AppleWriter ≈ **1/2 s**
- Sauver un texte de 46 Ko depuis AppleWriter ≈ **5 s**
- AppleWorks ≈ **1 s**
- Écrire un enregistrement TEXT de 100 octets ≈ **1/36 s**
  soit 3,5 Ko/s
  soit 2150 enregistrements/mn
- BSAVE de 36 Ko ≈ **1 s**
- Formater 1 Mo ≈ **1 s**
- Avec MouseDesk, copier 75 fichiers (900 Ko) de Speedisk vers /RAM5 ≈ **1 mn**
- De Launcher à GSPaint ≈ **17 s**

Temps cumulés :

	sur Unidisk 3,5'	*sur Speedisk*
PR£slot	0 s	***0s***
ProDOS 16	9 s	***2s***
Launcher	49 s	***10s***
MouseDesk	71 s	***15s***
GSWrite	149 s	***34 s***

À l'usage

On se prend rapidement à oublier le lecteur traditionnel et à imaginer un 'disque dur' de 20 Mo conçu de la même façon. Mais, en réalité, le nombre de programmes utilisés de façon journalière est restreint et tient sans nul doute sur 1 Mo. Par exemple : un programme de gestion du compte bancaire, un traitement de texte, un tableur, un programme de dessin, quelques programmes en Basic et les utilitaires essentiels de Pom's... Pour les collections de programmes rarement utilisés, la disquette reste le stockage économique.

Sur 384Ko, on se sent plus à l'étroit et seuls les programmes utilisés quotidiennement trouvent leur place.

Le spécialiste du traitement de texte n'hésite plus à faire les sauvegardes, quasi-instantanées, sans craindre que le disque virtuel /RAM ou /RAM5 perde la mémoire à la suite d'une micro-coupure.

Pour une utilisation plus spéciale (centres de formation, école) et tous les milieux à risques, le disque virtuel permanent est certainement la solution la moins sensible aux

indélicats.

L'aspect 'environnement de bureau' type Macintosh devient réellement utilisable car revenir à MouseDesk prend quelques secondes seulement alors que, sur disquette, l'utilisateur recourt parfois au RESET... L'Apple IIGS passe plus de temps à son autotest de mise sous tension qu'à booter Basic.System.

Un petit regret tout de même : il n'existe pas encore de contrôleur –ou plutôt de *patch*– pour l'utiliser sous DOS 3.3, mais, patience. Idem pour Pascal et CP/M.

Au SAMU

Cette carte a en effet été adoptée par un SAMU, activité dans laquelle fiabilité et accès rapide à l'information sont des critères vitaux.

Dans la carte

Strictement équivalente à un lecteur de disquette, on trouve dans la page $Cn (n est le n° de slot) les conventions habituelles : $Cn01, 03 et 05 valent bien $20, 0 et 3.

Pour la carte 1 Mo, en $CnFC et $CnFD il y a bien 00 08 qui signifient $800 blocs sur le disque : 2048 x 512 = 1024 Ko.

Surprise : en $CnFE on trouve $8F, le bit 7 est à 1 ce qui signifie 'support amovible'. Surprenant pour une telle carte !... Il s'agit en fait d'un défaut de ProDOS qui ne supporte pas ce bit à 0. Aucune conséquence pratique, pas plus que pour les disques durs dont les contrôleurs ont également ce bit à 1.

Plus standard, on dispose de 51 entrées dans le catalogue du volume. Au-delà il faut créer des sous-catalogues comme pour les autres volumes.

Pour utiliser l'horloge des cartes SP 400H et SP1000H, il est nécessaire d'installer le programme `HGSP.SYSTEM` comme premier fichier système de la carte. Ce programme installe le vecteur ProDOS en $BF06.$BF08 : `4C D8 Cn` c'est-à-dire `JSR $CnD8` puis effectue un premier `GET_TIME` pour mettre à jour la date dans la page globale en $BF90.$BF93. Ce programme lance alors le fichier système suivant dans le catalogue.

L'ensemble des `GET_TIME` aboutiront alors à une mise à l'heure de la page globale : c'est le cas par exemple d'un `SAVE` ou d'un `CLOSE` en Basic (les fichiers sont ainsi datés).

Le tarif

SP400 (384 Ko)	3 990,00 F
SP1000 (1 Mo)	5 990,00 F
SP400 horloge	4 580,00 F
SP1000 horloge	6 580,00 F

tarifs sur lesquels LOGMA applique une remise de 10% aux lecteurs de Pom's.

Les cartes sont garanties 1 an, et ce par échange en cas de problème.

Une carte vocale : Valentine

La carte de synthèse de parole Valentine n'est pas à proprement parler une nouveauté : nous en avions eu deux dans les mains il y a un an mais elles n'en étaient encore qu'à la phase de développement, le mode d'emploi ressemblait encore trop à un cahier des charges. Pour la petite histoire, on lisait à l'époque, ça et là des 'bancs d'essai' fort élogieux qui ne manquaient pas d'abuser le lecteur. Par ses possibilités, la nouvelle version de Valentine mérite cette présentation.

La carte

Elle a été conçue en collaboration avec l'Association Valentin Haüy pour les Aveugles et c'est bien entendu à ceux-ci que s'adresse en priorité la synthèse de parole.
La carte s'enfiche dans l'un des slots de l'Apple //e, ordinateur qui doit être muni du kit 65C02 ; le haut parleur interne de l'Apple est utilisé mais un jack est prévu pour un modèle extérieur. Outre la connexion au slot, un fil est à relier – à l'aide d'un support de circuit prévu à cet effet – au décodeur clavier. Le clavier est alors reconnu par la carte qui peut générer une interruption et donc travailler en immobilisant provisoirement le programme en cours. La version IIGS (moins coûteuse) qui n'est pas encore disponible, ne nécessitera pas

cette intervention puisque le clavier, comme sur le Macintosh, génère des interruptions à la disposition du programmeur.

De la dimension d'une carte SSC, les modèles que nous avons eu en mains souffraient encore de la présence de quelques straps, composants ajoutés côté pistes et modifications de dernière heure.

Autour sur processeur 8085 à 8 Mhz, nous trouvons 196Ko de ROM sur la carte : 77 pour les phonèmes, 50 pour les règles de prononciation et le reste pour le programme.

La synthèse

Avec un peu d'habitude, l'intégralité de la prononciation est reconnue ; après avoir tenté une voix d'homme, c'est finalement une voix féminine qui a été retenue, vitesse de prononciation et timbre sont réglable. Les occlusives *p t d* et *b* prononcées seules restent difficiles à différencier et demandent un effort d'attention, idem pour les fricatives *f* et *v* ou pour '.' et '-' dits «point» et «moins». Un mode 'télégraphique' (ou plutôt aéronautique) lève les ambiguïtés les mots étant épelés «Papa Tango Delta Bravo...». Les signes et accents sont dits *en clair* : «a commercial, e accent aigu, parenthèse ouverte...».

La carte sait prononcer sans erreur les pièges-type tels «les **présidents président**, les poules du **couvent couvent**, estomac». En revanche les mots-clé du Basic ne faisant pas partie des exceptions sont prononcés à la française. Les nombres sont fort bien dits «cent soixante milliards quatre cent onze millions...».

Les minuscules sont différenciées des majuscules et l'utilisateur choisit, selon le type de travail le mode qui convient : *A a* seront prononcés «A A MIN» ou «A MAJ A» ; le premier mode convient au Basic, riche en majuscules, le second au traitement de textes.

Quatre modes de fonctionnement

1–Echo-clavier

Le mode le moins utilisé car la disposition des touches d'un clavier est le plus simple à acquérir pour un malvoyant. Dans ce mode, après avoir fait *PR£slot* depuis le Basic (les problèmes commencent quand on n'est pas en Basic...), la carte prononce les touches appuyées : «Pé èR I èN Té» par exemple. Les affichages à l'écran ne génère pas de synthèse.
La synthèse se fait au rythme de un caractère/seconde ; un caractère survenant avant le délai d'une seconde interrompt la synthèse du précédent.

2–Echo-écran

Ce qui est affiché à l'écran est prononcé. Bien sûr, seuls les accès standard à l'écran sont interceptés : *PRINT* depuis le Basic, *JSR $FDED* en assembleur par exemple. Les affichages par gestion directe de la mémoire écran du style *STA $400,Y* sont invisibles pour la carte. Ce mode n'est d'ailleurs pas le plus utile ni le plus confortable.

3–Éditeur vocal

Le point fort de Valentine II : une interruption par *Ó-RETURN* et la carte prend le pas sur le programme. Il est alors possible de demander la lecture de l'écran (quelle que soit la méthode retenue pour l'affichage) aussi bien en 40 qu'en 80 colonnes. L'éditeur est puissant, il comprend les commandes suivantes :

- déplacement du 'curseur vocal' à l'aide des flèches, un *bip* étant généré à chaque tentative de dépassement des limites ;
- positionnement en début et fin de ligne (les espaces en fin de ligne ne comptent pas) ;
- positionnement en haut et en bas de l'écran ;
- positionnement au début du mot suivant ;
- positionnement au début du mot précédent ;
- positionnement absolu : V02 H17 pour aller au 17ème caractère de la 2ème ligne ;
- lecture d'un caractère ;
- lecture de l'écran à partir du curseur ;
- lecture d'une ligne ;
- lecture d'un mot ;
- épellation d'un mot.

La carte ne précise pas si un caractère est en mode flash mais indique le mode inverse : «b inv», «e inv» par exemple.

Perdu au milieu de l'écran, il suffit de presser *G* pour que la carte donne la position du curseur vocal sous la forme de deux chiffres (ligne, colonne). Nous avons constaté une erreur épisodique dans les numéros de ligne, l'écran étant parfois numéroté à partir de 2.

On peut toujours stopper la synthèse, ouf ! certains écrans sont en effet bien remplis.

Trois modes de prononciation

Le mode littéraire est celui retenu par défaut : les *()§°*-&£:/=><?* et *!* ne sont pas signalés.
Mode détaillé, tout est prononcé sauf *?!;,* et .
En mode épellation, rien n'est oublié.

4–Parole compressée

Dans ce mode, d'une utilisation plus spécialisée, la carte prononce des paroles préalablement compressées et stockées en fichiers. Le principe ressemble à celui du programme *Le Bavard* paru dans Pom's, qualité acoustique incomparable, celle de la carte étant alors parfaite. Ce mode ouvre la porte aux applications du type ordinateur à bord des voitures ou répondeur téléphonique. Le fabricant s'occupe de compresser les textes pour utilisation avec la carte.

Mode d'emploi et commandes

Le mode d'emploi (en traitement

de texte) n'est pas très précis sur les syntaxes : faut-il un espace ?, faut-il un point ? Il est disponible également en braille et sur cassette. Les commandes choisies pour l'éditeur vocal sont suffisamment mnémotechniques pour ne pas recourir à la documentation. Il est certain qu'un malvoyant ne connaissant pas l'Apple ne pourra se dispenser des cours de formation de l'Association Valentin Haüy pour utiliser AppleWorks avec la carte.

Dans la pratique

AppleWorks

Comme pour tous les programmes, le problème est d'activer Valentine (NDLR : !) par l'équivalent d'un *PR£*. Pour AppleWorks, pas de problèmes, on définit la carte comme étant l'imprimante et le premier *Ó-H* déclenche la copie d'écran salvatrice. L'éditeur vocal remplit alors fort bien son office ; avec un peu d'habitude on doit identifier rapidement un menu.

Dans le tableur, il est difficile de ne pas se perdre au milieu de la feuille de calcul, un peu de virtuosité n'est pas superflue.

Avoir retrouvé une faute à l'aide de Valentine est une chose, mais positionner le curseur du traitement de textes pour effectuer la modification... Il faut repérer la chaîne de caractères qui suit la position à rechercher puis demander à la fonction de recherche la chaîne voulue. Le curseur est alors en principe à l'endroit qui convient. Pas simple.

La position courante d'écriture est à jour avec bien des programmes ($24, $25) : Valentine aurait pu émettre un bip particulier lors du passage du curseur vocal sur cette position, c'était une aide possible.

ProCODE

Pas de problème de fonctionnement, mais comment activer la carte ? Il suffit de quitter par *CTRL-Q*, faire 2 *CTRL-P* si Valentine est en slot 2, puis *D92G* pour relancer ProCODE. *RESET* le relance également mais déconnecte la carte.

Interpom's

Pas de souci, faire simplement *PR£2* avant *-INTERPOMS*.

Assembleur BigMac

Malgré une petite perturbation de l'affichage, BigMac est compatible avec Valentine.

AppleWriter

Nous n'avons pas trouvé de possibilité d'utilisation. Contrairement aux indications de la documentation, Valentine est gérée 'à la manière' de la carte Super Série Apple et non 'comme'.

Dommage, si les registres *DATA* et *STATUS* était les mêmes (*$C089*, *$C088*), il serait possible au moins d'"imprimer" le texte sur la carte car AppleWriter reconnaît la SSC.

Programmes Pascal

Nous n'avons aucune possibilité d'utiliser Valentine avec le langage Pascal. Mêmes causes, mêmes effets, on ne peut pas *T)ransférer* un fichier sur la carte.

Sargon II

Les échecs Sargon ne seront pas à la portée des malvoyants : le plantage survient rapidement.

Assembleur

Sortir des caractères sur la carte en assembleur ne pose pas plus de problème qu'en Basic ; à noter que la routine indiquée dans le mode d'emploi était incompatible avec la carte dont nous avons disposée (caractères prononcés deux fois).

CP/M

Avant d'utiliser la carte, un *patch* du CP/M (indiqué dans la notice) est nécessaire ; ce patch devrait être livré sur une disquette pour éviter à l'utilisateur final de s'interroger sur l'assembleur Z80. Le fonctionnement ne sera pas total car CP/M intercepte le caractère *CTRL-P* avant la carte mais l'éditeur vocal reste opérationnel.

Conclusion

Certainement une excellente carte qui tranche totalement avec celle proposée il y a un an et qui met la micro-informatique Apple à la portée des aveugles. Il reste que l'Association Valentin Haüy sera peut-être la bonne source de conseils en fonction de l'application envisagée.

Le prix de la carte –environ 5 500,00 F TTC–, intervient dans la décision d'achat mais il n'est pas surestimé en regard des études investies.

Une carte a été conçue par un utilisateur de Valentine pour la doter de *macros*. Cette carte, peu coûteuse, s'insère dans un autre slot et permet par exemple à une seule touche de commander la mise en éditeur vocal, le positionnement en bas de l'écran, puis en fin de ligne, puis un mot en arrière, puis la lecture du mot. Ainsi, en cas de doute sur la frappe, le dernier mot saisi sera facilement prononcé par Valentine. Elle donne également une fonction de recherche de chaînes.

Lorsque les yeux font défaut, on peut penser à l'écran tactile qui remplace le moniteur : très rapide d'accès pour l'utilisateur, il donne une image fidèle de l'écran à l'aide de huit picots par caractère (soit 256 combinaisons possibles) disposés à la manière des points braille, le curseur étant symbolisé par les huit points *ON*. Il faut hélas y consacrer le prix de huit Valentines II...

Sté Ferma, 14 rue de l'abbé Carton, 75014 Paris.
Tél. : (1) 45 43 14 79

Ecriture proportionnelle en DHGR

F. Rozay

WRITE560 permet d'écrire en mode graphique sur un Apple // (//e, //c, IIGS) muni d'au moins 128Ko ; il ferait double emploi si le mode graphique utilisé était l'habituelle résolution de 280x192 points, mais le mode très haute résolution 560 x 192 a été adopté. L'écriture s'effectuera avec une chasse proportionnelle : le **m** occupera plus de place qu'un **i**.

N'aviez vous jamais rêvé de faire du *wysiwig* (What You See Is What You Get : ce que vous voyez est ce que vous obtenez) sur un autre ordinateur que le Mac ?

Utilisation

L'appel de ce programme est destiné prioritairement à des programmes en assembleur. Il vous sera toujours possible d'écrire une interface avec l'Applesoft, via quelques POKEs ou l'ampersand par exemple.

Le protocole concerne 4 points :

- Le programme doit être appelé par `JSR` ou `CALL` à l'adresse `CHROUT`.
- La partie haute de l'adresse de la chaîne à éditer doit être placée en `CHROUT1+2`, alors que la partie basse le sera en `CHROUT1+1`. Les caractères composant la chaîne doivent avoir le bit 7 à 1, et se terminer par un $8D.
- On peut préciser la position initiale de la chaîne sur l'écran en plaçant ses coordonnées dans l'accumulateur (position verticale de 0 à 191), et dans les registres d'index (position horizontale de 0 à 559). Enfin, il faut mettre la retenue à 1 avant d'appeler le programme.

Par exemple, pour se positionner sur la 100ème ligne et la 320ème colonne, il suffit de faire :

```
LDA £100
LDX £64
LDY £1   ; 1*256+64=320
SEC
```

- L'espace entre les lettres peut être modifié à loisir ; par défaut, deux points blancs sont intercalés entre chaque lettre. Pour modifier cette valeur, il faut la placer à l'adresse ESPACE+1.

Enfin, le programme `WRITE560` est utilisable indifféremment sous ProDOS ou DOS 3.3. Par ailleurs, son adaptation à Pascal ne devrait pas poser de problèmes.

Fonctionnement

Le codage des lettres consiste à les inverser puis les dessiner dans une matrice de 7x8 points et enfin de recoder le tout.

Par exemple pour un ***h*** :

```
 7654321
8......• 01
7......• 01
6......• 01      largeur 4
5....••• 07
4...•..• 09
3...•..• 09
2...•..• 09
1....... 00
```

pour un ***w*** :

```
 7654321
8....... 00
7....... 00
6..•...• 11
5..•...• 11      largeur 5
4..•...• 11
3..•.•.• 15
2..••.•• 1B
1....... 00
```

Le programme `WRITE560.DEMO` est un programme de démonstration qu'il suffit de lancer par :

```
PR£ 3
BRUN WRITE560.DEMO
```

suivi d'une instruction `TEXT` après son exécution.

Le source `T.WRITE560` est stocké sous forme d'un fichier TEXT au format Merlin/BigMac sur la disquette d'accompagnement de ce numéro. Il est appelé par le pseudo-opcode `PUT` de cet assembleur, aussi appelé `COPY` sur d'autres assembleurs. Par ailleurs, le source `T.WRITE560.DEMO` illustre les modalités d'appel.

Source T.WRITE560

Assembleur BigMac

```
HBAS     =    $26
PCL      =    $FA

TXTPAGE1 =    $C054
TXTPAGE2 =    $C055


CHROUT
         BCC  CHROUT1
```

```
          STA  ABS
          STX  ORDL
          STY  ORDH
CHROUT1   LDA  $FFFF
          BEQ  CHROUTFIN
          CMP  £$8D
          BEQ  RETURN
          CMP  £$A0
          BCC  AVANCE
          CMP  £$C0
          BCC  PONCTUA
          CMP  £$E0
          BCC  GRANDE
          JMP  PETITE

CHROUTFIN RTS
RETURN
          LDA  ABS
          CLC
          ADC  £10
          STA  ABS
          LDA  £0
          STA  ORDL
          STA  ORDH

AVANCE
          INC  CHROUT1+1
          BNE  AVANCE1
          INC  CHROUT1+2
AVANCE1   JMP  CHROUT1

PONCTUA
          SEC
          SBC  £$A0
          PHA
          LDA  £<PONCT
          STA  PCL
          LDA  £>PONCT
          STA  PCL+1
          PLA
          TAY
          LDA  LGPONC,Y
          STA  EMPAT
          JMP  CHROUT6

GRANDE
          SEC
          SBC  £$C0
          PHA
          LDA  £<MAJUS
          STA  PCL
          LDA  £>MAJUS
          STA  PCL+1
          PLA
          TAY
          LDA  LGMAJUS,Y
          STA  EMPAT
          JMP  CHROUT6

PETITE
          SEC
          SBC  £$E0
          PHA
          LDA  £<MINUS
          STA  PCL
          LDA  £>MINUS
          STA  PCL+1
          PLA
          TAY
          LDA  LGMINUS,Y
          STA  EMPAT

CHROUT6
          TYA
          ASL
          ASL
          ASL
          CLC
          ADC  PCL
          STA  PCL
          BCC  CHROUT2
          INC  PCL+1
CHROUT2
          LDA  £8
          STA  LENGTH
          JSR  DRAW
          LDA  EMPAT
          CLC
ESPACE    ADC  £2
          ADC  ORDL
          STA  ORDL
          BCC  CHROUT3
          INC  ORDH
CHROUT3   LDA  ORDH
          CMP  £2
          BNE  CHROUT8
          LDA  ORDL
          CMP  £$2A
          BCC  CHROUT8
          JMP  RETURN
CHROUT8   JMP  AVANCE

DRAW
          STA  TXTPAGE1
          LDA  ABS
          TAX
          LDA  ORDL
          STA  DGTL
          LDA  ORDH
          STA  DGTH
          JSR  DIV7
          LDY  £0
DRAW1     JSR  GETX
          STY  ECCO
          TXA
          PHA
          LDA  £0
          STA  COMPLEMENT
          LDA  (PCL),Y
          LDX  RESTE
          BEQ  DRAW5

DECALE    DEX
          ASL
          PHA
          ASL
          ROL  COMPLEMENT
          PLA
          CPX  £0
          BNE  DECALE

DRAW5
          STA  DRAND1+1
          LDA  QUOTIENT
          LSR
          TAY
          BCS  DRAW2
          STA  TXTPAGE2
DRAW2
          LDA  (HBAS),Y

DRAND1    EOR  £$FF
          STA  (HBAS),Y
          STA  TXTPAGE1
          LDX  QUOTIENT
          INX
          TXA
          LSR
          TAY
          BCS  DRAW4
          STA  TXTPAGE2
DRAW4
          LDA  (HBAS),Y
          EOR  COMPLEMENT
          STA  (HBAS),Y
          STA  TXTPAGE1
          PLA
          TAX
          INX
          CPX  £192
          BEQ  DRAW3
          LDY  ECCO
          INY
          CPY  LENGTH
          BNE  DRAW1
DRAW3     RTS

GETX
          LDA  VERTH,X
          STA  HBAS+1
          LDA  VERTL,X
          STA  HBAS
          RTS

DIV7      CLD
          LDA  £0
          STA  M0
          STA  M1
          STA  QUOTIENT
          LDA  DGTH
          BNE  DIV
          LDA  DGTL
          CMP  £7
          BCS  DIV
          STA  RESTE
          RTS

DIV       INC  QUOTIENT
          LDA  M0
          CLC
          ADC  £7
          STA  M0
          BCC  OVERFW
          INC  M1
OVERFW    LDA  M1
          CMP  DGTH
          BNE  DIV
          LDA  DGTL
          CMP  M0
          BCC  DIVIN
          SEC
          SBC  M0
          CMP  £7
          BCS  DIV
          STA  RESTE
          RTS
DIVIN     LDA  M0
          SEC
          SBC  DGTL
```

```
        STA  DIVID+1
        LDA  £7
        SEC
DIVID   SBC  £$FF
        CMP  £7
        BCS  DIV
        DEC  QUOTIENT
        STA  RESTE
        RTS

PONCT   HEX  0000000000000000 ;ESP
        HEX  0101010101000100 ;!
        HEX  0505050000000000 ;"
        HEX  0C02020702021D00 ;£
        HEX  041E050E140F0400 ;$
        HEX  0313080402191800 ;%
        HEX  0205050215091600 ;&
        HEX  0606020100000000 ;'
        HEX  0402010101020400 ;(
        HEX  0102040404020100 ;)
        HEX  04150E040E150400 ;*
        HEX  0004041F04040000 ;+
        HEX  0000000606020100 ;,
        HEX  0000000700000000 ;-
        HEX  0000000000030300 ;.
        HEX  0010080402010000 ;/
        HEX  0E11191513110E00 ;0
        HEX  0203020202020700 ;1
        HEX  0E11100C02011F00 ;2
        HEX  1F10080C10110E00 ;3
        HEX  080C0A091F080800 ;4
        HEX  1F010F1010110E00 ;5
        HEX  1C02010F11110E00 ;6
        HEX  1F10080402020200 ;7
        HEX  0E11110E11110E00 ;8
        HEX  0E11111E10080700 ;9
        HEX  0003030003030000 ;:
        HEX  0000020002020100 ;;
        HEX  0804020102040800 ;<
        HEX  0000070007000000 ;=
        HEX  0102040804020100 ;>
        HEX  1E3330180C000C00 ;?
MAJUS
        HEX  08100E101E111E00 ;à
        HEX  181412111F111100 ;A
        HEX  0F11110F11110F00 ;B
        HEX  0E11010101110E00 ;C
        HEX  0F11111111110F00 ;D
        HEX  1F01010F01011F00 ;E
        HEX  1F01010F01010100 ;F
        HEX  0E11010119111E00 ;G
        HEX  1111111F11111100 ;H
        HEX  0702020202020700 ;I
        HEX  1C08080808090600 ;J
        HEX  1109050305091100 ;K
        HEX  0101010101010F00 ;L
        HEX  4163554941414100 ;M
        HEX  4143454951614100 ;N
        HEX  0E11111111110E00 ;O
        HEX  0F11110F01010100 ;P
        HEX  0E11111115091600 ;Q
        HEX  0F11110F05091100 ;R
        HEX  0E11010E10111C00 ;S
        HEX  1F04040404040400 ;T
        HEX  1111111111110E00 ;U
        HEX  11111111110A0400 ;V
        HEX  4141414955634100 ;W
        HEX  11110A040A111100 ;X
        HEX  1111110A04040400 ;Y
        HEX  1F10080402011F00 ;Z
        HEX  0205020000000000 ;°
        HEX  00000E01010E0402 ;ç
        HEX  0E01060906080700 ;$
        HEX  040A110000000000 ;^
        HEX  000000000000000F ;_
MINUS
        HEX  0102040000000000 ;`
        HEX  000006080E090E00 ;a
        HEX  0101010709090700 ;b
        HEX  00000E0101010E00 ;c
        HEX  0808080E09090E00 ;d
        HEX  0000060907010E00 ;e
        HEX  040A020207020200 ;f
        HEX  00000E09090E0806 ;g
        HEX  0101010709090900 ;h
        HEX  0200030202020700 ;i
        HEX  08000C0808080906 ;j
        HEX  0101090503050900 ;k
        HEX  0302020202020700 ;l
        HEX  00001B1515111100 ;m
        HEX  0000070909090900 ;n
        HEX  0000060909090600 ;o
        HEX  0000070909070101 ;p
        HEX  00000E09090E0808 ;q
        HEX  00000D0301010100 ;r
        HEX  00000E0106080700 ;s
        HEX  02020F0202020C00 ;t
        HEX  0000090909D0B00 ;u
        HEX  00001111110A0400 ;v
        HEX  0000111115151B00 ;w
        HEX  0000110A040A1100 ;x
        HEX  000009090E080807 ;y
        HEX  00001F0804021F00 ;z
        HEX  080406090F010E00 ;é
        HEX  0204001111191600 ;ù
        HEX  040806090F010E00 ;è
        HEX  0500000000000000 ;¨
        HEX  0F0F0F0F0F0F0F0F ;

* Tables pour le calcul de HBAS

VERTL   HEX  0000000000000000
        HEX  8080808080808080
        HEX  0000000000000000
        HEX  8080808080808080
        HEX  0000000000000000
        HEX  8080808080808080
        HEX  0000000000000000
        HEX  8080808080808080
        HEX  2828282828282828
        HEX  A8A8A8A8A8A8A8A8
        HEX  2828282828282828
        HEX  A8A8A8A8A8A8A8A8
        HEX  2828282828282828
        HEX  A8A8A8A8A8A8A8A8
        HEX  2828282828282828
        HEX  A8A8A8A8A8A8A8A8
        HEX  5050505050505050
        HEX  D0D0D0D0D0D0D0D0
        HEX  5050505050505050
        HEX  D0D0D0D0D0D0D0D0
        HEX  5050505050505050
        HEX  D0D0D0D0D0D0D0D0
        HEX  5050505050505050
        HEX  D0D0D0D0D0D0D0D0
VERTH   HEX  2024282C3034383C
        HEX  2024282C3034383C
        HEX  2125292D3135393D
        HEX  2125292D3135393D
        HEX  22262A2E32363A3E
        HEX  22262A2E32363A3E
        HEX  23272B2F33373B3F
        HEX  23272B2F33373B3F
        HEX  2024282C3034383C
        HEX  2024282C3034383C
        HEX  2125292D3135393D
        HEX  2125292D3135393D
        HEX  22262A2E32363A3E
        HEX  22262A2E32363A3E
        HEX  23272B2F33373B3F
        HEX  23272B2F33373B3F
        HEX  2024282C3034383C
        HEX  2024282C3034383C
        HEX  2125292D3135393D
        HEX  2125292D3135393D
        HEX  22262A2E32363A3E
        HEX  22262A2E32363A3E
        HEX  23272B2F33373B3F
        HEX  23272B2F33373B3F

*  Largeur de toutes les lettres

LGPONC  DFB  5,1,3,5,5,5,5,3,3,
             3,5,5,3,3,2,5,5
        DFB  3,5,5,5,5,5,5,5,5,
             2,2,4,3,4,6,5
LGMAJUS DFB  5,5,5,5,5,5,5,5,5,
             3,5,5,4,7,7,5
        DFB  5,5,5,5,5,5,5,7,5,5,
             5,3,7,3,5,5
LGMINUS DFB  5,4,4,4,4,4,4,4,4,3,
             4,4,3,5,4,4
        DFB  4,4,4,4,4,4,4,5,5,5,
             4,5,5,5,5,5,5

M0      DS   1
M1      DS   1
QUOTIENT DS  1
RESTE   DFB  0
DGTL    DS   1
DGTH    DS   1
COMPLEMENT DS 1
EMPAT   DS   1
ORDL    DS   1
ORDH    DS   1
ABS     DS   1
LENGTH  DS   1

ECCO    DFB  0
```

//e
//e+
//c
][gs

DOS 3.3
ProDOS

Source T.WRITE560.DEMO

Assembleur Merlin/BigMac

```
         LST  OFF
         ORG  $1000

SET80COL =    $C00D
TXTCLR   =    $C050
MIXCLR   =    $C052
HIRES    =    $C057
SETAN3   =    $C05E
CLRHR1   =    $C0B2
SETHR2   =    $C0B5
SETHR3   =    $C0B7

         JMP  DEBUT

* lit le fichier principal
         PUT  WRITE560

* Passe en mode graphique 560*192

DEBUT
         STA  SET80COL
         STA  MIXCLR
         STA  HIRES
         STA  TXTCLR
         STA  SETAN3
         STA  CLRHR1
         STA  SETHR2
         STA  SETHR3
         STA  TXTPAGE1

* Efface et affiche un fond gris
*
         LDX  £0
EFF1     LDY  £0
         JSR  GETX
EFF2     TYA
         PHA
         LSR
         TAY
         BCS  EFF3
         STA  TXTPAGE2
EFF3     LDA  COULEUR
         STA  (HBAS),Y
         STA  TXTPAGE1
         PLA
         TAY
         INY
         TYA
         PHA
         LSR
         TAY
         BCS  EFF4
         STA  TXTPAGE2
EFF4     LDA  COULEUR+1
         STA  (HBAS),Y
         STA  TXTPAGE1
         PLA
         TAY
         INY
         CPY  £80
         BNE  EFF2
         INX
         CPX  £192
         BEQ  EFFIN

         LDY  £0
         JSR  GETX
EFF6     TYA
         PHA
         LSR
         TAY
         BCS  EFF7
         STA  TXTPAGE2
EFF7     LDA  COULEUR
         STA  (HBAS),Y
         STA  TXTPAGE1
         PLA
         TAY
         INY
         TYA
         PHA
         LSR
         TAY
         BCS  EFF8
         STA  TXTPAGE2
EFF8     LDA  COULEUR
         STA  (HBAS),Y
         STA  TXTPAGE1
         PLA
         TAY
         INY
         CPY  £80
         BNE  EFF6
         INX
         CPX  £192
         BNE  EFF1
EFFIN

* Envoie les messages à l'écran...
         LDA  £<MESSAGE0
         LDY  £>MESSAGE0
         STA  CHROUT1+1
         STY  CHROUT1+2
         LDA  £2
         STA  ESPACE+1
         LDA  £20
         LDX  £10
         LDY  £0
         SEC
         JSR  CHROUT

         LDA  £<MESSAGE1
         LDY  £>MESSAGE1
         STA  CHROUT1+1
         STY  CHROUT1+2
         LDA  £5
         STA  ESPACE+1
         CLC
         JSR  CHROUT

         LDA  £<MESSAGE2
```

```
        LDY   £>MESSAGE2
        STA   CHROUT1+1
        STY   CHROUT1+2
        LDA   £0
        STA   ESPACE+1
        CLC
        JSR   CHROUT
        LDA   £<MESSAGE3
        LDY   £>MESSAGE3
        STA   CHROUT1+1
        STY   CHROUT1+2
        LDA   £2
        STA   ESPACE+1
        CLC
        JSR   CHROUT
        RTS

COULEUR DFB   0,0
```

```
MESSAGE0 ASC  "Démonstration d'écriture avec espacement normal"8D00
MESSAGE1 ASC  "Démo.      avec 5 points pour chaque espace entre"
         ASC  "           les lettres"8D00
MESSAGE2 ASC  "Démo.      sans espaces entre les lettres"8D00
MESSAGE3 ASC  "Démo.      de tous les caractères que l'on peut obtenir"
         ASC  "           à partir du clavier"8D
         ASC  "DEMO       : ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnop"
         ASC  "qrstuvwxyz1234567890&é'(§è!çà)-^$¨*ù`%£,;:=?./+<>"
         HEX  00
```

Récapitulation WRITE560.DEMO

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :
BSAVE WRITE560.DEMO, A$1000,L$8AB

```
1000:4C 91 16 90 09 8D 8E 16
1008:8E 8C 16 8C 8D 16 AD FF
1010:FF F0 13 C9 8D F0 10 C9
1018:A0 90 1D C9 C0 90 24 C9
1020:E0 90 37 4C 71 10 60 AD
1028:8E 16 18 69 0A 8D 8E 16
1030:A9 00 8D 8C 16 8D 8D 16
1038:EE 0F 10 D0 03 EE 10 10
1040:4C 0E 10 38 E9 A0 48 A9
1048:A2 85 FA A9 11 85 FB 68
1050:A8 B9 22 16 8D 8B 16 4C
1058:85 10 38 E9 C0 48 A9 A2
1060:85 FA A9 12 85 FB 68 A8
1068:B9 43 16 8D 8B 16 4C 85
1070:10 38 E9 E0 48 A9 A2 85
1078:FA A9 13 85 FB 68 A8 B9
1080:63 16 8D 8B 16 98 0A 0A
1088:0A 18 65 FA 85 FA 90 02
1090:E6 FB A9 08 8D 8F 16 20
1098:BF 10 AD 8B 16 18 69 02
10A0:6D 8C 16 8D 8C 16 90 03
10A8:EE 8D 16 AD 8D 16 C9 02
10B0:D0 0A AD 8C 16 C9 2A 90
10B8:03 4C 27 10 4C 38 10 8D
10C0:54 C0 AD 8E 16 AA AD 8C
10C8:16 8D 88 16 AD 8D 16 8D
10D0:89 16 20 3F 11 A0 00 20
10D8:34 11 8C 90 16 8A 48 A9
10E0:00 8D 8A 16 B1 FA AE 87
10E8:16 F0 0C CA 0A 48 0A 2E
10F0:8A 16 68 E0 00 D0 F4 8D
10F8:07 11 AD 86 16 4A A8 B0
1100:03 8D 55 C0 B1 26 49 FF
1108:91 26 8D 54 C0 AE 86 16
1110:E8 8A 4A A8 B0 03 8D 55
1118:C0 B1 26 4D 8A 16 91 26
1120:8D 54 C0 68 AA E8 E0 C0
1128:F0 09 AC 90 16 C8 CC 8F
1130:16 D0 A4 60 BD 62 15 85
1138:27 BD A2 14 85 26 60 D8
1140:A9 00 8D 84 16 8D 85 16
1148:8D 86 16 AD 89 16 D0 0B
1150:AD 88 16 C9 07 B0 04 8D
1158:87 16 60 EE 86 16 AD 84
1160:16 18 69 07 8D 84 16 90
1168:03 EE 85 16 AD 85 16 CD
1170:89 16 D0 E7 AD 88 16 CD
1178:84 16 90 0C 38 ED 84 16
1180:C9 07 B0 D7 8D 87 16 60
1188:AD 84 16 38 ED 88 16 8D
1190:96 11 A9 07 38 E9 FF C9
1198:07 B0 C0 CE 86 16 8D 87
11A0:16 60 00 00 00 00 00 00
11A8:00 00 01 01 01 01 01 00
11B0:01 00 05 05 05 00 00 00
11B8:00 00 0C 02 02 07 02 02
11C0:1D 00 04 1E 05 0E 14 0F
11C8:04 00 03 13 08 04 02 19
11D0:18 00 02 05 05 02 15 09
11D8:16 00 06 06 02 01 00 00
11E0:00 00 04 02 01 01 01 02
11E8:04 00 01 02 04 04 04 02
11F0:01 00 04 15 0E 04 0E 15
11F8:04 00 00 04 04 1F 04 04
1200:00 00 00 00 00 06 06 02
1208:01 00 00 00 00 07 00 00
1210:00 00 00 00 00 00 00 03
1218:03 00 00 10 08 04 02 01
1220:00 00 0E 11 19 15 13 11
1228:0E 00 02 03 02 02 02 02
1230:07 00 0E 11 10 0C 02 01
1238:1F 00 1F 10 08 0C 10 11
1240:0E 00 08 0C 0A 09 1F 08
1248:08 00 1F 01 0F 10 10 11
1250:0E 00 1C 02 01 0F 11 11
1258:0E 00 1F 10 08 04 02 02
1260:02 00 0E 11 11 0E 11 11
1268:0E 00 0E 11 11 1E 10 08
1270:07 00 00 03 03 00 03 03
1278:00 00 00 00 02 00 02 02
1280:01 00 08 04 02 01 02 04
1288:08 00 00 00 07 00 07 00
1290:00 00 01 02 04 08 04 02
1298:01 00 1E 33 30 18 0C 00
12A0:0C 00 08 10 0E 10 1E 11
12A8:1E 00 18 14 12 11 1F 11
12B0:11 00 0F 11 11 0F 11 11
12B8:0F 00 0E 11 01 01 01 11
12C0:0E 00 0F 11 11 11 11 11
12C8:0F 00 1F 01 01 0F 01 01
12D0:1F 00 1F 01 01 0F 01 01
12D8:01 00 0E 11 01 01 19 11
12E0:1E 00 11 11 11 1F 11 11
12E8:11 00 07 02 02 02 02 02
12F0:07 00 1C 08 08 08 08 09
12F8:06 00 11 09 05 03 05 09
1300:11 00 01 01 01 01 01 01
1308:0F 00 41 63 55 49 41 41
1310:41 00 41 43 45 49 51 61
1318:41 00 0E 11 11 11 11 11
1320:0E 00 0F 11 11 0F 01 01
1328:01 00 0E 11 11 11 15 09
1330:16 00 0F 11 11 0F 05 09
1338:11 00 0E 11 01 0E 10 11
1340:1C 00 1F 04 04 04 04 04
1348:04 00 11 11 11 11 11 11
1350:0E 00 11 11 11 11 11 0A
1358:04 00 41 41 41 49 55 63
1360:41 00 11 11 0A 04 0A 11
```

```
1368:11 00 11 11 11 0A 04 04
1370:04 00 1F 10 08 04 02 01
1378:1F 00 02 05 02 00 00 00
1380:00 00 00 00 0E 01 01 0E
1388:04 02 0E 01 06 09 06 08
1390:07 00 04 0A 11 00 00 00
1398:00 00 00 00 00 00 00 00
13A0:00 0F 01 02 04 00 00 00
13A8:00 00 00 00 06 08 0E 09
13B0:0E 00 01 01 01 07 09 09
13B8:07 00 00 00 0E 01 01 01
13C0:0E 00 08 08 08 0E 09 09
13C8:0E 00 00 00 06 09 07 01
13D0:0E 00 04 0A 02 02 07 02
13D8:02 00 00 00 0E 09 09 0E
13E0:08 06 01 01 01 07 09 09
13E8:09 00 02 00 03 02 02 02
13F0:07 00 08 00 0C 08 08 08
13F8:09 06 01 01 09 05 03 05
1400:09 00 03 02 02 02 02 02
1408:07 00 00 00 1B 15 15 11
1410:11 00 00 00 07 09 09 09
1418:09 00 00 00 06 09 09 09
1420:06 00 00 00 07 09 09 07
1428:01 01 00 00 0E 09 09 0E
1430:08 08 00 00 0D 03 01 01
1438:01 00 00 00 0E 01 06 08
1440:07 00 02 02 0F 02 02 02
1448:0C 00 00 00 09 09 09 0D
1450:0B 00 00 00 11 11 11 0A
1458:04 00 00 00 11 11 15 15
1460:1B 00 00 00 11 0A 04 0A
1468:11 00 00 00 09 09 0E 08
1470:08 07 00 00 1F 08 04 02
1478:1F 00 08 04 06 09 0F 01
1480:0E 00 02 04 00 11 11 19
1488:16 00 04 08 06 09 0F 01
1490:0E 00 05 00 00 00 00 00
1498:00 00 0F 0F 0F 0F 0F 0F
14A0:0F 0F 00 00 00 00 00 00
14A8:00 00 80 80 80 80 80 80
14B0:80 80 00 00 00 00 00 00
14B8:00 00 80 80 80 80 80 80
14C0:80 80 00 00 00 00 00 00
14C8:00 00 80 80 80 80 80 80
14D0:80 80 00 00 00 00 00 00
14D8:00 00 80 80 80 80 80 80
14E0:80 80 28 28 28 28 28 28
14E8:28 28 A8 A8 A8 A8 A8 A8
14F0:A8 A8 28 28 28 28 28 28
14F8:28 28 A8 A8 A8 A8 A8 A8
1500:A8 A8 28 28 28 28 28 28
1508:28 28 A8 A8 A8 A8 A8 A8
1510:A8 A8 28 28 28 28 28 28
1518:28 28 A8 A8 A8 A8 A8 A8
1520:A8 A8 50 50 50 50 50 50
1528:50 50 D0 D0 D0 D0 D0 D0
1530:D0 D0 50 50 50 50 50 50
1538:50 50 D0 D0 D0 D0 D0 D0
1540:D0 D0 50 50 50 50 50 50
1548:50 50 D0 D0 D0 D0 D0 D0
1550:D0 D0 50 50 50 50 50 50
1558:50 50 D0 D0 D0 D0 D0 D0
1560:D0 D0 20 24 28 2C 30 34
1568:38 3C 20 24 28 2C 30 34
1570:38 3C 21 25 29 2D 31 35
1578:39 3D 21 25 29 2D 31 35
1580:39 3D 22 26 2A 2E 32 36
1588:3A 3E 22 26 2A 2E 32 36
1590:3A 3E 23 27 2B 2F 33 37
1598:3B 3F 23 27 2B 2F 33 37
15A0:3B 3F 20 24 28 2C 30 34
15A8:38 3C 20 24 28 2C 30 34
15B0:38 3C 21 25 29 2D 31 35
15B8:39 3D 21 25 29 2D 31 35
15C0:39 3D 22 26 2A 2E 32 36
15C8:3A 3E 22 26 2A 2E 32 36
15D0:3A 3E 23 27 2B 2F 33 37
15D8:3B 3F 23 27 2B 2F 33 37
15E0:3B 3F 20 24 28 2C 30 34
15E8:38 3C 20 24 28 2C 30 34
15F0:38 3C 21 25 29 2D 31 35
15F8:39 3D 21 25 29 2D 31 35
1600:39 3D 22 26 2A 2E 32 36
1608:3A 3E 22 26 2A 2E 32 36
1610:3A 3E 23 27 2B 2F 33 37
1618:3B 3F 23 27 2B 2F 33 37
1620:3B 3F 05 01 03 05 05 05
1628:05 03 03 03 05 05 03 03
1630:02 05 05 03 05 05 05 05
1638:05 05 05 05 02 02 04 03
1640:04 06 05 05 05 05 05 05
1648:05 05 05 05 03 05 05 04
1650:07 07 05 05 05 05 05 05
1658:05 05 07 05 05 05 03 07
1660:03 05 05 05 04 04 04 04
1668:04 04 04 04 03 04 04 03
1670:05 04 04 04 04 04 04 04
1678:04 04 05 05 05 04 05 05
1680:05 05 05 05 00 00 00 00
1688:00 00 00 00 00 00 00 00
1690:00 8D 0D C0 8D 52 C0 8D
1698:57 C0 8D 50 C0 8D 5E C0
16A0:8D B2 C0 8D B5 C0 8D B7
16A8:C0 8D 54 C0 A2 00 A0 00
16B0:20 34 11 98 48 4A A8 B0
16B8:03 8D 55 C0 AD 6D 17 91
16C0:26 8D 54 C0 68 A8 C8 98
16C8:48 4A A8 B0 03 8D 55 C0
16D0:AD 6E 17 91 26 8D 54 C0
16D8:68 A8 C8 C0 50 D0 D4 E8
16E0:E0 C0 F0 36 A0 00 20 34
16E8:11 98 48 4A A8 B0 03 8D
16F0:55 C0 AD 6D 17 91 26 8D
16F8:54 C0 68 A8 C8 98 48 4A
1700:A8 B0 03 8D 55 C0 AD 6D
1708:17 91 26 8D 54 C0 68 A8
1710:C8 C0 50 D0 D4 E8 E0 C0
1718:D0 94 A9 6F A0 17 8D 0F
1720:10 8C 10 10 A9 02 8D 9F
1728:10 A9 14 A2 0A A0 00 38
1730:20 03 10 A9 A0 A0 17 8D
1738:0F 10 8C 10 10 A9 05 8D
1740:9F 10 18 20 03 10 A9 DA
1748:A0 17 8D 0F 10 8C 10 10
1750:A9 00 8D 9F 10 18 20 03
1758:10 A9 00 A0 18 8D 0F 10
1760:8C 10 10 A9 02 8D 9F 10
1768:18 20 03 10 60 00 00 C4
1770:FB ED EF EE F3 F4 F2 E1
1778:F4 E9 EF EE A0 E4 A7 FB
1780:E3 F2 E9 F4 F5 F2 E5 A0
1788:E1 F6 E5 E3 A0 E5 F3 F0
1790:E1 E3 E5 ED E5 EE F4 A0
1798:EE EF F2 ED E1 EC 8D 00
17A0:C4 FB ED EF AE A0 E1 F6
17A8:E5 E3 A0 B5 A0 F0 EF E9
17B0:EE F4 F3 A0 F0 EF F5 F2
17B8:A0 E3 E8 E1 F1 F5 E5 A0
17C0:E5 F3 F0 E1 E3 E5 A0 E5
17C8:EE F4 F2 E5 A0 EC E5 F3
17D0:A0 EC E5 F4 F4 F2 E5 F3
17D8:8D 00 C4 FB ED EF AE A0
17E0:F3 E1 EE F3 A0 E5 F3 F0
17E8:E1 E3 E5 F3 A0 E5 EE F4
17F0:F2 E5 A0 EC E5 F3 A0 EC
17F8:E5 F4 F4 F2 E5 F3 8D 00
1800:C4 FB ED EF AE A0 E4 E5
1808:A0 F4 EF F5 F3 A0 EC E5
1810:F3 A0 E3 E1 F2 E1 E3 F4
1818:FD F2 E5 F3 A0 F1 F5 E5
1820:A0 EC A7 EF EE A0 F0 E5
1828:F5 F4 A0 EF E2 F4 E5 EE
1830:E9 F2 A0 C0 A0 F0 E1 F2
1838:F4 E9 F2 A0 E4 F5 A0 E3
1840:EC E1 F6 E9 E5 F2 8D C4
1848:C5 CD CF A0 BA A0 C1 C2
1850:C3 C4 C5 C6 C7 C8 C9 CA
1858:CB CC CD CE CF D0 D1 D2
1860:D3 D4 D5 D6 D7 D8 D9 DA
1868:A0 E1 E2 E3 E4 E5 E6 E7
1870:E8 E9 EA EB EC ED EE EF
1878:F0 F1 F2 F3 F4 F5 F6 F7
1880:F8 F9 FA B1 B2 B3 B4 B5
1888:B6 B7 B8 B9 B0 A6 FB A7
1890:A8 DD FD A1 DC C0 A9 AD
1898:DE A4 FE AA FC E0 A5 A3
18A0:AC BB BA BD BF AE AF AB
18A8:BC BE 00
```

TPom's

TPom's, programme du numéro précédent vous cause peut-être des soucis sur votre Apple //c. Trois octets méritent un changement pour vous tirer d'embarras. Faire :

```
BLOAD T.POMS
CALL -151
204C: 4C 55 20
BSAVE T.POMS,A$2000,L4033
```

Un couper/coller un peu violent nous a conduit à un humour involontaire : l'article précise que la carte Super Série Apple est nécessaire sur le //c..

Un nouveau produit Pom's :

Editeur VIDEOTEX

Votre micro-serveur fonctionne et vous avez décidé de réaliser quelques belles pages de menus à l'exemple des serveurs professionnels. Muni des normes Vidéotex et de votre traitement de texte favori, vous vous êtes attelé à l'ouvrage... Aïe! Quelle besogne de bénédictin !

Le programme CP/M édité par Pom's simplifie grandement la réalisation de telles pages .

Le programme ÉDITEUR

Les éditeurs Vidéotex peuvent se subdiviser en deux catégories :

- les éditeurs statiques
- les éditeurs dynamiques.

Pour les différencier, observez les mouvements du curseur. S'il balaye laborieusement l'écran comme le spot d'une image TV, de gauche à droite et de haut en bas, alors il s'agit probablement d'un éditeur statique. Au contraire, s'il se promène allègrement dans toutes les directions, alors il s'agit d'un éditeur dynamique. Ce type d'éditeur permet un effet d'animation, un peu comme le peintre dessine sur sa toile. Il est possible de simuler par exemple le déplacement d'un petit objet sur l'écran. Notre éditeur est de **type dynamique**.

Il est écrit en **Turbo-Pascal (CP/M)** et fonctionne sur **Apple //e** et sur **Apple][+** avec une **carte 80 colonnes** et une **carte Z80**.

On peut l'utiliser avec un ou deux lecteurs de disquettes, avec pour inconvénient si vous n'avez qu'un seul lecteur que les images seront sauvegardées sur la disquette programme.

Utilisation, possibilités

Le déplacement du curseur est commandé par les touches fléchées (gauche, droite, bas, haut). Sur l'Apple][+, les déplacements vers le haut et vers le bas s'obtiennent par CTRL-K et CTRL-J. À tout moment, la position du curseur, horizontale et verticale, est mise à jour sur le tableau de bord.

Une fois le curseur en place, que peut-on y écrire ? Du texte ou du graphique, c'est l'objet de l'une de commandes clavier.

Il faut alors fixer des attributs qui diffèrent selon le mode :

Mode texte

- la couleur du caractère (Noir, Bleu, Rouge, Magenta, Vert, Cyan, Jaune, Blanc) ;
- la couleur du fond ;
- la fixité ou le clignotement ;
- l'inversion de fond, encore appelée positif/négatif ;
- le soulignement ;
- la taille du caractère :
  - taille normale
  - taille double
  - hauteur et largeur double.

Mode graphique

- la couleur du caractère ;
- la couleur du fond ;
- le clignotement ;
- la disjonction.

Lors du déplacement du curseur, les attributs courants peuvent être préservés ou détruits, c'est au choix.

Presque tous les caractères d'un clavier AZERTY et d'un clavier Minitel sont accessibles. Ceci n'est possible qu'en attribuant à une même touche plusieurs significations selon :

- sa position, 'shiftée' ou non 'shiftée' ;
- son rangement dans les caractères accentués ou dans les caractères spéciaux.

On peut obtenir :

& é § è ç à ^ 1/4 1/2 % ù £ 3/4 # | \ ][_ ¨ ° ¨ ≠ Æ Ø

À tout moment on peut simuler l'envoi de l'image sur le Minitel pour contrôle.

Les corrections sont possibles de deux manières :

- caractère par caractère, à l'aide de la flèche gauche ou de la touche Del. Attention cependant à certains attributs qui ne sont pas modifiés (couleur de fond et soulignage) ;
- par effacement (Ctrl-X) du dernier bloc entré avant le dernier positionnement du curseur.

Enfin – charme de notre éditeur dynamique – on peut inclure dans le fichier des commandes ou des caractères non-standard :

- un effacement d'écran entre deux images successives ;
- une sonnerie pour prévenir par exemple d'une erreur ;
- un caractère ou une séquence de caractères ASCII indisponibles au clavier. Les codes ASCII de 32 à 127 sont autorisés. Ceci permet théoriquement de disposer de tous les effets Vidéotex à condition d'en connaître la norme.

Un même fichier peut regrouper une succession d'images séparées par des effacements d'écran ou par des effacements partiels donnant un effet d'animation.

Inversement, un fichier-image sur disquette peut être rappelé en mémoire et visualisé sur l'écran.

Éditeur Vidéotex :
200,00 F TTC Franco, 'sources' compris.
Bon de commande page 75

On trouve plus facilement avec 10 ans

Certaines évolutions sont aussi importantes que des révolutions.

Le plus difficile quand on débute, c'est de débuter. Car toutes les propositions d'emploi demandent une certaine expérience. Chercher un emploi est donc un problème insoluble. Pas pour l'Apple IIGS, il débute avec 10 ans d'expérience.

Comment ? C'est très simple, il a une architecture double : un nouvel ordinateur puissant traitant le graphique et le son, entoure un Apple II classique et miniaturisé. Il profite ainsi de 10 ans de développement de logiciels et d'expérience. Un grand progrès réalisé en partie grâce au microprocesseur 65C816, un 16 bits descendant du 65C02 de l'Apple II. L'Apple IIGS fonctionne à deux vitesses : 2,8 MHz en mode natif ; en mode émulation (c'est-à-dire lorsque vous utilisez un programme de votre ancien Apple II) vous avez le choix entre 1 MHz et 2,8 MHz.

Débuter connu, c'est 16.000 propositions d'emploi.

Avec l'IWM (Integrated Woz Machine*) l'Apple IIGS peut recevoir indifféremment des lecteurs de disquettes 3,5 pouces 800 Ko ou des lecteurs 5,25 pouces 140 Ko, ou les faire cohabiter, ce qui contribue à vous faire profiter de la plus grande bibliothèque de logiciels du monde. Le nombre de 16.000 logiciels étant estimatif car la seule chose qu'un Apple IIGS soit incapable de calculer, c'est le nombre d'emplois qu'il est capable de tenir.

Avec le nouveau système d'exploitation ProDOS, l'Apple IIGS permet la hiérarchisation des fichiers et des catalogues à la manière de

** En hommage à Steve Wozniak, co fondateur de la Société Apple et créateur de l'Apple II.*

C.L.M. & BBDO.

un emploi quand on débute d'expérience.

Macintosh, et la connexion de nouveaux périphériques.

G et S, cela veut dire Graphique et Son.

L'imagination débridée des développeurs suffira-t-elle pour utiliser les 4096 nuances de couleurs disponibles ? C'est probable. Ce qui est certain, c'est qu'une résolution de 640 points sur 200 lignes pour 4 couleurs, 320 points sur 200 lignes pour 16 couleurs va donner beaucoup de relief à certaines démonstrations. Si vous n'en croyez pas vos yeux, vous n'en croirez pas vos oreilles non plus. Le coprocesseur "son" choisi par Apple a déjà une brillante carrière derrière lui puisqu'il est employé par les plus grands synthétiseurs du marché. C'est l'ENSONIQ, capable de générer 16 voix, et il ne lui manque même pas la parole.

Les relations nécessaires pour réussir.

En 10 années, l'Apple II a eu tout le loisir de se faire d'excellentes relations de travail, imprimante, modem, disque dur, que l'Apple IIGS continuera à entretenir et développer ; celui-ci peut désormais être connecté au réseau AppleTalk et gérer la LaserWriter réservée jusqu'à présent à Macintosh. Il peut aussi recevoir le disque dur SCSI.

L'Apple IIGS possède 7 connecteurs d'extension permettant, à l'aide d'une multitude de cartes d'interfaces, de le relier à tous les types d'unités périphériques.

Acheter un Apple, c'est entrer dans le Club Apple pour échanger des informations, accéder au support technique par téléphone 7 jours sur 7, ou aux services télématiques du Club.

Apple présente l'Apple IIGS.

Apple

Apple, le logo Apple, Apple IIGS, ProDOS, Macintosh, AppleTalk, LaserWriter et GS Paint sont des marques déposées par Apple Computer Inc.

ESSAI

Writer +

Philippe Mathieu

Writer + est un logiciel de traitement de texte de "2ème génération", proposé par A.C.I., qui nous a déjà apporté 4ème dimension.

Comme la plupart des logiciels protégés, Writer + est copiable et peut être installé sur disque dur, sachant que, lors du démarrage, la disquette originale devra être temporairement introduite (dans ce cas particulier, de façon aléatoire).

Il peut tourner sur Macintosh 512Ko ou sur MacPlus, mais n'essayez pas de le faire fonctionner sur un "antique" Macintosh 128Ko. En fait, nous conseillons vivement le MacPlus. Qu'en est-il des nouveaux modèles de Macintosh ? L'Écho des Apple (N° 5, spécial Macintosh), classe Writer + dans les compatibles Macintosh SE.

Writer + possède toutes les fonctions classiques de traitement de textes, avec des avantages et des facilités par rapport à d'autres, avec aussi des inconvénients. Notre analyse ne sera pas approfondie sur ces fonctions classiques, mais nous insisterons sur tout ce qui fait que Writer + se présente comme un représentant de ce que nous avons appelé la 2ème génération.

En effet, Writer + possède d'une part des fonctions sophistiquées de traitement de texte, d'autre part un certain nombre de fonctions de mise en page faisant que, si on ne désire pas une mise en page trop complexe, ce seul logiciel permet des réalisations intéressantes en micro-édition.

Writer + est multi-fenêtres, le nombre de fenêtres n'étant pas limité, et on peut ouvrir à l'écran plusieurs documents ou exemplaires du même document, de façon à pratiquer des Copier/Coller entre fenêtres. Le document voulu peut être activé à partir du menu *Fichier* (contrairement à une habitude qui est de créer un menu spécial Écran, comme dans Excel par exemple).

Writer + autorise la redéfinition de "raccourcis" au clavier, propres à l'utilisateur. Si cette solution est séduisante, elle est aussi dangereuse. En effet, rien n'interdit (bien que l'utilisateur en soit avisé) d'utiliser pour une commande Writer + un raccourci classiquement utilisé dans les commandes habituelles, comme par exemple ⌘C ou ⌘X.

Un document peut, en largeur, occuper presque toute la place disponible dans le format de papier choisi. Par exemple, en format A4, la règle est graduée jusqu'à 20 centimètres.

Les fonctions de traitement de textes classiques

Writer +, comme la plupart de ses concurrents, propose au démarrage une règle sur laquelle peuvent être positionnés margeurs, marqueur d'indentation et tabulateurs.

Les différentes options de choix (polices de caractères, tailles et styles) sont accessibles par les menus Typo et Style.

Certaines commodités sont offertes, pour le déplacement "à la main" de paragraphes ou d'images, ou encore pour les recherches et remplacements qui peuvent porter non seulement sur des mots ou groupes de mots, mais sur les caractères, tailles et styles.

On peut, en une seule opération, modifier simultanément les caractères, leur taille et leur style. On peut aussi effectuer un basculement majuscules/minuscules. Plusieurs options sont proposées, et le maniement en est parfois délicat, selon qu'on unifie totalement le transfert ou qu'on procède différemment pour, par exemple, les majuscules après un point.

Un traitement statistique associé permet à tout moment de savoir quelle est la dimension du document en cours de création, ou d'une sélection de celui-ci, avec diverses informations sur les signes, mots et lignes.

Fonctions avancées de traitement de textes

Comme c'est le cas dans la plupart des nouveaux logiciels de traitement de texte, Writer + offre

une panoplie d'outils *facilitateurs*.

La césure automatique permet de couper un mot en fin de ligne selon les règles de la syntaxe française. Deux commodités sont offertes :

- forcer la césure par un trait d'union ;
- à l'inverse, refuser la césure, par exemple pour éviter de couper en deux "Macintosh".

Un glossaire très puissant est associé au logiciel. Les expressions du glossaire, qui ne sont théoriquement pas limitées en taille, sont désignées par un jeu d'abréviations libres.

Le texte peut être disposé en colonnes. Le nombre de colonnes n'est limité que par la largeur du papier, une colonne ne pouvant pas avoir moins de 1 cm de largeur (la documentation dit 1,5 environ...), et les colonnes sont forcément d'égales largeurs. La mise en place de ces colonnes est particulièrement aisée, ainsi que la modification de leur nombre. Il est possible de créer un "saut de colonne" forcé, au même titre qu'un saut de page. L'équilibrage du *remplissage* des colonnes est automatique.

Les colonnes constituent aussi un moyen détourné de placer côte à côte du texte et des illustrations, lesquelles peuvent provenir classiquement d'un album. Nous verrons plus loin une autre solution.

Un système de cadres permet de réaliser, dans des styles variés, l'entourage total ou partiel de paragraphes, colonnes, en-têtes et pieds de page, zone de notes, page entière, etc.

La gestion des notes de bas de page est assez sophistiquée, avec numérotation automatique et remise en page lors des modifications du texte.

Fonctions "insolites"

Les calculs

Grande originalité, Writer + possède des fonctions de calcul associées, dont le maniement n'est pas exagérément complexe, et qui offrent des possibilités intéressantes (beaucoup plus vastes que celles offertes par **More**). On passe par la création de cellules, en organisant ainsi un petit tableur sur mesure dans le texte (une fonction d'interprétation permet aussi d'introduire des formules de calcul directement dans le texte, en demandant à Writer + de faire le reste)

Nous avons particulièrement apprécié les possibilités de travail avec l'horloge du Macintosh.

Les fichiers

Un petit gestionnaire de fichier, simple et facile à mettre en œuvre, permet de faciliter les opérations de courrier personnalisé. Il ne faut pas y voir plus, et croire qu'on va pouvoir se passer d'un outil plus perfectionné.

Le lecteur ne sera pas surpris d'apprendre que ce fichier est compatible avec **4ème dimension**...

La gestion d'images

La gestion d'images est une grande originalité de Writer +, qui peut être très utile pour ceux qui manipulent couramment de nombreuses images : par exemple, tenue à jour d'un catalogue illustré, représentation de molécules, études de CAO, etc.

Des planches pouvant comporter 1 à 12 images peuvent être créées simplement, à partir d'illustrations type MacPaint ou MacDraw. Un fichier d'images peut comporter au maximum 10 planches, mais le nombre de ces fichiers n'est pas limité. La séquence d'opérations à accomplir est particulièrement simple :

- sélectionner la planche ;
- sélectionner l'image ;
- coller sur le texte.

À noter qu'il est ensuite possible de transformer l'image par déplacement ou changement de dimensions.

La gestion de paragraphes

Elle ressemble à la gestion d'images, fonctionnant par création de fichiers de paragraphes comportant au maximum 10 thèmes, avec 12 paragraphes par thème. Chaque paragraphe peut comprendre jusqu'à 32 000 caractères. Ceci venant en plus du glossaire, et le nombre de fichiers de paragraphes n'étant pas limité.

La gestion de tableaux

C'est une autre solution, plus souple et plus complète que la mise en colonnes, pour juxtaposer textes et illustrations et réaliser des tableaux où sont tracées lignes et colonnes.

Fonctions de mise en page

La création de structures

Writer + propose une fonction particulièrement puissante : la hiérarchisation d'un texte en un certain nombre de niveaux : chapitres, paragraphes, etc.

La mise en niveaux se fait à partir d'une fenêtre de dialogue. Elle est toujours descendante et consiste à :

- définir les niveaux eux-mêmes ;
- définir pour chacun une typographie et une indentation ;
- définir, si on le souhaite, une numérotation (6 possibilités sont offertes).

Ensuite, il suffira, soit a priori, soit a posteriori, d'indiquer dans la fenêtre de dialogue que le texte sur lequel on travaille est de tel niveau pour que, automatiquement, soient affectés les éléments prédéterminés pour ce niveau.

De plus, la numérotation est automatiquement remise à jour (c'est parfois un peu long) pour tout le document, en cas de suppression ou d'ajout d'un élément ayant le statut choisi.

La création d'un sommaire

Une fonction particulière permet de créer un sommaire à partir des niveaux définis, avec indication des numéros de page.

La création d'un index

On peut définir a priori une liste de mots qui devront faire l'objet d'un index alphabétique annexé au document.

La liste étant créée, on invoque la fonction *Index* qui produit une liste d'index comportant pour chaque mot défini :

- la liste des numéros des pages où il figure, séparés par des virgules ;
- une inscription de ce numéro en italique et en gras si le mot figure plusieurs fois dans la page.

La vérification de la mise en page

Il est possible de faire apparaître à l'écran une page entière, pour contrôler la mise en page. On peut également afficher à l'écran plusieurs pages (uniquement pour voir l'allure générale, car bien entendu aucun texte n'est lisible dans ce cas).

Apprendre Writer +

De la lecture de ce qui précède, on peut déduire aisément que l'apprentissage de Writer + n'est pas immédiat. La documentation, claire, est parfois un peu trop succincte. Il manque un index alphabétique, de sorte qu'il est parfois bien difficile d'y retrouver ce qu'on veut. Ce d'autant plus que la table des matières est elle-même embryonnaire. Voici pourtant deux fonctions qu'offre Writer +...

Nous conseillons à ceux qui abordent Writer + de ne pas chercher à réaliser du premier coup un apprentissage complet. C'est d'autant plus aisé qu'il est assez facile "d'ignorer" complètement certains thèmes, et de les aborder seulement quand les manipulations de base sont bien acquises.
Par exemple, on peut ne jamais créer de hiérarchie et de sommaire. Mais on apprendra utilement à s'en servir quand le besoin s'en fera sentir.

On peut aussi, pour commencer à placer des images, utiliser d'abord la technique classique du Copier/Coller, et apprendre ultérieurement à utiliser les fichiers d'images.

Le débutant sera aidé dans cette tâche par les exercices progressifs proposés dans la documentation. En revanche, on peut déplorer l'absence d'un fichier d'assistance en ligne.

Avantages, Faiblesses

L'avantage principal de Writer +, c'est sa grande richesse fonctionnelle. Bien entendu, tous les éléments de cette richesse ne seront pas utiles à tous.

Certains n'utiliseront jamais les fichiers d'images, l'emploi de l'Album étant suffisant pour couvrir leurs besoins. À l'inverse, d'autres choisiront Writer + à cause de cette fonction particulière.

Le fichier permettant le courrier personnalisé ne sera utilisé... que si on fait justement du courrier personnalisé ! Le glossaire sera suffisant pour certains, alors que d'autres auront un recours intensif aux fichiers de paragraphes.

Au chapitre des faiblesses, on peut noter une surabondance de menus, obligeant d'ailleurs à des abréviations non classiques comme *Fich* ou *Styl*. Il n'est pas toujours aisé de mémoriser les fonctions qui entrent dans chacun. Et ceci n'empêche pas le recours fréquent (trop fréquent à notre goût) à des fenêtres de dialogue additionnelles.

On peut s'interroger sur la véritable utilité des fonctions de calcul. Writer + aurait-il des ambitions d'intégré qui n'ose pas dire son nom ? Avec, déjà, un mini gestionnaire de fichier...

La gestion de la marge gauche est... curieuse. Impossible de se bloquer complètement à gauche. Sans compter quelques irrégularités de fonctionnement à l'impression.

Le réaffichage d'un écran complet, par exemple après modification de la typographie, est un peu long. Cela n'a toutefois rien de rhédibitoire.

Avantage ou inconvénient ? La possibilité de définir des raccourcis propres à l'utilisateur est très séduisante. La possibilité, par cette méthode, de redéfinir les raccourcis des menus de base nous semble toutefois extrêmement dangereuse.

Nous avons aussi vu quelques "bombes", pas trop quand même pour un logiciel de venue récente. Beaucoup, si l'on considère que cette venue récente a eu lieu avec un retard de près d'un an sur les premières annonces...

Conclusion

D'abord, un bel effort pour un produit français de valeur internationale. Writer + est un des traitements de texte les plus puissants du marché, sinon le plus puissant, avec, pour seul vrai rival, Word 3 de Microsoft. De nombreux utilisateurs de micro-édition simple pourront se contenter de Writer +, sans faire appel à un logiciel spécialisé de mise en page.

La vraie question est la suivante : vaut-il mieux un intégré rapide, divisant les fonctions pour traiter chacune d'elles de façon plus simple (mais avec, en corollaire, une moins grande facilité pour entre-croiser les diverses fonctions), ou un logiciel unique, mettant en permanence à disposition de l'utilisateur toutes les fonctions dont il peut avoir besoin (avec, en corollaire, un maniement de l'ensemble un peu plus complexe) ? Writer + propose une réponse du second type, à l'utilisateur de juger.

DÉCOMPACTEUR

Julien Thomas

Lorsque l'on écrit un programme qui sera exécuté à l'aide d'un interpréteur Basic, il est préférable d'opter pour la concision afin de rendre le logiciel le plus efficace possible.

Ceci encourage la multiplication des instructions sur une même ligne mais conduit généralement à une parfaite illisibilité. Avec un compilateur, et à moins d'être adepte d'une certaine forme d'ésotérisme, les programmes 'tassés' ne présentent pas d'intérêt puisque la position de l'instruction par rapport au code 'source' n'influe pas sur la vitesse de déroulement du logiciel. De plus, pendant la compilation, les messages d'erreur ne concernent que la première instruction incorrecte dans la ligne ; s'il y a trois instructions en faute, les deux dernières ne seront pas signalées.

Si l'écriture d'un programme 'aéré' est sans problème, il n'en va pas de même lorsqu'il s'agit de rendre lisible un programme créé dans l'optique d'une utilisation interprétée, les modifications manuelles se soldant généralement par des 'bugs' difficilement localisables.

Le but du petit programme – sans prétention – ici listé est donc de faire automatiquement ce travail fastidieux. Il traite les fichiers Basic en format TEXT (issus d'un éditeur de texte comme 'Edit', du Basic Microsoft avec l'option 'texte seul', etc.) et génère un fichier décompacté, lui aussi en format TEXT, baptisé d'un nom correspondant à celui du fichier original auquel on ajoute le suffixe ".M" pour 'Modifié'.

Le temps nécessaire au traitement ne peut être formulé avec exactitude puisqu'il peut varier dans des proportions importantes selon le style d'écriture du programme de départ ; cependant, et pour se donner une idée, on peut prendre comme exemple le programme "Minitel" – particulièrement dense – du numéro 27 de Pom's. "Minitel" est 'éclairci' en :

- 14 minutes avec le programme "Décompacteur" interprété ;
- 5 minutes avec le programme "Décompacteur" compilé (la disquette d'accompagnement de ce numéro contient les deux versions, la version compilée étant, bien sûr, une application 'double-cliquable').

• *Segment d'un programme Basic :*

```
S97:FRAMEPOLY VARPTR(D(N)):IF I=0 OR I=7 OR I=9 THEN TEXTMODE
    1:TEXTFACE 8 ELSE TEXTMODE 3:TEXTFACE 0
MOVETO D(N+2)+9,D(N+1)+11:PRINT A$(I):IF I=0 THEN TEXTFACE 8:MO
    VETO D(2)+9,D(1)+23:PRINT "Fin"
N=N+27:NEXT:FRAMEPOLY VARPTR(M(0)):FRAMEROUNDRECT VARPTR(N(0)
    ),4,4:TEXTMODE 1:TEXTFACE 0:MOVETO 227,31:PRINT "'  "":PUT
    (152,14),K,PSET:PUT(177,14),J,PSET:IF A3=1 AND DE THEN PUT(
    195,14),L,PSET
IF DT THEN PAINTROUNDRECT VARPTR(P(0)),4,4:TIMER ON
```

• *Le même segment après traitement :*

```
S97:
FRAMEPOLY VARPTR(D(N))
IF I=0 OR I=7 OR I=9 THEN TEXTMODE 1:TEXTFACE 8 ELSE TEXTMODE
     3:TEXTFACE 0
MOVETO D(N+2)+9,D(N+1)+11
PRINT A$(I)
IF I=0 THEN TEXTFACE 8:MOVETO D(2)+9,D(1)+23:PRINT "Fin"
N=N+27
NEXT
FRAMEPOLY VARPTR(M(0))
FRAMEROUNDRECT VARPTR(N(0)),4,4
TEXTMODE 1
TEXTFACE 0
MOVETO 227,31
PRINT "'  ""
PUT(152,14),K,PSET
PUT(177,14),J,PSET
IF A3=1 AND DE THEN PUT(195,14),L,PSET
IF DT THEN PAINTROUNDRECT VARPTR(P(0)),4,4:TIMER ON
```

Programme 'Décompacteur'

```
DATA AND,ABS,ALL,APPEND,ASC,AS,ATN,BASE,B
    EEP,BREAK,BUTTON
DATA BACKPAT,CALL,CDBL,CHAIN,CHR$,CINT,CL
    EAR,CLOSE,CLS
DATA COMMON,CONT,COS,CSNG,CVD,CVI,CVS,CIR
    CLE,CSRLIN,CVSBCD
DATA CVDBCD,DATA,DATE$,DEFINT,DEFSNG,DEFD
    BL,DEFSTR,DEF,DELETE
DATA DIM,DIALOG,EQV,EDIT,ELSE,END,EOF,ERA
    SE,ERL,ERROR,ERR,EXP
DATA EXIT,ERASERECT,ERASEOVAL,ERASEROUNDR
    ECT,ERASEARC
DATA ERASEPOLY,FIELD,FILES,FIX,FN,FOR,FRE
    ,FRAMERECT,FILLRECT
DATA FRAMEOVAL,FILLOVAL,FRAMEROUNDRECT,FI
    LLROUNDRECT
DATA FRAMEARC,FILLARC,FRAMEPOLY,FILLPOLY,
```

```
    GET,GOSUB,GOTO
DATA GETPEN,HEX$,HIDECURSOR,HIDEPEN,IMP,I
    F,INKEY$,INPUT,INSTR
DATA INT,INITCURSOR,INVERTRECT,INVERTOVAL
    ,INVERTROUNDRECT
DATA INVERTARC,INVERTPOLY,KILL,LEFT,LEN,L
    ET,LINE,LIST,LLIST,LOAD
DATA LOC,LOF,LOG,LPOS,LPRINT,LSET,LCOPY,L
    OCATE,LBOUND,LIBRARY
DATA LINETO,MOD,MERGE,MID$,MKD$,MKI$,MKS$
    ,MOUSE,MENU
DATA MKSBCD$,MKDBCD$,MOVETO,MOVE,NAME,NEW
    ,NEXT,NOT,OR,OCT$
DATA ON,OPEN,OPTION,OUTPUT,OFF,OBSCURECUR
    SOR,PEEK,POKE,POS
DATA PRINT,PUT,POINT,PRESET,PSET,PENSIZE,
    PENMODE,PENPAT
DATA PENNORMAL,PAINTRECT,PAINTOVAL,PAINTR
    OUNDRECT,PAINTARC
DATA PTAB,PAINTPOLY,PICTURE,RANDOMIZE,REA
    D,REM,RENUM,RESTORE
DATA RESUME,RETURN,RIGHT$,RND,RSET,RUN,RE
    SET,SAVE,SGN,SIN,SPACE$
DATA SPC,SQR,STEP,STOP,STR,STRING$,SWAP,S
    YSTEM,SUB,STATIC,SOUND
DATA SHARED,SCROLL,SETCURSOR,SHOWCURSOR,S
    HOWPEN,TAB,TAN,THEN
DATA TIME,TO,TRON,TROFF,TIMER,TEXTFONT,TE
    XTFACE,TEXTMODE,TEXTSIZE
DATA USING,USR,UBOUND,UCASE$,VAL,VARPTR,W
    AIT,WEND,WHILE,WIDTH
DATA WRITE,WINDOW,WAVE,XOR,CASE,ENDIF,IGN
    ORE,INCLUDE,PAGE,SELECT
DATA =,!,%,#

DEFINT A-Z
DIM Mots.Cle$(221)
TEXTFONT 4
TEXTSIZE 9
WIDTH 80
Retour.Chariot$=CHR$(13)
Delimiteur$=CHR$(34)
CLOSE
FOR Index=0 TO 221
READ Mots.Cle$(Index)
NEXT

Debut:
Fichier$=FILES$(1,"TEXT")
IF Fichier$="" THEN CLOSE:END
OPEN Fichier$ FOR INPUT AS 1
OPEN Fichier$+".M" FOR OUTPUT AS 2

Boucle.0:
Drapeau.Delimiteur=0
Chaine$=""
Boucle.1:
IF EOF(1) THEN CLOSE:GOTO Debut
Caractere$=INPUT$(1,1)
IF Caractere$=Delimiteur$ THEN Chaine$=Ch
    aine$+Caractere$:IF Drapeau.Delimiteu
    r THEN Drapeau.Delimiteur=0:GOTO Bou
    cle.1 ELSE Drapeau.Delimiteur=-1:GOT
    O Boucle.1
IF NOT Drapeau.Delimiteur THEN IF Caract
    ere$=":" THEN Verification
IF Caractere$=Retour.Chariot$ THEN Drapea
    u.Delimiteur=0:PRINT#2,Chaine$:PRINT
     Chaine$:GOTO Boucle.0
Chaine$=Chaine$+Caractere$:GOTO Boucle.1


Verification:
PRINT#2,Chaine$;:PRINT Chaine$;
IF INSTR(Chaine$,"IF ")=0 THEN GOSUB Sp
    :GOTO Boucle.0
PRINT#2,":";:PRINT ":";
Boucle.2:
IF EOF(1) THEN CLOSE:GOTO Debut
Caractere$=INPUT$(1,1)
IF Caractere$=Retour.Chariot$ THEN PRINT
    #2,"":PRINT:GOTO Boucle.0
PRINT#2,Caractere$;:PRINT Caractere$;:GO
    TO Boucle.2

Sp:
Index=0
Boucle.Sp:
IF INSTR(Chaine$,Mots.Cle$(Index)) THEN
    PRINT#2,"":PRINT:RETURN
Index=Index+1
IF Index=222 THEN PRINT#2,":":PRINT ":"
    :RETURN
GOTO Boucle.Sp
```


STRUCTUREXPERT™

La solution pour programmer de manière structurée en MSBASIC™ et ZBASIC™.
STRUCTUREXPERT™ vous permet de disposer d'un ensemble d'analyses qui peuvent être visualisées (fenêtres "scrollables"), sauvées séparément sur fichiers, ou imprimées :

- Arborescence complète (descendante et ascendante) des appels GOSUB et CALL.
- Listes triées sur plusieurs types de critères des GOSUB et des CALL.
- Statistiques : fréquence d'appel de chaque routine.
- Détection d'erreurs logiques (boucles non contrôlées, récursivité illégale, labels et routines non utilisés).

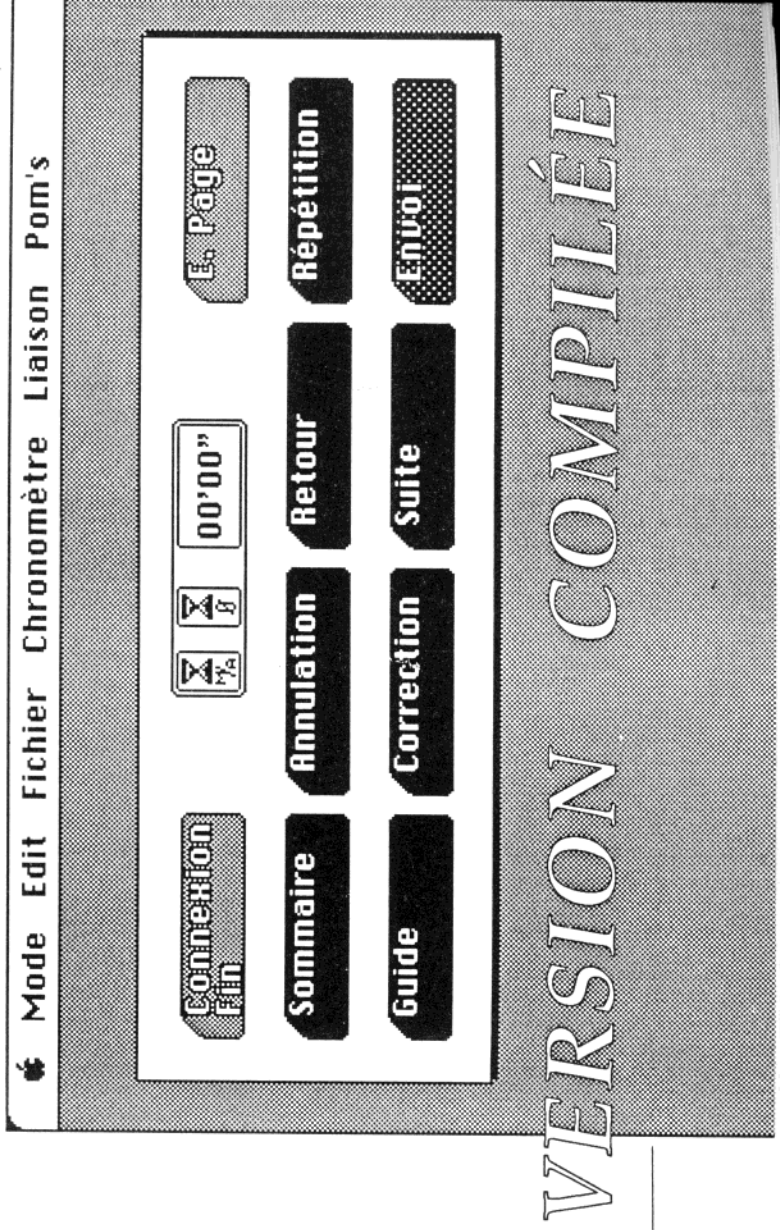
STRUCTUREXPERT™ 495 F TTC + 20 F de port
Paiement par CB ou chèques adressés à
Q.S.I. – 149, rue Oberkampf – 75011 PARIS
Tél. : (1) 42 49 37 95

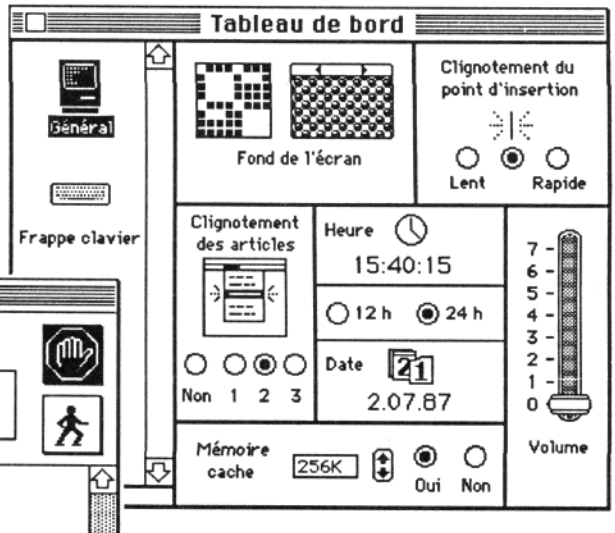
Disquette Pom's Macintosh n° 31

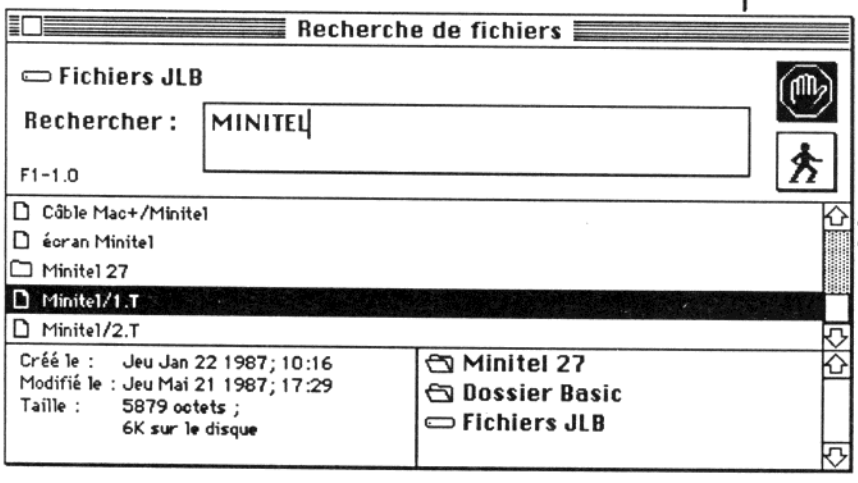
Vous y trouverez une nouvelle version – compilée – du programme **Minitel** publié dans le numéro 27 de Pom's. Cette version est autonome : il s'agit d'une application 'double-cliquable' ; les fichiers 'Minitel/1', 'Minitel/2' et 'Basic Runtime' ne sont plus nécessaires. Beaucoup plus rapide aussi : de 3 à 10 fois selon les fonctions utilisées. Enfin, notre programme est désormais compatible HFS ; les utilisateurs de Macintosh Plus et Macintosh SE vont enfin pouvoir utiliser largement leurs dossiers !

Le **nouveau système** – 4.0 – et le nouveau **Finder** – la version 5.4 – sont aussi sur la disquette 31. Le nouveau système d'exploitation est accompagné de nouvelles version des accessoires **Recherche de Fichiers**, **Sélecteur d'imprimantes** et **Tableau de bord**. Il comporte aussi de nombreuses améliorations moins visibles (nouvelles icônes...) ou absolument invisibles sans désassembleur (routines modifiées ou réécrites). Le 'Finder' a aussi eu droit à quelques améliorations : un article 'éteindre' sous l'article 'redémarrer'; un article 'ranger la fenêtre' qui devient 'aligner la sélection' lorsqu'il y en a une ; une corbeille qui se gonfle lorsque quelque chose est dedans ; des fenêtres d'informations repensées, etc....

Et puis, comme d'habitude, les fichiers, sources et programmes listés dans le cahier Macintosh de ce numéro 31 de Pom's.







MinAcc : l'accessoire minimum

Alain Bohec

Dans le numéro 22 de Pom's, on pouvait trouver plusieurs astuces qui permettaient une réduction importante du fichier 'Système', ceci à fin de gagner de la place sur les disquettes dites de démarrage.

Un de ces 'trucs' consistait en la suppression de tous les accessoires de bureau sauf un ; le 'Finder' ne supporterait pas qu'on les retire tous. Puisqu'il faut en laisser un dans le système, autant que celui-ci soit un accessoire 'minimum' : MinAcc occupe seulement 232 octets.

Il serait possible de faire encore plus petit, mais rien ne serait alors visible sur l'écran.

Fichier Exec 'MinAcc.Job'

```
Asm   MinAcc.Asm    Exec    Edit
Link  MinAcc.Link   Finder  Edit
```

Fichier 'MinAcc.Link'

```
]
/Resources
MinAcc
/output Fichier 'MinAcc'
/Type 'DFIL' 'DMOV'
$
```

Source 'MinAcc.Asm'

```
; Montre une fenêtre sans titre
; avec un petit rond dedans dont
; le pointeur est placé dans le
; champ WindowPic de la fenêtre.

RESOURCE 'DRVR' 27 'MinAcc'

; PRELIMINAIRES
; INCLUDES et EQUATES
INCLUDE    SysEqu.D
INCLUDE    ToolEqu.D
INCLUDE    Traps.Txt

;MOTS D'ENTREE
Derou
; Comportement. uniquement réponse
; (obligatoire) aux controles, pas
; de période, évènements de mise à
; jour (64) et mDown (2), pas de
;menu
    DC.W    $0400
    DC.W    0
    DC.W    $0040
    DC.W    0

; écarts vers les Routines
    DC.W    digoradur-Derou
    DC.W    OrnDone-Derou
    DC.W    OrnCtl-Derou
    DC.W    OrnDone-Derou
    DC.W    klozadur-Derou

;OUVERTURE

digoradur
    MOVEM.L    A1-A4,-(SP)
    MOVE.L     A1,A4

; sauvegarde du GrafPtr de
; l'application en cours

    CLR.L      -(SP)
    MOVE.L     SP,-(SP)
    _GetPort
    TST.L      dCtlWindow(A4)
    BNE        prenestrameus

; Si l'on n'a pas déja une fenêtre,
; il faut en fabriquer une :
    CLR.L      -(SP)
    CLR.L      -(SP)
    PEA        gwerenn
    CLR.L      -(SP)
    CLR.W      -(SP)
    MOVE.W     #18,-(SP)
    MOVE.L     #-1,-(SP)
    MOVE       #$0100,-(SP)
    CLR.L      -(SP)
    _NewWindow
    MOVE.L     (SP)+,A0
    MOVE.L     A0,
               dCtlWindow(A4)
    MOVE.W     dCtlRefNum(A4),
               WindowKind(A0)

; ROND
    MOVEM.L    A1-A4,-(SP)
    MOVE.L     dCtlWindow(A4),
               -(SP)
    _SetPort
    MOVE.L     dCtlWindow(A4),
               -(SP)
    CLR.L      -(SP)
    PEA        vue
    _OpenPicture
    _SetWindowPic
    PEA        q
    _FrameOval
    _ClosePicture
    MOVEM.L    (SP)+,A1-A4

; Si l'on a déja une fenêtre
prenestrameus
distro
    _SetPort
    MOVEM.L    (SP)+,A1-A4

OrnDone
    MOVEQ      #0,D0
    RTS

;FERMETURE
klozadur
    MOVEM.L    A1-A4,-(SP)
    MOVE.L     A1,A4

; même procédure
; qu'à l'ouverture :
    CLR.L      -(SP)
    MOVE.L     SP,-(SP)
    _GetPort
    MOVE.L     dCtlWindow(A4),
               -(SP)
    CLR.L      dCtlWindow(A4)
    _DisposWindow
    BRA.Sdistro

;CONTROLE
OrnCtl
    MOVEQ      #0,D0
    MOVE.L     jIODone,-(SP)
    RTS

;DONNEES
gwerenn   DC.W    40,1,60,21
q         DC.W    2,2,18,18
vue       DC.W    0,0,20,20

          END
```

Kino(201)

Kino

Alain Bohec

Principes de fonctionnement

Kino est une application qui permet de faire défiler à l'écran les images d'un album de type usuel ou de type "Multi-Scrap". Les images sont automatiquement centrées et s'affichent sur un fond qui peut être modifié.

Les albums d'un seul de ces deux types sont considérés comme des documents de *Kino*. Le finder lance donc *Kino* lorsque l'on clique deux fois sur l'icône d'un album du type adéquat.

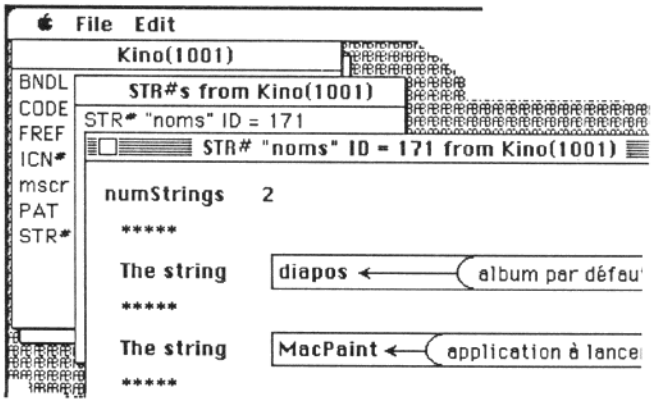
Lorsqu'on lance directement *Kino*, cette application cherche à ouvrir un album par défaut, ayant l'un quelconque des deux types ci-dessus. Le nom de cet album peut être choisi à l'avance, lors de la configuration de *Kino* (voir le paragraphe 'Configuration de Kino').

La durée d'affichage des images peut également être fixée à l'avance. Elle peut varier de quelques dixièmes de secondes à plus d'un an.

En sortie, *Kino* peut lancer une application quelconque, dont le nom peut également être choisi lors de la configuration.

Configuration de Kino

1] Album à ouvrir par défaut et application à lancer



Avant usage, *Kino* doit être configuré. Pour cela, un éditeur de ressources est nécessaire, par exemple *ResEdit*.

A l'aide de *ResEdit*, ouvrir la ressource de *Kino* intitulée :

STR# "noms" ID = 171

Cette ressource contient le nom de l'album que *Kino* cherchera à ouvrir dans le cas d'un lancement direct, et le nom de l'application à lancer lors de la sortie de *Kino*. À l'aide de *ResEdit*, vous pouvez modifier librement ces noms, qui sont initialement "diapos" et "MacPaint".

Lorsque le nom de l'application à lancer est celui de *Kino*, *Kino* se relance lui-même et refait défiler les images de "diapos".

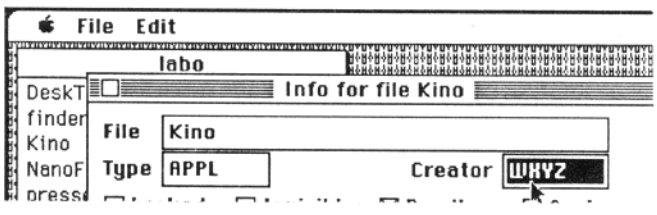
2] Durée d'affichage de chaque image et 'bip'

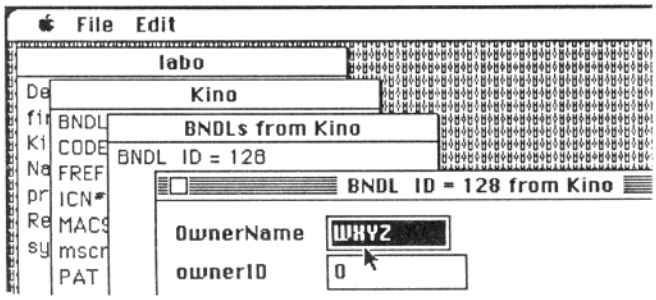
Le nom de *Kino* ne doit comporter aucune parenthèse, ou alors être de la forme "mot(n)" où mot est une suite de caractères sans parenthèse, et n un nombre compris entre 0 et 7F FF FF FF en hexadécimal, soit 2 147 483 647 en décimal. Ce nombre est la durée d'affichage des images en tierces. Par exemple, si le nom de *Kino* est "Kino(600)", chaque image restera affichée 10 secondes.

En fait, la durée d'affichage ne peut être inférieure aux quelques dixièmes de seconde qui sont nécessaires pour collecter l'image sur le disque. La durée maximale est de 414 jours.

Si la durée d'affichage est un nombre impair de tierces, un bip se fait entendre à l'apparition de chaque image.

3] Albums reconnus comme documents de Kino





Pour fixer le type des albums reconnus comme documents de *Kino*, il faut remplacer, dans les informations sur *Kino* et dans le 'bundle' de Kino, la séquence des 4 caractères WXYZ ci-dessus par la séquence adéquate qui est **MACS** pour les albums de type usuel, et **mscr** pour les albums de type *Multi-Scrap*.

Pour accéder à ces chaînes de caractères, il est nécessaire d'utiliser l'éditeur de ressources, que nous supposerons être *ResEdit*. Pour intervenir sur les informations sur *Kino*, sélectionner le nom de cette application dans la liste des applications, et choisir la rubrique adéquate du menu du fichier. Pour intervenir sur le 'bundle', ouvrir *Kino*, puis sa ressource de type BNDL et d'ID 128.

Il est également possible de changer le type et le créateur des albums.

4] Fond de l'écran

Pour modifier le fond sur lequel s'affiche les images présentées par *Kino*, il suffit de modifier avec un éditeur de ressource la ressource de type 'PAT ' de Kino. Il est bien entendu possible de choisir un fond blanc.

Installation et exploitation de Kino

Une fois *Kino* configuré, son bon fonctionnement est obtenu en le plaçant sur une disquette de démarrage. Dans le cas d'un disque formaté HFS, les albums à ouvrir par *Kino* doivent être placés dans le même dossier que *Kino*.

Celui qui doit être ouvert lors du lancement direct de *Kino* doit avoir le nom figurant dans la ressource STR# ID 171. Si ce nom ne correspond pas à un album, ou si l'album ayant ce nom ne contient pas d'images, l'exécution de *Kino* s'achève immédiatement.

Si *Kino* n'est pas placé sur la disquette de démarrage, il ne peut être lancé en cliquant deux fois sur l'icône d'un album.

Après lancement, *Kino* fait défiler les images de l'album choisi, ou celles de l'album par défaut.

Pour être efficaces, les actions décrites ci-dessous doivent intervenir quelques dixièmes de seconde après l'apparition d'une image à l'écran. *Kino* collecte en effet une image sur le disque aussitôt après avoir affiché la précédente.

Pour arrêter sur une image, presser le bouton de la souris.

Pour passer à l'image suivante avant que la durée d'affichage soit écoulée, appuyer sur le bouton de la souris et le relâcher, ou presser la touche de tabulation. Pour revenir à l'image précédente, presser la touche d'effacement.

Pour sortir de *Kino*, deux solutions sont possibles :

- presser la touche F, ce qui lance le finder ;
- presser la touche E, ce qui lance, si elle a été placée convenablement sur le disque, l'application ayant le nom figurant dans la ressource STR# ID 171, par exemple MacPaint.

Si *Kino* ne trouve pas cette application, le Finder est lancé.

En l'absence d'intervention, après avoir affiché la dernière image, *Kino* cherche à lancer l'application choisie, et, s'il ne la trouve pas, retourne au finder.

N.D.L.R. : l'application ResEdit se trouve sur la disquette Pom's Mac G. Un autre éditeur de ressources – REdit – est disponible sur la disquette d'accompagnement du numéro 24 de Pom's.

Source de l'Application 'Kino'

```
*                ¤¤¤¤¤¤¤¤¤¤¤¤
*                • StartUp •
*                ¤¤¤¤¤¤¤¤¤¤¤¤
*        Succession d'Images à l'écran
*-----------------------------------------
*    Installation d'un GrafPort pour desiner
*    sur l'écran entier
*    On quitte après un certain délai ou par
*    pression sur le bouton de la souris.
*.........................................
* A2 : adresse du SMAP
* A3 : PicHandle
* A4 : handle de la liste des noms
* D0 : ID de l'image
* D3 : rang de l'image dans l'album
* D4 : nombre d'images dans l'album
* D5 : délai
* D6 : date de changement
*.........................................
*¤¤¤ PRELIMINAIRES ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
*—— INCLUDES ————————————————————————
include  MacTraps.D   ; .TRAP
```

```
include   ToolEqu.D    ; equates du ToolBox
include   QuickEqu.D   ; equates de QuickDraw
include   mABcro.Txt   ; célèbres macros
*—— EQUATES ————————————————————————
CurApName       equ     $910
ParenthOuv      equ     $28
ParenthFerm     equ     $29
abID            equ     171
ppID            equ     -32768
HollDegwez      equ     $0000FFFF
mButDwnEvt      equ     1       ;bouton en bas
keyDwnEvt       equ     3       ;touche en bas
touchE          equ     14      ;N° de touche
touchF          equ     3       ;N° de touche
touchTab        equ     48      ;touche de tabul
touchBac        equ     51      ;touche de BackS
DelDef          equ     100     ;délai par déf.
HautEc          equ     342
LargEc          equ     512
LargCoin        equ     50      ;ovale du…
HautCoin        equ     30      ;…rect entourage
LongPtr         equ     80      ;long. pointeur
AdNom           equ     18      ;écart du nom
fnfErr          equ     -43     ;fichier introu.
*————————————————————————————————————
                XDEF    DERU
*.....................................
DERU
*¤¤¤ PREPARATIFS ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
*—— COLLECTE DU NOM DE LA BOBINE ————————
*+++ Liste des Noms ++++++++++++++++++++++++
* Cette liste doit être ouverte
* dans tous les cas
* pour lancer l'application suivante.
* Le handle est placé dans A4.
;*****************************************
    GetRes #'STR#',#abID
    dep .L,A4       ;handle de la liste
*+++ Paramètres ++++++++++++++++++++++++++++
    moveA.L #AppParmHandle,A0 ;adr handle
    moveA.L (A0),A0           ;handle
    moveA.L (A0),A0           ;adresse
    move.W  2(A0),D0          ;nombre
    beq.S   @10               ;0:pas de dias
    clr     2(A0)
*+++ Adresse du Nom ++++++++++++++++++++++++
*--- cas du lancement par des diapos -------
    lea     12(A0),A0         ;adres nom
    bra.S   ouverture
*--- lancement direct ----------------------
@10 moveA.L (A4),A0           ;adr liste
    lea     2(A0),A0          ;adr nom
*¤¤¤¤¤¤¤¤¤
ouverture
*—— OUVERTURE DU CARROUSEL ——————————————
    eP W                      ;pour RefNum
    emp .L,A0
    _OpenResFile
    dep .W,D0                 ;RefNum
    Si I.W,#-1,D0,EXIT
*+++ Etude du ResourceMap ++++++++++++++++++
*--- préparatifs ---------------------------
    moveA.L #TopMapHndl,A0
    moveA.L (A0),A0           ;handle -> A0
    moveA.L A0,A1             ;sauvegarde
    _HLock
    moveA.L (A0),A2           ;adresse -> A2
    addA.W  24(A2),A2         ;adr liste typ
*--- présence du SMAP ID = 0 ? -------------
;présence du SMAP ?
    move.L  #'SMAP',D1
    BSR     RECHTYPE
    bmi.S   @10
;vérification de l'ID
    move.W  6(A0),D0
    moveA.L A2,A0
    addA.W  D0,A0             ;adr 1er ID
    move.W  (A0),D0           ;premier ID
    bne.S   @10
    bra.S   @20
@10 emp .W,#1
    _SysBeep
@11 emp .W,#1
    _SysBeep
    bra.W   EXIT
*--- présence  d'images ? ------------------
@20 move.L  #'PICT',D1
    BSR     RECHTYPE
    bmi.S   @11
    move    4(A0),D4          ;nbre images-1
    addQ    #3,D4             ;nombre+2
*--- finitions -----------------------------
    moveA.L A1,A0             ;restauration
    _HUnlock
*¤¤¤¤¤¤¤
ACTIONS
*—— COLLECTE DU DELAI ————————————————————
    moveA.L #CurApName,A0
    move.B  (A0),D0           ;longueur nom
*+++ Recherche de la Parenthèse Ouvrante +++
@0  addQ.W  #1,A0             ;incrémente
    SSi I.B,#ParenthOuv,(A0),@5
    subQ.B  #1,D0
    beq.S   @20
    bra.S   @0                ;boucle
*+++ Recherche de la Parenthèse Fermante +++
@5  move.L  A0,A1             ;adr dans A0
    addQ.W  #1,A1
@10 addQ.W  #1,A1             ;incrémente
    SSi I.B,#ParenthFerm,(A1),@15
    bra.S   @10
*+++ Formation de la Phrase +++++++++++++++
@15 move.L  A1,D0             ;dernière adr
    sub.L   A0,D0             ;longueur+1
    sub.B   #1,D0             ;longueur
    move.B  D0,(A0)
*+++ Conversion en Délai ++++++++++++++++++++
    emP .W,#1
    _Pack7
    move.L  D0,D5             ;délai = D5
    bra.S   geo
@20 move.L  #DelDef,D5        ;délai défaut
*¤¤¤
```

Kino(201)

```
geo
*—— INSTALLATION DU GRAFPORT ————————————
*+++ Initialisations et Collectes +++++++++
*--- QuickDraw et curseur ------------------
    pea     thePort(A5)
    _InitGraf
    _InitCursor
    _HideCursor
*--- ouverture du GrafPort ----------------
    pea     g
    _OpenPort
*--- motif --------------------------------
    CollRes #abID,Pattern
    dep .L,A0
    _Hlock                      ;verrouillé
    move.L  (A0),A6             ;adresse
*+++ Signature +++++++++++++++++++++++++++++
; Si le bouton est pressé lorsque l'on
; arrive ici,
; les initiales AB apparraissent en haut à
; gauche, dans la barre de menu.
; Elles y restent tant que le bouton est
; pressé
;*****************************************
    ep W
    _Button
    dep .W,D0
    beq.S   suite
    P2Couic #4,#14,MoveTo
    pea     sign
    _DrawString
@10 ep W
    _Button
    dep .W,D0
    bne.S   @10
*+++ Peinture +++++++++++++++++++++++++++++++
suite
    pea     g+PortRect
    emp .L,A6
    _FillRect
*—— INITIALISATIONS ————————————————————
    move.L    #HollDegwez,D0
    _FlushEvents
    GetRes #'SMAP',#0           ;index
    dep .L,A0
    _HLock                      ;verrouillage
    moveA.L (A0),A2             ;adres SMAP
    moveQ   #1,D3               ;1ere image
    BSR     Prepa
*¤¤¤ DEFILE DES IMAGES ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
TRO
*—— INCREMENTATION ————————————————————
    addQ    #1,D3               ;compteur
    Si .B,D3,D4,EXIT
*—— DESSIN ————————————————————————————
*+++ Bip ? +++++++++++++++++++++++++++++++++
    btst    #0,D5
    beq.S   eff
    emp .W,#1
    _SysBeep
*+++ Préparation de l'Ecran +++++++++++++++
eff pea     g+PortRect
    emp .L,A6
    _FillRect
    pea     entourage
    emp .W,#LargCoin
    emp .W,#HautCoin
    _EraseRoundRect
*+++ Ah! +++++++++++++++++++++++++++++++++++
    emp .L,A3
    pea     cadre               ;rect figure
    _DrawPicture
    emp .L,A3
    _ReleaseResource
*—— ATTENTE ————————————————————————————
*+++ Préparatifs ++++++++++++++++++++++++++
    BSR     Prepa
    move.L  #0,A0
    _Delay
    move.L  D0,D6
    add.L   D5,D6
*+++ Boucle +++++++++++++++++++++++++++++++++
@10 move    #-1,D0              ;tous évènts
    lea     Degwez,A0           ;point sur év.
    _GetOSEvent
    beq.S   @20                 ;évènemt: @20
    cmp.L   T,D6
    bpl.S   @10                 ;boucle
    bra.S   TRO                 ;autre image
*--- un évènement a eu lieu ----------------
@20 move.W  Degwez,D2           ;numéro évt
    SSi I.W,#mButDwnEvt,D2,attente
    SSiNon I.W,#keyDwnEvt,D2,@10
    move.B  TouNo,D0            ;No de touche
    cmp.B   #touchF,D0          ;retour Finder
    bne.S   @30                 ;F = Finder
    _ExitToShell
@30 SSi I.B,#touchE,D0,EXIT
    SSi I.B,#touchTab,D0,TRO
    SSi I.W,#2,D3,@10
    SSiNon I.B,#touchBac,D0,@10
    subQ    #2,D3
    BSR     Prepa
    bra.S   TRO                 ;on continue
*.....................................................
*   L'image reste affichée tant que que le
*   bouton de la souris est pressé
;*****************************************
attente
    ep W
    _Button
    dep .W,D0
    bne.S   attente
    bra.S   TRO
*¤¤¤ SORTIE ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
EXIT
*—— COLLECTE DU NOM ————————————————————
    moveA.L (A4),A1             ;adr list noms
    moveQ   #3,D0               ;nbre de noms
    add.B   2(A1),D0
    addA    D0,A1
*—— VERIFICATION ———————————————————————
    move.L  #LongPtr,D0
```

Kino(201)

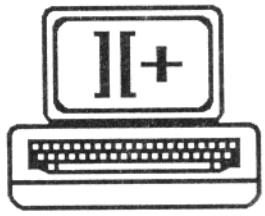
```
    _NewPtr,CLEAR
    _GetVol
    move.L  A1,AdNom(A0)
    _GetFileInfo
    SSi I.W,#fnfErr,D0,@10
*—— LANCEMENT ————————————————————————
    mouv0 L,A1,anv
    _Launch
@10 _ExitToShell
*££££££££££££££££££££££££££££££££££££££££
*¤¤¤ SOUS-ROUTINES ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
RECHTYPE
*—— RECHERCHE D'UN TYPE DANS UN FICHIER ——
; modifie A0,D0
; entrée : adresse de la liste des
; types : A2
; type recherché : D1
; sortie : N = 0 si type trouvé,
; A0 : adresse du type
; N = 1 sinon
;****************************************
    lea      2(A2),A0
    move.W   (A2),D0
@0  SSi .L,(A0),D1,@1           ;type 'PICT' ?
    subQ.W   #1,D0
    bmi.W    @1                 ;Sinon…
    addQ     #8,A0              ;incrém adr
    bra.S    @0                 ;…et on boucle
@1  tst.W    D0
    RTS
*¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
PREPA       ;No de l'image : D3
*—— PREPARATION DU DESSIN ———————————————
*+++ Calcul de l'ID ++++++++++++++++++++++++
* l'ID est placé dans D0
;****************************************
    move.W  #ppID,D0            ;initialise
    moveA.L A2,A0               ;adr du SMAP
@0  move.B  (A0),D1             ;rang
    SSi .B,D1,D3,@10
    addQ     #1,D0
    addQ     #1,A0
    bra.S    @0
*+++ Collecte de l'Image +++++++++++++++++++
@10 CollRes D0,Picture          ;PicHandle
    dep A.L,A3                  ;PicHandle: A3
*+++ Calcul de Rectangles +++++++++++++++++
@20
*--- mise à jour de vect ------------------
    pea      vect
    P2Couic #LargEc,#HautEc,SetPt
*--- écriture du rectangle provisoire à ----
*--- l'adresse cadre ----------------------
    moveA.L (A3),A0             ;adr image: A0
    lea      cadre,A1           ;adresse: A1
    move.L   2(A0),(A1)         ;tr  TopLeft
    move.L   6(A0),4(A1)        ;tr BottomR
*--- calcul de vect ----------------------
    emp .L,BotR
    pea      vect
    _SubPt
    emp .L,TopL
    pea      vect
    _SubPt
    lea      vect,A0
    asr.W    (A0)
    asr.W    2(A0)
*--- placement du rectangle (au centre) ----
    pea      cadre
    P2Couic vect+2,vect,OffSetRect
*--- rectangle d'entourage ----------------
    lea      entourage,A0
    move.W   cadre,D0
    asr.W    #1,D0
    move.W   D0,(A0)
    move.W   cadre+2,D0
    asr.W    #1,D0
    move.W   D0,2(A0)
    move.W   cadre+4,D0
    add.W    #HautEc,D0
    asr.W    #1,D0                 ;division 2
    move.W   D0,4(A0)
    move.W   cadre+6,D0
    add.W    #LargEc,D0
    asr.W    #1,D0                 ;division 2
    move.W   D0,6(A0)
*————————————————————————————————————————
    RTS

*££££££££££££££££££££££££££££££££££££££££
*¤¤¤ DONNEES ¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
*—— GRAPHIQUE ———————————————————————————
g           dcB.B    PortRec,0    ;GrafPort
vect        dc.W     0,0          ;centrage
cadre
 TopL :     dc.W     0,0
 BotR :     dc.W     0,0
entourage
            dc.W     0,0
            dc.W     0,0
*—— EVENEMENT ———————————————————————————
Degwez
            dc.W     0
            dc.W     0
 TouNo:     dc.B     0
            dc.B     0
 T:         dc.L     0            ;temps écoulé
            dc.L     0
            dc.W     0
*—— NOMS ————————————————————————————————
anv         dc.L     0
sign        dc.B     2
            dc.B     'AB'
*————————————————————————————————————————
            FIN
¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤
•••••••••••••••••••••••••••••••••••••••••
```

Kino(201)

Les nouvelles versions d'InterPom's

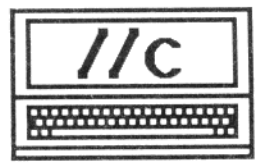
Version IBM® PC/AT™ et compatibles

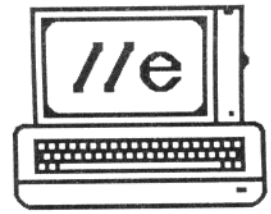
- Transmission par le réseau téléphonique de tous les types de fichiers et programmes entre deux IBM® PC/AT™ ou compatibles.
- Protocole 'intelligent' garantissant des transferts d'informations ultra-fiables.
- Émission de tous les types de fichiers et programmes vers les Macintosh.
- Émission des fichiers au format 'texte' vers les Apple //.
- Réception des fichiers 'texte' émis par les Apple // et Macintosh. Les fichiers au format 'texte' sont recodés (caractères accentués et spéciaux).

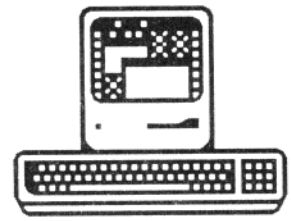


Version Apple //

- Transmission par le réseau téléphonique de tous les types de fichiers et applications entre deux Apple //.
- Protocole 'intelligent' garantissant des transferts d'informations ultra-fiables.
- Émission de tous les types de fichiers et programmes vers les Macintosh.
- Émission des fichiers au format 'texte' vers les IBM® PC/AT™ et compatibles.
- Réception des fichiers 'texte' émis par les IBM® PC/AT™ et Macintosh. Les fichiers au format 'texte' sont recodés (caractères accentués et spéciaux).
- Possibilité de réception sans recodage des fichiers 'texte' émis par les IBM® PC/AT™ et Macintosh.
- Utilisation possible avec le Modem intégré au Minitel, avec un Modem classique, ou en local entre deux Apple //, un Apple // et un PC™ ou encore un Macintosh et un Apple //.
- Possibilité d'utiliser le port intégré du IIGS.



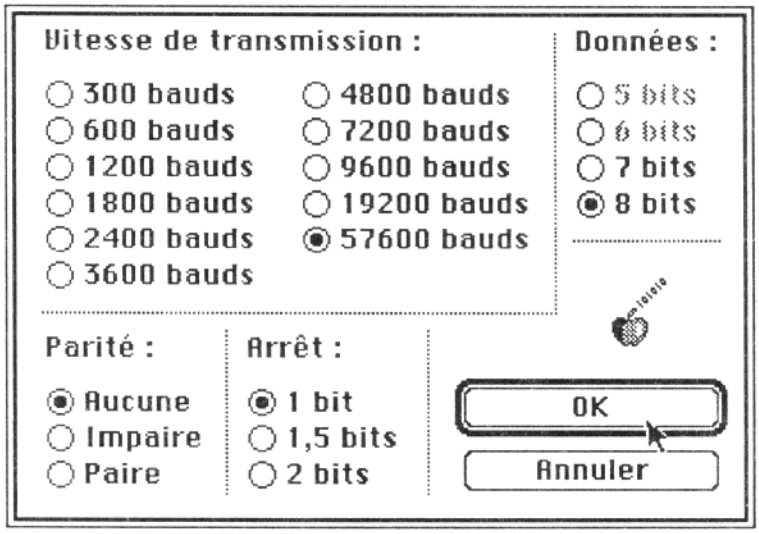




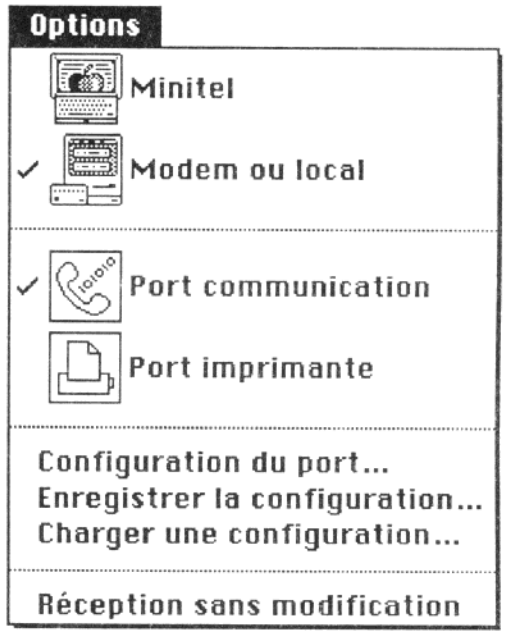
Version Macintosh

- Transmission par le réseau téléphonique de tous les types de fichiers (textes, polices de caractères, feuilles de calculs, accessoires de bureau...) et applications entre deux Macintosh.
- Protocole 'intelligent' garantissant des transferts d'informations ultra-fiables.
- Émission des fichiers au format 'texte' vers les Apple //.
- Émission des fichiers au format 'texte' vers les IBM® PC/AT™ et compatibles.

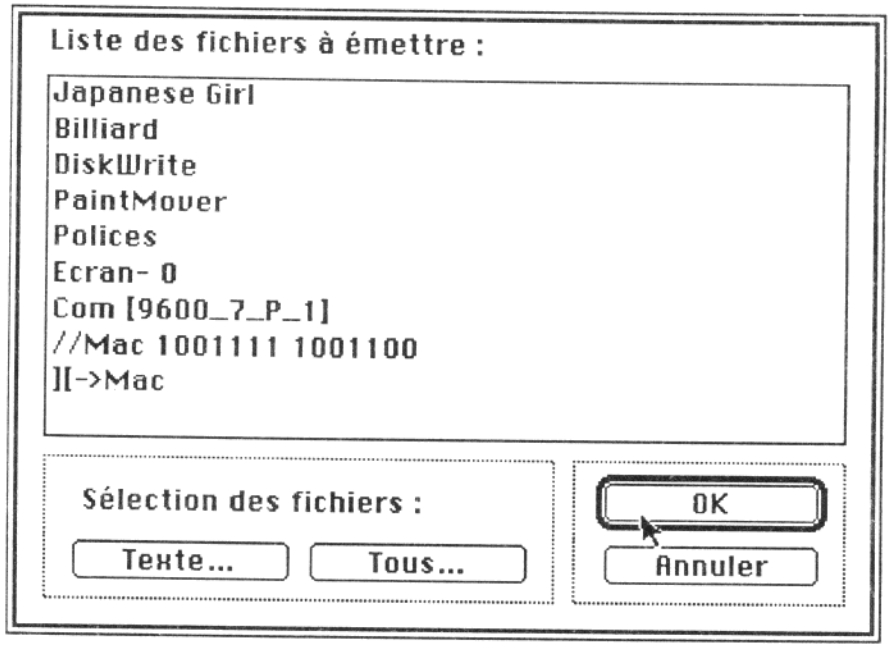
- Réception de tous les types de fichiers émis par les Apple // et IBM® PC/AT™. Les fichiers au format 'texte' sont recodés (caractères accentués et spéciaux) et stockés sous la forme de fichiers 'MacWrite' ; les autres types sont stockés sous la forme de fichiers 'Edit' (utilisables depuis n'importe quel programme acceptant le format TEXT).
- Possibilité de réception sans recodage des fichiers 'texte' émis par les Apple // et IBM® PC/AT™ ou compatibles.



- Option de transmissions groupées et automatiques de fichiers ou applications (jusqu'à dix). La possibilité demeure même dans le cas de fichiers ou applications répartis sur plusieurs volumes différents.
- Utilisation possible sur le port série 'Communication' ou sur le port 'Imprimante'. Le port 'Imprimante' n'est toutefois pas accessible si le réseau AppleTalk est connecté.
- Utilisation possible avec le Modem intégré au Minitel, avec un Modem classique, ou en local entre deux Macintosh, un Macintosh et un IBM® ou encore un Macintosh et un Apple //.
- Configuration possible du port courant pour une utilisation avec un Modem ou en local entre deux appareils. On peut ainsi transmettre des informations jusqu'à 57 600 bauds soit, en tenant compte du protocole utilisé, plus de 5 Ko par seconde.



- Sauvegarde possible de la configuration en cours dans l'optique de l'utilisation ultérieure d'une même configuration.
- Possibilité de 'démarrage configuré' par un 'double-clic' sur un fichier de configuration.
- Menu 'Quitter' permettant le départ vers une application au lieu du 'Finder'.
- Compatible tous environnements.



Bon de commande page 75

Disquettes Pom's Mac B et Mac C - Bon de commande page 74

Polices de la disquette Pom's Mac B

Portez ce bol de vieux whisky aux 257 juges qui fument

LORTEZ CE BOL DE VIEUX WHISKY AUX 257 JUGES QUI FUMENT

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257

Portez ce bol de vieux whisky aux 257 juges qui

Portez ce bol de vieux whisky aux 257 juges qui

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges

Portez ce bol de vieux whisky aux 257 juges

Polices de la disquette Pom's Mac C

PORTEZ CE BOL DE VIEUX WHISKY AUX 257 JUGES

Portez ce bol de vieux whisky aux

Portez ce bol de vieux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 25

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 25

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges

Portez ce bol de vieux whisky aux 257

Portez ce bol de vieux whisky aux 257 juges qui fument

Portez ce bol de vieux whisky aux 257 juges qui fument

PORTEZ CE BOL DE VIEUX WHISKY AUX 257

Il s'agit d'un système graphique double-haute résolution écrit en Pascal. COGO vous permet de manipuler des graphiques grâce à un langage de description des objets - points, angles- et à l'emploi de fonctions primitives de manipulation très puissantes : cercle, tangente, intersections, parallèles, etc. Il est ainsi possible de tracer des grilles, des cercles, des segments de droite, des tangentes communes à deux cercles, de calculer des distances, des angles...

L'éditeur permet une saisie rapide du langage. Une instruction COGO peut-être exécutée dès la saisie pour faciliter la mise au point, ou au sein d'un programme.

Vous avez un Apple //e avec Chat Mauve ou un //c ?
Vous avez Pascal 1.2 ?

Utilisez

COGO

Par Nicolas Montsarrat

Apple //e, //c

Ce programme, destiné à résoudre des problèmes de géométrie plane, comporte des instructions de stockage sur fichier afin de permettre la reprise d'un calcul.

200,00 F TTC, franco
Bon de commande page 74

DOS 3.3

][+
//e
//e+
//c
][gs

Un classique : le voyageur

Serge Cattan

Programme LE VOYAGEUR

```
10  REM  ****************************************
20 :
30  REM  CATTAN Serge        LE VOYAGEUR       11/86
40 :
50  REM ****************************************
60  ONERR  GOTO 2280
70 T$ = "----------------------------------------
   ------------------------------------"
80 D$ =  CHR$ (4)
90  HOME
100  INVERSE : PRINT  SPC( 40)
110  PRINT "              LE VOYAGEUR
    ";
120  PRINT  SPC( 40)
130  NORMAL
140  POKE 34,4: VTAB 8
150  PRINT "ANALYSE D'UN PROBLEME        --> 1"
160  PRINT "REPRISE ANALYSE EXISTANTE    --> 2"
170  PRINT "MODIFICATION DONNEES         --> 3"
180  PRINT "MODE D'EMPLOI                --> 4"
190  PRINT "RETOUR AU BASIC              --> 5"
200  POKE 36,11: VTAB 18: INPUT "VOTRE CHOIX ? ";R$
   :R =  VAL (R$)
210  ON R GOTO 260,230,240,3090,250
220  GOTO 200
230  GOSUB 2070: GOTO 570
240  GOSUB 2070: GOSUB 1580: GOTO 570
250  POKE 34,0: END
260  HOME
270  VTAB 8: PRINT "NOM DE L'ANALYSE (? POUR LE CAT
   ALOGUE) :": PRINT : INPUT "          - ";NE$
280  IF NE$ = "?" OR  LEN (NE$) = 0 THEN  GOSUB 237
   0: GOTO 260
290  PRINT : INPUT "NOMBRES DE TRAJETS  ? ";NR$
300 NR =  VAL (NR$): IF NR < 2 THEN  PRINT " UN CHI
   FFRE > 1 S.V.P.": GOTO 290
310  GOSUB 330: GOTO 400
320  REM ========== DIM TABLEAU ==================
   =
330  IF ME = 1 THEN  RETURN
340  DIM M(NR,NR),C(NR,NR),D(NR,NR),X(NR + 1),Y(NR
   + 1),V$(NR)
350 ME = 1: RETURN
360  HOME
370  REM ----------------------------------------
   ---
380  REM  SAISIE DES COUTS
390  REM ----------------------------------------
   ---
400  FOR I = 1 TO NR
410  PRINT "NOM DE LA VILLE ";I;" :";: INPUT V$(I)
420  NEXT I
430  HOME
440  FOR I = 1 TO NR
450  FOR J = 1 TO NR
460  IF I = J THEN C(I,J) = 1E + 30: GOTO 500
470  PRINT "TRAJET DE ";V$(I);" A ";V$(J);" ";: INP
   UT VT$
480 C(I,J) =  VAL (VT$)
490  IF C(I,J) = 0 THEN  INVERSE : PRINT "UN NOMBRE
    >0 S.V.P.": NORMAL : GOTO 470
500 D(I,J) = C(I,J)
510  NEXT J
520  PRINT
530  NEXT I
540  REM ----------------------------------------
   ---
550  REM  MODELE DE LITTLE
560  REM ----------------------------------------
   ---
```

Non, il ne s'agit pas de vous infliger une série TV sur la 5ème dimension, mais de vous proposer une solution à un vieux problème de récréation mathématique.

Il s'agit de déterminer le meilleur itinéraire à parcourir entre plusieurs villes, afin de minimiser un facteur quelconque, par exemple :

- le nombre de kilomètres ;
- le temps du parcours total.

Le modèle de Little permet d'obtenir un algorithme qui fournit une solution. Les seules contraintes sont :

- il est interdit de passer deux fois dans la même ville ;
- la fonction d'évaluation doit être obligatoirement minimisée (il est ainsi impossible d'utiliser une fonction qu'il faudrait maximiser).

Le programme LE VOYAGEUR, écrit en Applesoft donc facilement adaptable, permet de créer et de modifier à loisir les hypothèses d'un problème ; les solutions sont éditables sur imprimante (Epson RX80 mais vous n'aurez aucune difficulté à convertir les quelques lignes). En prime, le programme vous offre un *tutorial* pour vous familiariser avec les concepts manipulés.

```
570  GOSUB 1090: REM  * MODIF RESOLUTION *
580 M = 1E + 30:K = 0:R0 = 0:R1 = 0:R2 = 0:CS = 0:O
    UI = 0
590 K = K + 1
600  GOSUB 850: REM  * SUBSTITUTION *
610 HOT = OUI + CS
620 C(R1,0) = 1:C(0,R2) = 1:C(R2,R1) = 1E + 30
630 R0 = 0:X(K) = R1:Y(K) = R2
640  GOSUB 1090: REM  * MODIF RESOLUTION *
650 OUI = OUI + R0
660  IF OUI > M AND HOT > M THEN FIN = 1: GOTO 750
670  IF HOT < M AND OUI <  = HOT THEN M = HOT
680  IF OUI <  = HOT THEN 750
690  IF OUI < M THEN M = OUI
700  FOR I = 0 TO NR
710  FOR J = 0 TO NR:C(I,J) = M(I,J): NEXT J
720  NEXT I
730 C(X(K),Y(K)) = 1E + 30:OUI = HOT
740  GOSUB 1090: REM  * MODIF RESOLUTION *
750  IF K <  > NR - 1 AND FIN <  > 1 THEN 590
760  IF FIN = 1 THEN 830
770 K = K + 1
780  FOR I = 1 TO NR
790  FOR J = 1 TO NR
800  IF C(I,0) = 0 AND C(0,J) = 0 THEN X(K) = I:Y(K
    ) = J:I = NR:J = NR
810  NEXT J
820  NEXT I
830  GOTO 1470
840  REM
850  REM ----------------------------------------
    ---
860  REM  SUBSTITUTION
870  REM ----------------------------------------
    ---
880 CS = 1E - 30
890  FOR I = 1 TO NR
900  IF C(I,0) = 1 THEN 1040
910  FOR J = 1 TO NR
920  IF C(0,J) = 1 OR C(I,J) <  > 0 OR C(I,J) > 1E
    + 30 THEN 1030
930 M1 = 1E + 30
940  FOR J1 = 1 TO NR
950  IF C(0,J1) <  > 1 AND C(I,J1) < M1 AND J <  >
    J1 THEN M1 = C(I,J1)
960  NEXT J1
970 M2 = 1E + 30
980  FOR I1 = 1 TO NR
990  IF C(I1,0) <  > 1 AND C(I1,J) < M2 AND I1 <  >
     I THEN M2 = C(I1,J)
1000  NEXT I1
1010 MIN = M1 + M2
1020  IF MIN > CS THEN CS = MIN:R1 = I:R2 = J
1030  NEXT J
1040  NEXT I
1050  FOR I = 0 TO NR
1060  FOR J = 0 TO NR:M(I,J) = C(I,J): NEXT J
1070  NEXT I
1080  RETURN
1090  REM ----------------------------------------
    ----
1100  REM  MODIF RESOLUTION
1110  REM ----------------------------------------
    ----
1120  FOR I = 1 TO NR
1130  IF C(I,0) = 1 THEN 1250
1140 MIN = 1E + 30
1150  FOR J = 1 TO NR
1160  IF C(0,J) = 1 THEN 1190
1170  IF C(I,J) = 0 THEN F = 1:J = NR: GOTO 1190
1180  IF C(I,J) <  = (MIN) THEN MIN = C(I,J)
1190  NEXT J
1200  IF F = 1 THEN F = 0: GOTO 1250
1210  FOR J = 1 TO NR
1220  IF C(0,J) = 0 THEN C(I,J) = C(I,J) - MIN
1230  NEXT J
1240 R0 = R0 + MIN
1250  NEXT I
1260  FOR J = 1 TO NR
1270  IF C(0,J) = 1 THEN 1390
1280 MIN = 1E + 30
1290  FOR I = 1 TO NR
1300  IF C(I,0) = 1 THEN 1330
1310  IF C(I,J) = 0 THEN F = 1:I = NR: GOTO 1330
1320  IF C(I,J) < (MIN) THEN MIN = C(I,J)
1330  NEXT I
1340  IF F = 1 THEN F = 0: GOTO 1390
1350  FOR I = 1 TO NR
1360  IF C(I,0) = 0 THEN C(I,J) = C(I,J) - MIN
1370  NEXT I
1380 R0 = R0 + MIN
1390  NEXT J
1400  REM ----------------------------------------
    ----
1410  FOR I = 1 TO NR
1420  FOR J = 0 TO NR
1430  IF C(I,0) = 1 AND C(0,J) = 1 THEN C(J,I) = 1E
    + 30
1440  NEXT J
1450  NEXT I
1460  RETURN
1470  REM ----------------------------------------
    ----
1480  REM  AFFICHAGE RESULTAT
1490  REM ----------------------------------------
    ----
1500  GOSUB 2410
1510  VTAB 23: PRINT : INPUT "VOULEZ-VOUS SAUVEGARD
    ER CETTE ANALYSE ?";RE$
1520  IF  LEFT$ (RE$,1) = "O" THEN  GOSUB 1910
1530  REM
1540  PRINT :R$ = "": INPUT "VOULEZ-VOUS IMPRIMER C
    ES RESULTATS ?";R$
1550  IF  LEFT$ (R$,1) = "O" THEN  GOSUB 2710
1560  POKE 34,0: RUN
1570  REM ------------------------
1580  REM  MODIFICATIONS
1590  REM ------------------------
1600  HOME
1610  INPUT "MODIFIER LA VILLE DE DEPART ? ";RE$
1620  IF  LEFT$ (RE$,1) <  > "O" THEN  PRINT : GOTO
     1650
1630 RE$ = "": INVERSE : PRINT VD$;" ": NORMAL : IN
    PUT "NOUVEAU NOM ( ENTER=OK ) ? ";RE$
1640  IF  LEN (RE$) > 0 THEN VD$ = RE$
1650  INPUT "MODIFIER LE NOM D'UNE VILLE ? ";RE$
1660  IF  LEFT$ (RE$,1) <  > "O" THEN  PRINT : GOTO
     1710
1670  FOR I = 1 TO NR
1680 RE$ = "": INVERSE : PRINT V$(I);" ": NORMAL :
    INPUT "NOUVEAU NOM ( ENTER=OK ) ? ";RE$
1690  IF  LEN (RE$) > 0 THEN V$(I) = RE$
1700  PRINT : NEXT I
1710  FOR I = 1 TO NR
1720  IF VD$ = V$(I) THEN 1760
1730  NEXT I
```

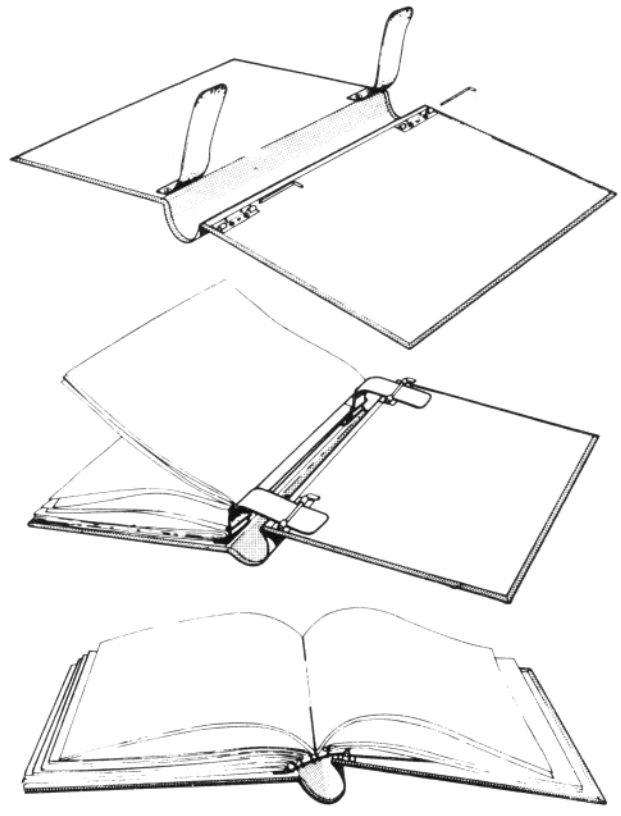
```
1740  PRINT : FLASH : PRINT "VILLE DE DEPART INCONN
   UE": NORMAL
1750  FOR T = 1 TO 1500: NEXT : GOTO 1600
1760  INPUT "MODIFIER UN TRAJET ? ";RE$
1770  IF  LEFT$ (RE$,1) <  > "O" THEN  PRINT : GOTO
    1830
1780  INPUT "MODIFICATION DU TRAJET NUMERO ? ";I$:I
    =  VAL (I$)
1790  INPUT "              AU TRAJET NUMERO ? ";J$:J
    =  VAL (J$)
1800  IF I > NR OR J > NR OR I = J THEN  PRINT : PR
   INT : FLASH : PRINT "     ERREUR     ": NORMAL
   : FOR T = 1 TO 1000: NEXT : GOTO 1560
1810  PRINT : PRINT "VALEUR ACTUELLE :";D(I,J)
1820  INPUT "NOUVELLE VALEUR :";D(I,J)
1830  PRINT : INPUT "UN AUTRE MODIFICATION ?";RE$
1840  IF  LEFT$ (RE$,1) = "O" THEN 1590
1850  PRINT : FLASH : PRINT  TAB( 5);"CALCUL EN COU
   RS  ": NORMAL
1860  RETURN
1870  REM ------------------------
1880  POKE 36,0: VTAB 23: PRINT "APPUYEZ SUR UNE TO
   UCHE POUR LA SUITE.": GET C$
1890  RETURN
1900  REM ------------------------
1910  REM  SAUVEGARDE SUR DISQUE
1920  REM ------------------------
1930  PRINT
1940  PRINT D$;"OPEN";NE$
1950  PRINT D$;"WRITE";NE$
1960  PRINT NR: PRINT VD$
1970  FOR I = 1 TO NR
1980  PRINT V$(I)
1990  NEXT I
2000  FOR I = 1 TO NR
2010  FOR J = 1 TO NR
2020  PRINT D(I,J)
2030  NEXT J,I
2040  PRINT D$;"CLOSE"
2050  RETURN
2060  REM ----------------------------
2070  REM  LECTURE ENREGISTREMENT
2080  REM ----------------------------
2090  VTAB 23: INPUT "NOM DE L'ANALYSE (? POUR LE C
   ATALOGUE) ?";NE$
2100  IF NE$ = "?" OR  LEN (NE$) = 0 THEN  GOSUB 23
    70: HOME : GOTO 2090
2110  HOME : INVERSE : POKE 36,8: VTAB 23: PRINT "A
   NALYSE FICHIER ...";NE$: NORMAL
2120  PRINT
2130  PRINT D$;"OPEN";NE$
2140  PRINT D$;"READ";NE$
2150  INPUT NR: INPUT VD$
2160  GOSUB 330
2170  FOR I = 1 TO NR
2180  INPUT V$(I)
2190  NEXT I
2200  FOR I = 1 TO NR
2210  FOR J = 1 TO NR
2220  INPUT D(I,J)
2230 C(I,J) = D(I,J)
2240  NEXT J,I
2250  PRINT D$;"CLOSE"
2260  RETURN
2270  REM --------------------------------
2280  REM  TRAITEMENT ERREUR --------------
2290  REM --------------------------------
2300  IF  PEEK (222) = 5 THEN  VTAB 10: PRINT "ERRE
   UR FICHIER": PRINT D$;"DELETE";NE$: GOSUB 2370:
    GOTO 2320
2310  PRINT "ERREUR": SPEED= 255
2320  FOR IT = 1 TO 1000: NEXT
2330  POKE 34,0: RUN
2340  REM --------------------------------
2350  REM  LECTURE CATALOGUE
2360  REM --------------------------------
2370  PRINT D$;"CATALOG": PRINT : PRINT "APPUYER SU
   R UNE TOUCHE": GET R$: RETURN
2380  REM ----------------------------------------
   ----
2390  REM  SP RESULTATS
2400  REM  ----------------------------------------
   -----
2410  HOME
2420  PRINT : PRINT
2430  PRINT "**************************"
2440  PRINT "* ARCS DE L'ARBORESCENCE *"
2450  PRINT "**************************"
2460  PRINT : PRINT
2470 NL = 0
2480  FOR I = 1 TO K
2490  PRINT "DEPART:"; LEFT$ (V$(X(I)),11);: POKE 3
   6,20: PRINT "ARRIVEE:"; LEFT$ (V$(Y(I)),11):NL
   = NL + 1
2500  NEXT I
2510  IF FIN = 0 THEN 2540
2520  PRINT : PRINT "LE CIRCUIT NE PEUT ETRE DETERM
   INE"
2530  PRINT "   PLUS AVANT SANS ERREUR": RETURN
2540  IF  LEFT$ (R$,1) <  > "O" THEN  GOSUB 1880
2550  HOME : PRINT : PRINT
2560  PRINT "LE PLUS COURT CHEMIN EST:"
2570  PRINT "========================="
2580  IF R <  > 2 AND R <  > 3 AND  LEFT$ (R$,1) <
    > "O" THEN  INPUT "VILLE DE DEPART ? ";VD$
2590  PRINT : PRINT "DEPART :    ": PRINT
2600  FOR I = 1 TO NR
2610  FOR J = 1 TO NR
2620  IF VD$ = V$(X(J)) THEN  PRINT V$(X(J)):VD$ =
   V$(Y(J)): GOTO 2660
2630  NEXT J
2640  PRINT : FLASH : PRINT "VILLE INCONNUE": NORMA
   L
2650  FOR T = 1 TO 1500: NEXT : GOTO 2550
2660  NEXT I
2670  PRINT VD$
2680  PRINT : PRINT "ARRIVEE"
2690  RETURN
2700  REM ----------------------------
2710  REM  LISTING IMPRIMANTE
2720  REM ----------------------------
2730  PRINT D$;"PR£3"
2740  PRINT : PRINT D$;"PR£1"
2750  PRINT  CHR$ (27); CHR$ (64): REM  INIT IMPRIM
   ANTE
2760  PRINT T$;
2770  PRINT : PRINT  CHR$ (27); CHR$ (14);: REM  GR
   OS CARACTERES
2780  PRINT "LE VOYAGEUR:ANALYSE DU FICHIER ";NE$;
2790  REM  PRINT CHR$(15): REM 132 COLONNES
2800  PRINT  CHR$ (20): REM  80 COLONNES
2810  PRINT T$: PRINT : PRINT
2820  PRINT "*************"
2830  PRINT "*LES VILLES*"
2840  PRINT "*************"
2850  FOR I = 1 TO NR
```

```
2860  PRINT I;" = ";V$(I)
2870  NEXT I
2880  PRINT : PRINT
2890  PRINT "**********************"
2900  PRINT "*MATRICE DES TRAJETS*"
2910  PRINT "**********************"
2920  PRINT : PRINT
2930  PRINT "TR:";
2940  FOR I = 1 TO NR
2950  POKE 36,I * 5 + 1: PRINT I;
2960  NEXT
2970  PRINT : PRINT  LEFT$ (T$,NR * 5 + 4)
2980  FOR I = 1 TO NR
2990  PRINT I;" !";
3000  FOR J = 1 TO NR
3010  POKE 36,J * 5: IF I = J THEN  PRINT "0";: GOT
    O 3030
3020  PRINT D(I,J);
3030  NEXT J
3040  PRINT : NEXT I
3050  PRINT  LEFT$ (T$,NR * 5 + 4)
3060  GOSUB 2420
3070  PRINT : PRINT D$;"PR£3": PRINT  CHR$ (21): PR
    INT : POKE 34,0: RUN
3080  REM  -------------
3090  REM  MODE D'EMPLOI
3100  REM  -------------
3110  HOME : SPEED= 110
3120  PRINT "Ce programme à été développé à partir
3130  PRINT "du modéle de LITTLE appelé également"
3140  PRINT "modéle du voyageur de commerce."
3150  PRINT "Il permet de déterminer le trajet"
3160  PRINT "optimal à suivre, lorsque sur un graph
    e"
3170  PRINT "on veut passer par tous les repéres un
    e"
3180  PRINT "seule fois et une seule."
3190  PRINT "Ce modéle correspond donc éxactement a
    u"
3200  PRINT "problème du voyageur de commerce qui"
3210  PRINT "doit visiter plusieurs villes en
3220  PRINT "minimisant par exemple la distance à"
3230  PRINT "parcourir."
3240  PRINT
3250  PRINT "Pour poser le problème, il faut "
3260  PRINT "déterminer la matrice des couts du
3270  PRINT "voyage de ville à ville.Ces couts"
3280  PRINT "peuvent etre exprimés en francs, en"
3290  PRINT "en litres d'essence, ou en kilométres"
3300  PRINT "Il peuvent etre différents à l'aller"
3310  PRINT "et au retour.": PRINT
3320  PRINT : PRINT "EXEMPLE:"
3330  PRINT "--------"
3340  PRINT "Un voyageur désire se rendre une fois
3350  PRINT "et un seule dans 5 villes différentes.
    "
3360  PRINT "Le but est de déterminer la tournée qu
    e"
3370  PRINT "le voyageur doit réaliser pour parcou-
    "
3380  PRINT "-rir le moins de kilométres possibles.
    "
3390  PRINT "Il faut tout d'abord établir la matric
    e"
3400  PRINT "de saisie:"
3410  PRINT : PRINT "TRAJET: 1  2  3  4  5"
3420  PRINT
3430  PRINT "     1  0  21 35 19 14"
3440  PRINT "     2  36 0  48 3
    5 42"
3450  PRINT "     3  25 34 0  2
    5 31"
3460  PRINT "     4  10 15 22 0
    13"
3470  PRINT "     5  32 24 39 2
    9 0"
3480  PRINT
3490  PRINT "Ce tableau indique
     que le nombre de"
3500  PRINT "kilometres séparan
    t la ville 1 de la"
3510  PRINT "ville 2 est de 21
    à l'allée, et de 36"
3520  PRINT "au retour etc.
3530  PRINT : PRINT : PRINT "Ce
     programme à été inspiré
    par le livre"
3540  PRINT " de JEAN PIERRE BL
    ANGER édité aux"
3550  PRINT "éditions P.S.I. MO
    DELE PRATIQUE DE"
3560  PRINT "DECISION.": PRINT
    : PRINT
3570  SPEED= 255: GOSUB 1880
3580  POKE 34,0: GOTO 90
```

Fichier TEXT de démonstration

```
5
PARIS
PARIS
MEAUX
SEZANNE
TROYES
MELUN
1E+30
44
139
107
54
47
1E+30
53
134
56
139
59
1E+30
64
93
112
139
68
1E+30
128
56
54
90
122
1E+30
```


Les reliures Pom's



60,00 F TTC Franco – Bon de commande page 74

Turbo-Pascal : Graphisme, suite...

Nicolas Martin & Michel de Crémoux

La présentation de Turbo-Pascal, de Nicolas Montsarrat dans Pom's 23, nous avait donné un avant-goût de l'utilisation du graphisme sous ce langage. Le programme *GRAPH.PAS* illustrait une solution au principal reproche adressé à Turbo-Pascal : l'absence de graphisme.

Nous avons repris à notre compte l'objectif initial, c'est-à-dire disposer d'un ensemble de procédures Pascal dédiées au graphisme HGR, en proposant d'exploiter la ROM du Basic Applesoft, écrite bien sûr en 6502, sous le CP/M Z-80 du Turbo-Pascal.

Pari gagné ! La procédure *HLINE* obtenue est beaucoup plus rapide. De plus, nous avons inclus d'autres procédures telles que *HCOLOR*, *HPOS*, *HPLOT_TO*.

Utilisation

Le fichier *DIAG.PAS* est un fichier source Turbo-Pascal qui contient la définition de l'ensemble des procédures graphiques.

Rappelons qu'il n'est pas possible de créer des *LIBRARIES* ou des *UNITS* en Turbo-Pascal ; il n'est donc possible que de créer des fichiers inclus appelés au moment de la compilation.

Le source *ESSAI.PAS* est un petit programme de démonstration qui, en utilisant les définitions de *DIAG.PAS*, obtient la même animation graphique que le programme *GRAPH.PAS* de Pom's 23.

ESSAI.PAS n'est, bien sûr, fourni qu'à titre d'illustration de la méthode. Pour le compiler :

transférer les fichiers *DIAG.PAS* et *ESSAI.PAS* de la disquette Pom's vers une disquette CP/M à l'aide de Universal File Conversion (sinon, à vos claviers!) ;

charger Turbo-Pascal (désolé, vous ne le trouverez pas sur la disquette Pom's...) ;

dans le menu des Options du compilateur, demandez une compilation vers un fichier *.COM* et à partir de l'adresse 5000 (sinon...).

La compilation de *ESSAI.PAS*, utilisant *DIAG.PAS*, produira un fichier *ESSAI.COM* directement utilisable. Les habitués du Pascal Apple apprécieront certainement la rapidité de la compilation...

Procédures graphiques

Voici la liste des procédures définies par *DIAG.PAS* :

Hge passe en mode graphique en effaçant la page graphique

Hg idem mais sans effacer

Hplot (x,y) idem qu'en Applesoft (x<280, y<192)

Hpos (x,y) place le curseur sur le point indiqué sans rien tracer

HplotTo (x,y) trace du point courant vers le point indiqué

Hline (x1,y1,x2,y2) trace une droite entre les deux points indiqués

HgrInitCercle procédure d'initialisation du tracé de cercle ; à appeler obligatoirement, mais une seule fois par programme

Cercle (cx,cy,r) trace un cercle de centre cx,cy et de rayon r

Hcolor (couleur) définit la couleur: noir, blanc, ocre, bleu, vert ou rouge

BackGround (couleur) définit la couleur du fond de l'écran

Touche attente d'une touche clavier pour continuer le programme

Reponse (rep) retourne le code ASCII de la touche frappée (*rep* est de type *CHAR*)

Text retourne au mode texte. Ne pas utiliser *Readln*

Texte idem mais efface l'écran et autorise l'emploi de *Readln*.

Tous les paramètres numériques de ces procédures sont des entiers.

Pendant l'activation du mode graphique, il est interdit d'effectuer des entrées/sorties (en raison de l'architecture mémoire de CP/M), hormis Read (Kbd,Variable) qui ne renvoie pas d'écho à l'écran ; on peut également exécuter un Readln juste avant de revenir au mode texte par Texte.

Source DIAG.PAS
(Turbo-Pascal)

```
Type

 Couleur = ( Noir,Ocre,Bleu,Blanc,Vert,Rouge ) ;

Var

 Cosinus,Sinus : Array [ 0..90 ] of Real ;

Procedure Jsr ( Adh,Adl : Byte ) ;
 Begin
  Mem [ $F3D0 ] := Adl ;          { Poids faible sous-programme }
  Mem [ $F3D1 ] := Adh ;          { Poids fort    " }
   InLine    ( $2A/$DE/$F3/       { LD  HL , $F3DE }
         $77 ) ;                  { LD (HL),  A }
 End;

Procedure Hge ;
 Begin
  Mem [ $E000 ] := 0 ;
  Mem [ $F0E6 ] := $40 ;
   Jsr ( $F3,$F2 ) ;              { Efface l'ecran du moniteur }
  Mem [ $E055 ] := 0 ;
  Mem [ $E052 ] := 0 ;
  Mem [ $E057 ] := 0 ;
  Mem [ $E050 ] := 0 ;
 End;

Procedure Hg ;
 Begin
  Mem [ $F0E6 ] := $40 ;
  Mem [ $E000 ] := 0 ;
  Mem [ $E055 ] := 0 ;
  Mem [ $E052 ] := 0 ;
  Mem [ $E057 ] := 0 ;
  Mem [ $E050 ] := 0 ;
 End;

Procedure Hplot ( x,y : Integer ) ;
 Begin
  Mem [ $F045 ] := y ;
  Mem [ $F047 ] := x Div 256 ;
  Mem [ $F046 ] := x Mod 256 ;
   Jsr ( $F4,$57 ) ;              { Hplot du basic }
 End;

Procedure Hpos ( x,y : Integer ) ;
 Begin
  Mem [ $F045 ] := y ;
  Mem [ $F047 ] := x div 256 ;
  Mem [ $F046 ] := x mod 256 ;
   Jsr ( $F4,$11 ) ;
 End;

Procedure HplotTo ( x,y : Integer ) ;
 Begin
  Mem [ $F045 ] := x Mod 256 ;
  Mem [ $F046 ] := x Div 256 ;
  Mem [ $F047 ] := y ;
   Jsr( $F5,$3A ) ;
   Jsr( $F5,$CB ) ;
 End;

Procedure Hline ( x1,y1,x2,y2 : Integer ) ;
 Begin
  Hpos ( x1,y1 ) ;
  HplotTo ( x2,y2 ) ;
 End;

Procedure HgInitCercle ;
 Var
  i : Integer ;
 Begin
  For i := 0 to 90 do
   Begin
    Cosinus [ i ] := Cos ( i/360*3.14159 ) ;
    Sinus   [ i ] := Sin ( i/360*3.14159 ) ;
   End;
 End;

Procedure Cercle ( cx,cy,r : Integer ) ;
 Var
  n,Teta,x,y : Integer ;

 Procedure Print ;
  Begin
   if ((x>-1) and (x<280) and (y>-1) and (y<192)) then Hplot ( x,y ) ;
  End;

 Procedure Ex ( Var x,y : Integer ) ;
  Var
   i : Integer ;
  Begin
   i := y ;
   y := x - cx + cy ;
   x := i - cy + cx ;
  End;

 Procedure Exm ( Var x,y : Integer ) ;
  Var
   i : Integer ;
  Begin
   i := y ;
   y := cx - x + cy ;
   x := cy - i + cx ;
  End;

Begin
 n := 1 ;
 Case r of
  1..5  : n := 20 ;
  5..10 : n := 10 ;
  10..18 : n := 6 ;
  19..28 : n := 3 ;
  29..45 : n := 2 ;
 End;
 Teta := 0 ;
 Repeat
  x := Trunc ( r * Cosinus [ teta ] + cx ) ;
  y := Trunc ( r * Sinus [ teta ] + cy ) ;
  Print ;
  Exm ( x,y ) ;
  Print ;
  x := 2 * cx - x ;
  Print ;
  Ex ( x,y ) ;
  Print ;
  y := 2 * cy - y ;
  Print ;
  Exm ( x,y ) ;
  Print ;
  x := 2 * cx - x ;
  Print ;
  Ex ( x,y ) ;
  Print ;
```

```
    Teta := Teta + n ;
  Until Teta >= 90 ;
 End;

Procedure Hcolor ( Coul : Couleur ) ;
 Var
  Valeur : Byte ;
 Begin
  Case Coul of
    Noir : Valeur := $00 ;
    Ocre : Valeur := $2A ;
    Bleu : Valeur := $55 ;
    Blanc: Valeur := $7F ;
    Vert : Valeur := $AA ;
    Rouge: Valeur := $D5 ;
  End;
  Mem [ $F0E4 ] := Valeur ;      { Couleur de trace }
End;

Procedure BackGround ( Coul : Couleur ) ;
 Var
  Valeur : Byte ;
 Begin
  Case Coul of
    Noir : Valeur := $00 ;
    Ocre : Valeur := $2A ;
    Bleu : Valeur := $55 ;
    Blanc: Valeur := $7F ;
    Vert : Valeur := $AA ;
    Rouge: Valeur := $D5 ;
  End;
  Mem [ $F045 ] := Valeur ;      { Couleur de trace }
   Jsr($F3,$F4) ;
 End;

Procedure Touche;
 Var
  Rep : Char ;
 Begin
  Read ( Kbd,Rep ) ;
 End;

Procedure Reponse ( Var Rep : Char ) ;
 Begin
  Read ( Kbd,Rep ) ;
 End;

Procedure Text ;
 Begin
  Mem [ $E001 ] := 0 ;
   Jsr ( $F3,$99 ) ;
 End;

Procedure Texte ;
 Begin
  Mem [ $E001 ] := 0 ;
   Jsr ( $F3,$99 ) ;
   ClrScr ;
 End;
```

```
][+
//e
//e+
//c
][gs
```

Source ESSAI.PAS
(Programme de démonstration Turbo-Pascal)

```
{ Ce programme contient un exemple d'utilisation de certaines   }
{ procedures graphiques. Elles ne sont pas toutes utilisees ici. }
{ La liste complete est donnee dans l'article. Il ne faut pas oublier  }
{ de le compiler avec une adresse de debut de 5000.                    }

Program Essai ;

 Var
    r,a : Real ;
  i,x,y : Integer ;
    Car : Char ;
Caractere : Char ;

{ Le fichier inclus Diag.Pas contient les instructions graphiques.  }
{ Il est possible de l'inclure directement au programme en le lisant }
{ comme un bloc. Dans ce cas, la ligne ci-dessous doit disparaitre.  }
{$IDIAG.PAS }


Begin
 ClrScr ;
 Writeln ( 'Texte avant passage en mode graphique' ) ;
 Writeln ( 'Veuillez patienter' ) ;

{ HgInitCercle n'est necessaire que si l'on a l'intention d'utiliser la }
{ la procedure Cercle, dans ce cas elle est indispensable. Son          }
{ execution prend environ une dizaine de secondes, et permet un        }
{ gain de temps dans le trace des cercles. Elle est necessaire.        }
 HgInitCercle ;

{ Hge fait passer en mode graphique, avec effacement.                  }
 Hge ;

{ Hcolor selectionne la couleur.                                       }
{ Les couleurs sont : Blanc,Noir,Ocre,Bleu,Vert et Rouge.              }
 Hcolor ( Blanc ) ;

 i := 1 ;

 Repeat ;

{ Cercle trace un cercle de centre et de rayon donne.                  }
 Cercle ( 8 * i + 10 , 5 * i + 10 , 9 ) ;

 i := i + 1 ;
 Until i=30 ;

 r:=0.0;
 a:=0;

{ Hpos positionne le curseur graphique sur l'ecran graphique.          }
 Hpos(140,95);

 Repeat
  x := Trunc ( 140 + r * Cos ( a ) ) ;
  y := Trunc ( 95 + r * Sin ( a ) ) ;

{ HplotTo trace une droite entre la position du curseur et la          }
{ position indiquee.                                                   }
  HplotTo ( x,y ) ;

  r := r + 1.5 ;
  a := a +1.553 ;
 Until r>95 ;

{ Text permet de revenir en mode texte sans effacer la                 }
{ page texte. Attention, si l'on emploie Text, il ne faut pas          }
{ employer de Readln avant Text pour attendre qu'une touche            }
{ soit frappee au clavier pour continuer, mais Reponse ou Touche       }
{ qui permettent d'attendre qu'une touche du clavier soit frappee      }
{ pour continuer l'execution du programme.                             }
```

```
{ Leur utilisation est obligatoire si l'on veut faire une pause.  }
Reponse ( Caractere ) ;
Text;
Writeln ( 'Le texte a bien ete conserve par Text' ) ;
Writeln ;
Writeln ( 'Touche tapee : ',Caractere ) ;
Writeln ;
Writeln ( 'Tapez une touche pour quitter le programme' ) ;
Touche ;

ClrScr ;

End.
```

Bibliographie

Alexandre Duback

Impression sur graphisme et mise en page sur Macintosh, par Andréas Pfeiffer, Cédic/Nathan.

Bon ouvrage qui représente un renouvellement important par rapport à la précédente édition (Applications graphiques pour Macintosh, 1985).
On garde les grands classiques : en dessin, MacPaint et ses satellites ClickArt et les planches d'images, MacDraft, MacSpace, Chart, SlideShow. Dans la mise en pages, ReadySetGo, MacEditeur et PageMaker. Et aussi MacTablet, MacVision et ThunderScan.
On trouve beaucoup de nouveautés comme : en dessin, ComicWorks, Fontastic, FullPaint, MacBillBoard, MacDraw, StatWorks (il manque toutefois les Criket). Particulièrement en dessin 3D, SpaceEdit, EZDraft, Mac3D, MacArchitrion. Et aussi Excel, JustText... Il manque en revanche 4ème dimension, ainsi que les nouveaux traitements de textes Writer + et Word 3.

La mise en page est bonne, l'auteur ayant conservé l'astucieuse formule consistant à mettre en relief ses remarques importantes sous forme de fac-similés de calepins.

En revanche, on peut lui reprocher, comme à son prédécesseur, d'être positionné un peu en porte à faux : insuffisant, évidemment, pour servir de mode d'emploi résumé et, trop classique, ne laissant guère de place à des trucs ou astuces d'utilisation.

La programmation en Basic avec ProDOS, InterÉditions, 265 pages. Traduit de l'anglais.

Il s'agit d'un livre fort bien présenté (style manuel de l'ImageWriter II, manuels de référence) dont la reliure spirale facilite la consultation.

L'ensemble des ordres ProDOS accessibles en Basic est abordé de façon didactique à l'aide de nombreux exemples. Le livre vise en priorité le nouvel acquéreur d'un Apple // démuni de documentation.

L'index bien fait permet de retrouver facilement la syntaxe oubliée, la signification de l'erreur 20 ou les adresses mémoires affectés par un IN£1000 par exemple.

L'ouvrage est accompagné d'une disquette comprenant de nombreux exemples de programmes, mais également de l'utilitaire APA qui à lui seul justifierait l'achat. APA donne de nouvelles fonctions à l'Apple // :

- numérotation automatique des lignes en Basic ;
- renumérotation ;
- fusion de programmes ;
- suppression des REMs ;
- listage des variable ;
- conversions décimal/hexa...

Excel, La pratique, par Mary Campbell, InterÉditions, 512 pages. Traduit de l'anglais.

Voici un livre comme savent le faire les américains : clair et bien illustré, mais écrit pour des lecteurs au ras des pâquerettes ; la traduction est bien faite, ce qui est assez rare pour être noté.

Nous avons trouvé dans cet ouvrage une lacune de première importance : il n'y a pas de copie d'écran avec les formules.

Cela ne nous semble pas très sérieux de former les gens à l'utilisation d'un tableur sans que les formules ne soient fréquemment présentées en regard des feuilles de calcul.

Par ailleurs, l'auteur ne semble avoir pas remarqué qu'il y avait des bogues dans Excel et, par conséquent, n'indique pas la façon de les contourner.

En conclusion, nous conseillons plutôt le recours à d'autres ouvrages d'initiation à Excel.

Le Pham Hiep

LeMouse : souris & menus

A force de faire "joujou" avec la souris, nous avons trouvé qu'il pourrait être intéressant d'écrire un petit programme "portable" et modulaire pour exploiter la souplesse de la souris. Ainsi, ce programme vous permettra de créer des menus déroulants d'une manière simple, et sera à même de donner un nouveau 'look' à vos programmes...

LEMOUSE a été écrit pour les Apple //c et utilise les instructions du 65C02, ainsi que les caractères de la souris.

Fonctionnement du programme

Ce programme donne un exemple possible du travail en Basic, avec les interruptions et dans un environnement ProDOS. Une fois cette routine lancée, l'utilisateur du programme Basic dispose d'une ligne de commandes accessible avec la souris.

Le programme est quasiment transparent et, si par 'hasard' vous perdiez l'interruption de la souris (absence du curseur), vous pourriez toujours le reconnecter en passant par "&". Ce programme est très simple, nous pouvons même envisager de placer la partie principale dans le Basic puis les tables de menus... en RAM et de créer une nouvelle commande (MOUSE par exemple).

Pour vous donner une idée de "l'ambiance" créée , vous trouverez ici une série de menus dont seules les fonctions de ProDOS et MODE sont disponibles (le reste, vous pouvez le compléter suivant votre machine... il suffit d'insérer votre routine à l'emplacement *FnMENUx* prévu).

Par le menu ProDOS : vous disposez de :

- **CATALOG**, pour faire un catalogue normal ;
- **EXECUTE** permet d'exécuter le programme dont le nom est WORKFILE ;
- **CHARGE**, charge en mémoire le programme Basic dont le nom est WORKFILE ;
- **SAUVE** sauvegarde sur disquette le programme Basic en mémoire sous le nom WORFILE.

Après 'Execute' et 'Charge', ProDOS déconnecte la souris ; vous pouvez la reconnecter en passant par "&"

Par le menu MODE : vous accèdez à :

- **VTBL ON** (vertical blanking), active les interruptions suivant le rafraîchissement de l'écran ; ce mode est 'dynamique' et permet d'avoir une gestion souple et agréable de la souris ;
- **VTBL OFF**, l'interruption ne vient que si la souris a été déplacée. Pour voir la différence avec le premier mode, il vous suffit de mettre sous mode VTBL OFF, puis cliquer sur un menu et lâchez le bouton très doucement – si vous faites cela correctement, le menu devrait rester affiché – puis faites la même chose avec l'autre mode) ;
- **MOUSE OFF** désactive le travail sous interruptions de la souris (vous pouvez revenir par "&").

Structure du programme

104-112, les noms des menus disponibles à afficher sur la ligne 0 de l'écran.
117-270, les tables des menus, dont la structure est la suivante :

- les deux premiers octets designent la marge gauche et droite de la fenêtre déroulante ;
- le troisième octet représente le nombre de fonctions (n) du menu, plus deux ;
- le quatrième octet n'est pas utilisé.
- les *n* couples d'octets suivants contiennent les adresses (faible/fort) de la *n*ième fonction de ce menu (FnMENUx) ;
- viennent ensuite les caractères de la fenêtre à afficher (vérifiez que les caractères de chaque ligne, plus les 'bornes' gauche et droite, correspondent bien au nombre de caractères MDMENU-MGMENU+1).

117-270, les cinq tables.
276-283, emplacement des fonctions du premier menu.
286-334, les fonctions de ProDOS.
335-389, emplacement des fonctions des menus suivants.
392-469, les sous-programmes.
471-567, gestion de la carte 80 colonnes (cela assure la compatibilitée entre les //c les //e 65C02 ; les possesseurs de //c pourront remplacer READCAR par PICKY et AFFCAR par COUT).
569-674, les sous-programmes.
676-706, affiche le 'marqueur' à gauche de la fonction choisie (si vous voulez avoir des caractères

en inverse, il suffit de changer :

```
CMP   #'D'
BEQ   votre routine
      d'inversion des
      caractères
JSR   <normal>
```

708-886, routine de traitement des différents cas (bouton et de souris).
888-906, sous-programmes de calcul.
910-944, partie principale du traitement des interruptions.
947-980, sous-programmes pour initialiser la "fenêtre" de la souris.
985-1042, programme principal.
1047-1069, faire un catalogue en passant par l'interpréteur Basic.

Nous pensons que cet utilitaire, du fait de sa structure modulaire, est simple et assez ouvert ; il est donc envisageable d'écrire des "menus" afin de constituer une sorte de "super boîte à outils fonctionnant avec des menus déroulants.

À vous de jouer maintenant...

• • •

Programme 'START.MOUSE'

```
10  PRINT  CHR$ (4);"-LEMOUSE3.0.C"
20  HOME : PRINT " ASSUREZ QUE VOTRE SOURI
    S EST EN PLACE "
60  PRINT : PRINT  CHR$ (4);"CATALOG"
70  GET A$
```

Programme 'WORKFILE'

Il s'agit du programme sur lequel vous travaillez habituellement. Il peut être lancé de façon directe par le startup.

Source 'LEMOUSE 3.0'

Assemblage par ProCODE

```
1   *
2   *
3   ****************************
4   *                          *
5   *       LEMOUSE 3.0        *
6   * par Le Pham HIEP 8/85    *
7   *                          *
8   ****************************
9   *
10
11          ORG  $7000
12  IRQ     =    $3FE
13
14  COUT    =    $FDED
15
16  BASL    =    $28
17
18  WNDTOP  =    $22
19
20  INVFLG  =    $32
21  *
22
23
24
25  ETATM   =    $778
26  YH      =    $5F8
27  YL      =    $4F8
28  XH      =    $578
29  XL      =    $478
30  SETMOUSE =   $12
31  SERVEMOUSE = $13        ;
32  READMOUSE =  $14
33  CLEARMOUSE = $15
34  POSMOUSE =   $16
35  CLAMPMOUSE = $17
36  HOMEMOUSE =  $18
37  INITMOUSE =  $19
38
39  *
40
41          JMP  DEBUT
42
43
44
45
46  SENSI   DFB  $03        ;précision de la souris 1-4
47  BASLM   DFB  $00,$04
48  BASLM2  DFB  $00,$00
49  MHORI   DFB  $00,$00
50  MVERTI  DFB  $00
51  MVERTI2 DFB  $00
52  OLDCAR  DFB  $A0
53
54  NFLAG   DFB  00
55  NFLAG2  DFB  00
56  MGMENU  DFB  00
57  MDMENU  DFB  00
58  NFCTN   DFB  00
59
60  XTAMP   DFB  00
61
62  *
63  MAXMENU DFB  6
64  BREAKF  DFB  00
65
66  *
67  MAXFEH  DFB  79,00      ;ici 80 col
68  MAXFEV  DFB  23         ;et 24 ligne
69  *
70  CALCTAM DFB  $00
71  TAMPON  DFB  $A0
72
73
74
75
76  INTMODE DFB  $0F        ;mode d'interruption de la
                             souris
77  N       DFB  $04        ;n du slot souris
78  CN      DFB  $C4        ;CN
79  N0      DFB  $40        ;n0
80  ****************************
81  *
82  TABMENU DA   MENU1      ;la pomme
83          DFB  1
84          DA   MENU2      ;prodos
85          DFB  5
86          DA   MENU3      ;basic
87          DFB  14
88          DA   MENU4      ;monitor
89          DFB  22
90          DA   MENU5      ;imprimant
91          DFB  32
92          DA   MENU6      ;mode
93          DFB  44
94          DA   XXXX
95          DFB  51
96
97  *
98  *
99  * ETC
100 *
101
102 ****************************************
103
104 TITRE   HEX  41
105         ASC  "   PRODOS"
```

```
106           ASC  "   BASIC"
107           ASC  "   MONITOR"
108           ASC  "   IMPRIMANT"
109           ASC  "   MODE"
110           DFB  00
111
112
113
114  ************************************
115
116
117  MENU1    DFB  00,16       ;MGMENU MDMENU
118           DFB  05,00       ;2 fctn+2
119
120           DA   F1MENU1
121           DA   F2MENU1
122           DA   F3MENU1
123
124           ASC  "(c) Le P&TAM 85"
125           HEX  5A
126           HEX  5F
127           ASC  "Calculette     "
128           HEX  5A
129           ASC  '_'
130           ASC  "Date           "
131           ASC  'Z'
132
133           ASC  'LLLLLLLLLLLLLLL'
134
135
136  *------------------------------
137
138
139  MENU2    DFB  04,16
140           DFB  06,00
141
142           DA   F1MENU2
143           DA   F2MENU2
144           DA   F3MENU2
145           DA   F4MENU2
146
147           HEX  5F
148           ASC  "Catalogue "
149           HEX  5A
150
151           HEX  5F
152           ASC  "Execute    "
153           HEX  5A
154           HEX  5F
155           ASC  "Charge     "
156           HEX  5A
157           HEX  5F
158           ASC  "Sauve      "
159           HEX  5A
160
161           ASC  'LLLLLLLLLLL'
162
163
164  *------------------------------
165
166
167  MENU3    DFB  13,24
168           DFB  04,00
169
170           DA   F1MENU3
171           DA   F2MENU3
172
173           HEX  5F
174           ASC  "RENUMBER "
175           HEX  5A
176           HEX  5F
177           ASC  "SEARCH     "
178           HEX  5A
179
180           ASC  'LLLLLLLLLL'
181
182
183  *------------------------------
184
185
186  MENU4    DFB  21,32
187           DFB  06,00
188
189           DA   F1MENU4
190           DA   F2MENU4
191           DA   F3MENU4
192           DA   F4MENU4
193
194           ASC  '_'
195           ASC  "List       "
196           ASC  'Z'
197           ASC  '_'
198           ASC  "Hexa       "
199           ASC  'Z'
200           ASC  '_'
201           ASC  "Registre "
202           ASC  'Z'
203           ASC  '_'
204           ASC  "Execute    "
205           ASC  'Z'
206           ASC  'LLLLLLLLLL'
207
208
209
210  *------------------------------
211
212
213  MENU5    DFB  31,41
214           DFB  08,00
215
216           DA   F1MENU5
217           DA   F2MENU5
218           DA   F3MENU5
219           DA   F4MENU5
220           DA   F5MENU5
221           DA   F6MENU5
222
223           ASC  '_'
224           ASC  "Listing "
225           ASC  'Z_'
226           ASC  "Text 40 "
227           ASC  'Z_'
228           ASC  "Text 80 "
229           ASC  'Z_'
230           ASC  "Hgr 1    "
231           ASC  'Z_'
232           ASC  "Hgr 2    "
233           ASC  'Z_'
234           ASC  "DHGR     "
235           ASC  'Z'
236
237           ASC  'LLLLLLLLL'
238
239
240  *--------------------------
241
242
243  MENU6    DFB  43,57
244           DFB  05,00
245
246           DA   F1MENU6     ;prodos on
247           DA   F2MENU6     ;prodos off
248           DA   F3MENU6
249
250           ASC  '_'
251           ASC  "VTBL OFF     "
252           ASC  'Z'
253           ASC  '_'
254           ASC  "VTBL ON      "
255           ASC  'Z'
256           ASC  '_'
257           ASC  "MOUSE OFF    "
258           ASC  'Z'
259
260           ASC  'LLLLLLLLLLLLL'
261
262  *-------------------------
263
264  MENU7    =    *
265
266  ************************
267  XXXX     EQU  *
268           DFB  $CC,$C5
269           DFB  00
270           DFB  00
271
272
273
274
275  ************************
276  F1MENU1  EQU  *
277           RTS
```

```
278
279 F2MENU1  EQU  *
280           RTS
281
282 F3MENU1  =    *
283           RTS
284
285
286 F1MENU2  EQU  *
287           SEI
288           LDX  #<CATALOG
289           LDY  #>CATALOG
290           STX  COMMAND+1
291           STY  COMMAND+2
292           JSR  CONT
293           CLI
294           RTS
295
296 F2MENU2  EQU  *
297           SEI
298           LDX  #<EXEC
299           LDY  #>EXEC
300           STX  COMMAND+1
301           STY  COMMAND+2
302           JSR  CONT
303           CLI
304           RTS
305 EXEC      ASC  "-WORKFILE"
306           HEX  8D00
307
308
309 F3MENU2  =    *
310           SEI
311           LDX  #<LOADW
312           LDY  #>LOADW
313           STX  COMMAND+1
314           STY  COMMAND+2
315           JSR  CONT
316           CLI
317           RTS
318 LOADW     ASC  "LOAD WORKFILE"
319           HEX  8D00
320
321
322 F4MENU2  =    *
323           SEI
324           LDX  #<SAVEW
325           LDY  #>SAVEW
326           STX  COMMAND+1
327           STY  COMMAND+2
328           JSR  CONT
329           CLI
330           RTS
331
332 SAVEW     ASC  "SAVE WORKFILE"
333           DFB  $8D,00
334
335 F1MENU3  =    *
336           RTS
337
338 F2MENU3  =    *
339           RTS
340
341
342
343 F1MENU4  =    *
344           RTS
345
346 F2MENU4  =    *
347           RTS
348
349 F3MENU4  =    *
350           RTS
351
352 F4MENU4  =    *
353           RTS
354
355
356
357 F1MENU5  =    *
358           RTS
359
360 F2MENU5  =    *
361           RTS
362
363 F3MENU5  =    *
364           RTS
365
366 F4MENU5  =    *
367           RTS
368
369 F5MENU5  =    *
370           RTS
371
372 F6MENU5  =    *
373           RTS
374
375
376 F1MENU6  =    *
377           LDA  #$07       ;l'interruption vient du mo
                               uvement de la souris
378           STA  INTMODE
379           RTS
380
381 F2MENU6  =    *
382           LDA  #$0F       ;l'interruption vient du ra
                               ffraichisement d'ecran
383           STA  INTMODE
384           RTS
385
386 F3MENU6  =    *          ;desactive la souris
387           LDA  #$01       ; souris en mode passif
388           STA  INTMODE
389           RTS
390
391
392 ****************************************
393 * rout acti Mouse
394 * input : Y setmouse,servemouse.....  A mode eventu
                               lle
395 * utl   : X Y
396 * output: rien
397 ****************************************
398
399 LEMOUSE  LDX  $C400,Y
400           STX  XXMOUSE+1
401 SMOUSE   LDX  CN
402           LDY  N0
403 XXMOUSE  JSR  $C400
404           RTS
405
406
407 *********************************
408 *rout  utl prodos via MLI
409 *entré  :A fctn code
410 *       :XY adr du parametre (faible fort)
411 *appel  : JSR LEPRODOS
412 *sorti  :cf manuel
413 *********************************
414
415 LEPRODOS STA  LEPRO1+3
416           STX  LEPRO1+4
417           STY  LEPRO1+5
418 LEPRO1   JSR  $BF00
419           DFB  00
420           DA   $FFFF
421           RTS
422
423
424 *********************************
425
426 ALLOCM   =    *
427           LDY  #SETMOUSE
428           LDA  INTMODE
429           JSR  LEMOUSE
430
431           RTS
432
433
434 ************************************
435 *
436 *TFER calcul 16 bits
437 *ftnc :(AY)/(2^sensi) => (AY)
438 *entré:AY valeur cal
439 *utl  :tout
440 ************************************
441
442 TFER      LDX  SENSI
443           STY  CALCTAM
444           CLC
445           CLD
446 TFER1     ROR
```

```
447           ROR  CALCTAM
448           DEX
449           BNE  TFER1
450           LDY  CALCTAM
451           RTS
452
453  ********************************
454  *TFERINV
455  *fctn    : AY * 2^SENSI  => AY
456  *
457  ********************************
458
459  TFERINV  LDX  SENSI
460           STY  CALCTAM
461           CLC
462           CLD
463  ]INV     ROL  CALCTAM
464           ROL
465
466           DEX
467           BNE  ]INV
468           LDY  CALCTAM
469           RTS
470
471  ******************************
472  *       SCREEN
473  *(extrait de la routine de 80 col)
474  *
475  *fctn   :prend ou met le car
476  *entré  :V=0 pr store
477  *        V=1    pick
478  *        A  car à stocker
479  *        Y  pos hori
480  *sort   :A car eventuel
481  *
482  ******************************
483  RD80VID  =    $C01F
484  YSAVE1   =    $1F
485  TXTPAGE1 =    $C054
486  TXTPAGE2 =    $C055
487  *
488  *
489  SCREEN   EQU  *
490           STY  YSAVE1     ;YSAVE1= $1F
491           PHA
492  *eviter de changer V via BIT
493           LDA  RD80VID    ;
494           BPL  ]SCRN40    ;bra si 40 col
495  *
496  *80 col
497  *
498           LDA  YSAVE1
499           LSR
500           TAY
501           BVS  ]STOR80
502  *
503  *80 col pick car
504  *
505           PHP
506           SEI             ;blocq IRQ pr screenholes
507           LDA  TXTPAGE2   ;TXTPAGE2= $C055
508           BCC  ]SCRN2
509           LDA  TXTPAGE1   ;TXTPAGE1= $C054
510  ]SCRN2   LDA  (BASL),Y   ;BASL = $28
511           TAY
512           LDA  TXTPAGE1
513           PLP
514           PLA
515           TYA
516           PHA
517           BVC  ]EXIT
518
519  ]STOR80  PLA
520           PHA
521           PHP
522           SEI
523           PHA
524           LDA  TXTPAGE2
525           BCC  ]SCRN3
526           LDA  TXTPAGE1
527
528  ]SCRN3   PLA
529           STA  (BASL),Y
530           LDA  TXTPAGE1
531           PLP
532           BVS  ]EXIT
533
534  *40 col
535  ]SCRN40  LDY  YSAVE1
536           BVS  ]STOR40
537           PLA
538           LDA  (BASL),Y
539           PHA
540           BVC  ]EXIT
541
542  ]STOR40  PLA
543           PHA
544           STA  (BASL),Y
545
546  ]EXIT    PLA
547           LDY  YSAVE1
548           RTS
549  ******************************
550  *
551  *
552  READCAR  EQU  *
553           CLV
554           JSR  SCREEN
555           RTS
556  *
557  *
558  ************************
559  AFFCAR   EQU  *          ;affiche le caractères ds A
560           PHA
561           BIT  INVFLG     ;INVFLG = $ 32
562           BMI  ]STORE2
563           EOR  #$80       ;car inversé
564  ]STORE2  BIT  SEV        ;(SEV)=$60
565           JSR  SCREEN
566           PLA
567  SEV      RTS
568
569  ********************
570  *AFCURM affi curseur saugarde le car ds OLDCAR
571  *
572
573  AFCURM   EQU  *
574           JSR  SBASL
575           JSR  NEWXYM
576           TYA
577           JSR  MBASCALC
578           LDY  MHORI
579           JSR  READCAR
580           STA  OLDCAR
581
582           LDA  #$42
583  AFCURM2  LDY  MHORI
584           JSR  AFFCAR
585           JSR  RBASL
586
587           RTS
588
589  ********************************
590  *     AFOLDCAR
591  * affiche le caractères qui se
592  * trouvait sous le curseur de la
593  * souris
594  *
595  *
596  AFOLDCAR EQU  *
597           JSR  SBASL
598           LDA  MVERTI
599           JSR  MBASCALC
600           LDA  OLDCAR
601           JSR  AFCURM2
602
603           RTS
604
605
606  ******************************
607  *
608
609  CBASLM   EQU  *
610           LDA  BASL
611           STA  BASLM
612           LDA  BASL+1
613           STA  BASLM+1
614           RTS
615
616
617  ********************************
618  *SBASL sauvegarde la position actuelle du curseur
```

```
*
SBASL    LDA  BASL
         STA  BASLM2
         LDA  BASL+1
         STA  BASLM2+1
         RTS

*********************************
*RBASL recupere laposition du curseur
*
RBASL    LDA  BASLM2
         STA  BASL
         LDA  BASLM2+1
         STA  BASL+1
         RTS

*********************************
*     NEWXYM
* calcule la nouvelle position
* du curseur de la souris
*
NEWXYM   EQU  *

         LDX  #$04

         LDA  $578,X     ;prend le poids fort
         LDY  $478,X     ; puis faible de la positio
                          n horizontal
         JSR  TFER       ; puis divise par 2^sensi
         STA  MHORI+1    ;mettez le resultat dans MH
                          ORI,+1
         STY  MHORI      ; pour AFFCURM plus tard

         LDX  #$04
         LDA  $5F8,X     ;de meme pour la position
         LDY  $4F8,X     ; vertical
         JSR  TFER
         STY  MVERTI
         RTS

*****************************************
***   remise à zéro de tous le drapeaux
*

ZERO     EQU  *          ;si //e  remplace  lda #00
         STZ  NFLAG      ;sta nflag ....
         STZ  MVERTI2
         RTS
*****************************************
*     PMENU & SCAR
* routines pour sauvergarder et charger
* les caracteres du menu à afficher
*
PMENU    LDA  $FF00,X
         RTS

SCAR     STA  $FF00,X
         RTS

*********************************
*      MARQ
* affiche le marqueur à gauchea
* du fonction choisit
*
MARQ     =    *
         JSR  SBASL
         LDX  #2

KK       LDA  MVERTI2
         BNE  ]EFMARQ
         LDX  #1
         LDA  MVERTI
]EFMARQ  JSR  MBASCALC

         LDY  MGMENU
         JSR  READCAR
         CMP  #' '
         BEQ  ]MARQ2
         CMP  #'D'
         BNE  ]MARQ1
         LDA  #' '
         BNE  ]MARQ1
]MARQ2   LDA  #'D'
]MARQ1   JSR  AFFCAR
         LDA  MVERTI
         STA  MVERTI2
         DEX
         BNE  ]EFMARQ
         JSR  RBASL
         RTS

*****************************************
*    LETTIRQ
* routines de detection et analyse  les différentes
* cas vient du click du bouton
*
LETTIRQ  EQU  *          ;on arrange  d'abord les
         LDX  #04        ; bit du registre d'etat
         LDA  $778,X     ; de la souris
         ROL
         ROL
         ROL
         AND  #3
         EOR  #3
         BEQ  BIT61      ;Bouton est encore appuiée
         DEA
         BEQ  BIT71      ;Bouton vient etre appuiée
         DEA
         BEQ  BIT70      ;Bouton vient etre relaché

BIT60    =    *          ;Bouton est encore relaché
         RTS             ; pas d'utilisation

BIT61    EQU  *          ;Bouton reste appuié
         LDA  NFLAG      ;si le menu pas affiché
         BEQ  ]FOK       ; alors fin
         LDA  MVERTI     ;si non regarde est on sur
                          la première ligne
         BEQ  ]FOK       ; si c'est le cas pas de "m
                          arqueur"
         JSR  MARQ       ;
]FOK     RTS


BIT70    EQU  *          ;Bouton vient etre relaché
         LDA  NFLAG      ;teste si un menu affiché
         BEQ  F          ; ou non
         JSR  BIT70X     ;si oui recupères l'ecran s
                          ous le menu
         LDA  MVERTI
         BEQ  F
         INA
         CMP  NFCTN
         BPL  F
         ASL             ;puis prépare le vecteur du
         TAX             ; JSR FnMENUx    en   TOTO
         JSR  PMENU
         STA  TOTO+1     ; charge le poids faible de
                           FnMENUx
         INX             ;
         JSR  PMENU
         STA  TOTO+2     ; charge le poids fort de F
                          nMENUx

TOTO     JSR  $0000      ;saut à FnMENUx
         RTS             ;puis retourne au prog. pri
                          ncipal


BIT70X   JSR  AFFMENU    ;affiche l'ancien ecran sou
                          s le menu
         JSR  ZERO       ;remet les drapeaux à zéro
F        RTS


BIT71    EQU  *          ;Bouton vient d'etre appuié

         LDA  NFLAG
         BNE  NOMENU     ;menu deja aff

]CONTI   LDA  MVERTI
         BEQ  BIP1       ;on est pas à la ligne 0 do
                          nc pas de menu à afficher
                         ;
SNOMENU  JSR  ZERO
NOMENU   RTS             ;


BIP1     =    *          ;recherche quel est le Menu
                          à afficher puis l'affiche
         LDX  #$00       ; en sauvegardant les carac
                          tères qui se trouvent
```

```
779             LDA  MHORI      ; sous le Menu
780             LDY  MAXMENU
781             INY
782  ]BOU       =    *          ;Boucle de recherche le num
                                 ero du menu (dans NFLAG)
783             DEY             ; à afficher
784             BMI  SNOMENU
785             INX
786             INX
787             INX
788             INC  NFLAG
789
790             CMP  TABMENU-1,X; A-(TABMENU-1,X)
791
792             BEQ  ]F1        ; cas particulier le curseu
                                 r pointe sur le premier
793                             ; caracteres du nom du menu
794             BPL  ]BOU       ;
795  *
796             JSR  SBASL      ;sauvegarde la position act
                                 uelle du curseur
797
798             LDA  #00
799             JSR  MBASCALC
800             LDY  MHORI
801             JSR  READCAR    ;prend le caractères sous l
                                 e curseur
802             PHA
803
804             JSR  RBASL .
805             PLA
806             CMP  #" "       ;si c'est l'espace alors pa
                                 s de menu à afficher
807             BEQ  SNOMENU
808  *
809             DEC  NFLAG      ;si non ajuste le numero du
                                 menu
810             DEX             ; et X pointeur de la table
811             DEX
812             DEX
813
814  ]F1        EQU  *
815             LDA  NFLAG
816             STA  BREAKF
817
818  ]CONTI     =    *
819  AFF0       LDY  TABMENU-3,X
820
821             LDA  TABMENU-2,X
822             BEQ  SNOMENU
823
824  ]SUITE     =    *
825             STA  SCAR+2     ;ici on charge pointeurs
826             STA  PMENU+2    ;recuperation et affichage
827             STY  SCAR+1     ; des car du menu et sous l
                                 e menu
828             STY  PMENU+1
829
830             ASL  NFLAG
831             ASL  NFLAG
832             ASL  NFLAG
833             ASL  NFLAG      ;n[ menu du premier niveau
834                             ;possibilite de creer des m
                                 enu déroulant de 2 niveaux
835             LDX  #00
836             JSR  PMENU
837             STA  MGMENU     ;on charge la marge gauche
                                 , droite ,
838             INX
839             JSR  PMENU      ;
840             STA  MDMENU
841             INX
842             JSR  PMENU      ; puis le nombre de lignes
                                 du menu                843
844             BEQ  SNOMENU    ;assure le coté droite ( ca
                                 s particulier )
845
846  ]CONTI     STA  NFCTN
847
848             ASL
849             TAX
850             STX  XTAMP      ;et maintenant index X poin
                                 te sur le premier
851                             ;  caractères à afficher
852  AFFMENU    =    *
853             JSR  SBASL
854
855             LDX  XTAMP
856             LDA  #01
857  X0         PHA
858
859             JSR  MBASCALC
860             LDY  MGMENU
861
862  X1         JSR  READCAR
863             CMP  #'D'       ;cas particulier ou le cara
                                 ctere est le marqueur
864             BNE  X2
865             LDA  #'_'       ; alors remplace par le car
                                 actere de la borne gauche
866  X2         PHA             ;sauvegarde le caractere su
                                 r l'ecran ds la pile
867             JSR  PMENU      ;charge le caracteres a aff
                                 icher
868             JSR  AFFCAR     ; on l'affiche
869
870             PLA
871             JSR  SCAR       ;sauvegarde le caractere de
                                 l'ecran a la place du menu
872
873             INX
874             INY
875             CPY  MDMENU
876             BMI  X1
877
878             PLA
879             INA
880             CMP  NFCTN      ;a-t-on affiché les lignes
881             BMI  X0
882
883             JSR  RBASL      ;si oui recupere la positio
                                 n current du curseur
884
885             RTS             ;et revient au programme pr
                                 incipal
886
887
888  ****************************
889  *  calcule l'addresse de la ligne A
890  *
891  MBASCALC PHA
892           LSR
893           AND  #$03
894           ORA  #$04
895           STA  BASL+1
896           PLA
897           AND  #$18
898           BCC  MBASC2
899           ADC  #$7F
900
901  MBASC2   STA  BASL
902           ASL
903           ASL
904           ORA  BASL
905           STA  BASL
906           RTS
907  *********************************
908
909  *********************************
910  TRAITIRQ EQU  *
911
912           PHA             ;pour les 6502
913           PHX             ; TXA PHA .....
914           PHY             ;sauvegarde des registres
915
916           CLD
917           LDY  #SERVEMOUSE;regarde si l'interuption v
                                 ient bien de
918           JSR  LEMOUSE    ;   la souris
919           BCS  ]FIN
920
921           LDY  #SETMOUSE  ;si c'est le cas
922           LDA  #$01
923           JSR  LEMOUSE    ; alors désactive l'interru
                                 ption
924
925
926           JSR  AFOLDCAR   ;affiche le caractere qui a
                                 ete sous le curseur
927
928           LDY  #READMOUSE ;a jour l'etat et position
                                 de la souris
```

```
929           JSR  LEMOUSE
930
931           JSR  LETTIRQ    ;traitement des "clicques"
932           JSR  AFCURM     ;affiche la nouvelle positi
                               on du curseur
933
934
935           CLC
936  ]FIN     LDY  #SETMOUSE  ;remet mode d'interuption
937           LDA  INTMODE
938           JSR  LEMOUSE
939
940           PLY
941           PLX
942           PLA             ;recupere  les registres a
                               la sortie
943
944           RTI
945
946
947  ***********************************
948  *
949  *      INITPOSM initialise la fenetre  de la souris
950  *  la fenetre de la souris sera calculée par
951  *  la potsition maximum horizontale (puis verticale)
952  *  en multipliant par 2^SENSI
953  *
954  INITPOSM EQU  *
955           LDA  #00
956           STA  $478
957           STA  $578
958           LDY  MAXFEH
959           LDA  MAXFEH+1
960           JSR  TFERINV
961           STA  $5F8
962           STY  $4F8
963           TXA
964           LDY  #CLAMPMOUSE
965           JSR  LEMOUSE
966  *
967           TXA
968           STA  $478
969           STA  $578
970           LDY  MAXFEV
971           TXA
972           JSR  TFERINV
973           STA  $5F8
974           STY  $4F8
975           TXA
976           INA
977           LDY  #CLAMPMOUSE
978           JSR  LEMOUSE
979
980           RTS
981  ****************************************
982  *
983  *          DEBUT du programme
984  *
985  DEBUT     EQU  *
986           JSR  $C300      ;active la 80 col
987           JSR  $FB39      ;fenetre standard
988
989           JSR  $FC58      ;HOME
990           LDA  #2
991           STA  WNDTOP     ;fixe marge haute de ecran
992
993           LDA  #00
994           JSR  MBASCALC   ;affiche la premiere ligne
                               de commande
995
996           LDY  #01
997  ]BOU     LDA  TITRE-1,Y
998           BEQ  ]FIN
999           JSR  AFFCAR
1000          INY
1001          BNE  ]BOU
1002
1003 ]FIN     LDA  #01
1004          JSR  MBASCALC
1005          LDA  #'S'       ;affiche le grand trait
1006          LDY  #79
1007 ]CON     JSR  AFFCAR
1008          DEY
1009          BPL  ]CON
1010          JSR  $FC58
1011
1012          LDA  #$0F
1013          STA  INTMODE
1014          LDA  #" "
1015          STA  OLDCAR
1016
1017
1018          LDY  #INITMOUSE ;active la souris
1019          JSR  LEMOUSE
1020
1021          JSR  INITPOSM   ;déclare la taille de la fe
                               netre à travailler
1022
1023          LDY  #CLEARMOUSE
1024          JSR  LEMOUSE
1025
1026          LDY  #SETMOUSE
1027          LDA  INTMODE
1028          JSR  LEMOUSE
1029
1030          LDX  #<TRAITIRQ
1031          LDY  #>TRAITIRQ
1032          STX  $3FE
1033          STY  $3FF
1034
1035          LDA  #$4C
1036          STA  $3F5
1037          LDA  #$00
1038          STA  $3F6
1039          LDA  #$70
1040          STA  $3F7
1041
1042          RTS
1043
1044  ************************
1045
1046
1047  CONT      EQU  *
1048  *
1049  * CATALOG SOUS PRODOS
1050  *
1051  *
1052            LDX  #00
1053  COMMAND   LDA  CATALOG,X
1054            STA  $200,X
1055            CMP  #$8D
1056            BEQ  COMFIN
1057            INX
1058            BRA  COMMAND
1059  *
1060  COMFIN    JSR  $BE03
1061            rts
1062
1063
1064  MESEROR   LDA  $BE0F
1065            JSR  $BE0C
1066            RTS
1067  *
1068  CATALOG   ASC  "CATALOG"
1069            HEX  8D
```

Récapitulation LEMOUSE3.0.C

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par BSAVE LEMOUSE3.0.C, A$7000,L$601

```
7000- 4C 75 75 03 00 04 00 00
7008- 00 00 00 00 A0 00 00 00
7010- 00 00 00 06 00 4F 00 17
7018- 00 A0 0F 04 C4 40 63 70
7020- 01 AD 70 05 F5 70 0E 1E
7028- 71 16 61 71 20 B7 71 2C
7030- F9 71 33 41 A0 A0 A0 D0
7038- D2 CF C4 CF D3 A0 A0 A0
7040- C2 C1 D3 C9 C3 A0 A0 A0
7048- CD CF CE C9 D4 CF D2 A0
7050- A0 A0 C9 CD D0 D2 C9 CD
7058- C1 CE D4 A0 A0 A0 CD CF
7060- C4 C5 00 00 10 05 00 FD
7068- 71 FE 71 FF 71 A8 E3 A9
7070- A0 CC E5 A0 D0 A6 D4 C1
7078- CD A0 B8 B5 5A 5F C3 E1
7080- EC E3 F5 EC E5 F4 F4 E5
7088- A0 A0 A0 A0 5A 5F C4 E1
7090- F4 E5 A0 A0 A0 A0 A0 A0
7098- A0 A0 A0 A0 5A 4C 4C 4C
70A0- 4C 4C 4C 4C 4C 4C 4C 4C
70A8- 4C 4C 4C 4C 4C 04 10 06
70B0- 00 00 72 10 72 2B 72 4A
70B8- 72 5F C3 E1 F4 E1 EC EF
70C0- E7 F5 E5 A0 5A 5F C5 F8
70C8- E5 E3 F5 F4 E5 A0 A0 A0
70D0- 5A 5F C3 E8 E1 F2 E7 E5
70D8- A0 A0 A0 A0 5A 5F D3 E1
70E0- F5 F6 E5 A0 A0 A0 A0 A0
70E8- 5A 4C 4C 4C 4C 4C 4C 4C
70F0- 4C 4C 4C 4C 4C 0D 18 04
70F8- 00 69 72 6A 72 5F D2 C5
7100- CE D5 CD C2 C5 D2 A0 5A
7108- 5F D3 C5 C1 D2 C3 C8 A0
7110- A0 A0 5A 4C 4C 4C 4C 4C
7118- 4C 4C 4C 4C 4C 4C 15 20
7120- 06 00 6B 72 6C 72 6D 72
7128- 6E 72 5F CC E9 F3 F4 A0
7130- A0 A0 A0 A0 5A 5F C8 E5
7138- F8 E1 A0 A0 A0 A0 A0 5A
7140- 5F D2 E5 E7 E9 F3 F4 F2
7148- E5 A0 5A 5F C5 F8 E5 E3
7150- F5 F4 E5 A0 A0 5A 4C 4C
7158- 4C 4C 4C 4C 4C 4C 4C 4C
7160- 4C 1F 29 08 00 6F 72 70
7168- 72 71 72 72 72 73 72 74
7170- 72 5F CC E9 F3 F4 E9 EE
7178- E7 A0 5A 5F D4 E5 F8 F4
7180- A0 B4 B0 A0 5A 5F D4 E5
7188- F8 F4 A0 B8 B0 A0 5A 5F
7190- C8 E7 F2 A0 B1 A0 A0 A0
7198- 5A 5F C8 E7 F2 A0 B2 A0
71A0- A0 A0 5A 5F C4 C8 C7 D2
71A8- A0 A0 A0 A0 5A 4C 4C 4C
71B0- 4C 4C 4C 4C 4C 4C 4C 2B
71B8- 39 05 00 75 72 7B 72 81
71C0- 72 5F D6 D4 C2 CC A0 CF
71C8- C6 C6 A0 A0 A0 A0 5A 5F
71D0- D6 D4 C2 CC A0 CF CE A0
71D8- A0 A0 A0 A0 5A 5F CD CF
71E0- D5 D3 C5 A0 CF C6 C6 A0
71E8- A0 A0 5A 4C 4C 4C 4C 4C
71F0- 4C 4C 4C 4C 4C 4C 4C 4C
71F8- 4C CC C5 00 00 60 60 60
7200- 78 A2 F9 A0 75 8E E2 75
7208- 8C E3 75 20 DF 75 58 60
7210- 78 A2 20 A0 72 8E E2 75
7218- 8C E3 75 20 DF 75 58 60
7220- AD D7 CF D2 CB C6 C9 CC
7228- C5 8D 00 78 A2 3B A0 72
7230- 8E E2 75 8C E3 75 20 DF
7238- 75 58 60 CC CF C1 C4 A0
7240- D7 CF D2 CB C6 C9 CC C5
7248- 8D 00 78 A2 5A A0 72 8E
7250- E2 75 8C E3 75 20 DF 75
7258- 58 60 D3 C1 D6 C5 A0 D7
7260- CF D2 CB C6 C9 CC C5 8D
7268- 00 60 60 60 60 60 60 60
7270- 60 60 60 60 60 A9 07 8D
7278- 1A 70 60 A9 0F 8D 1A 70
7280- 60 A9 01 8D 1A 70 60 BE
7288- 00 C4 8E 94 72 AE 1C 70
```

Suite de la récapitulation page 73...

Micro-Informations

Jean-Michel Gourévitch

Dans un univers micro-informatique bouleversé par les annonces et les nouveautés d'IBM, Apple est curieusement devenu un symbole de stabilité.

Résultat : la clientèle se transforme. Ainsi, en France, plus de 20% des Macintosh sont actuellement vendus à des sociétés figurant parmi les 100 premières entreprises du pays. Et ce changement de profil des utilisateurs ne manquera pas de se répercuter sur les futurs produits de la firme de Cupertino. L'interchangeabilité des fichiers avec le monde IBM étant devenue une priorité, Apple sortira dès septembre un lecteur (Apple 5'25 PC) capable de lire des disquettes IBM, ainsi qu'un système de récupération sur le Mac des fichiers de l'IBM, baptisé Apple File Exchange.

Les autres annonces à venir d'Apple concernent des logiciels de télécommunication ainsi qu'un programme baptisé **Wild Card** et écrit par Bill Atkinson, l'auteur de MacPaint, qui serait à la fois un générateur d'application et un instrument d'intelligence artificielle. Du côté des imprimantes, on étudie à Cupertino un modèle couleurs ainsi qu'une LaserWriter de bas de gamme (ne comportant pas le langage PostScript). À propos de PostScript, il faut signaler la sortie d'un livre en français sur ce langage de programmation particulier. L'ouvrage de 164 pages distribué par BIP est divisé en deux parties : une introduction à PostScript et un recueil de programmes. Livre seul : 285,00 F. Livre et disquette d'accompagnement : 550,00 F. **LaserPaint** de **DOS Ltd**, aussi distribué par BIP, est un logiciel de pilotage pour la LaserWriter Apple et les photocomposeuses PostScript. 6 300,00 F, en français.

Le logiciel de lecture optique automatique et reconnaissance de caractères dactylographiés **READSTAR**][+ d'**INOVATIC** est maintenant disponible pour Macintosh Plus. 43 000,00 F H.T. avec un scanner Microtek 300A.

Dans son dernier numéro, Pom's annonçait la création de lecteurs employant des disquettes de 10 ou 20 Mégas. Ils n'auront pas mis longtemps à déferler sur le marché. **Jasmine**, qui fabriquait déjà des disques durs parmi les moins chers du marché, propose désormais un lecteur employant des disquettes de 10 Mégas de format 5'1/4 (comme les anciennes disquettes de l'Apple][, mais conditionnées dans un étui plastique analogue aux disquettes 3,5 pouces du Mac). Prix pour le "Mega Drive" : 999 dollars et, pour chaque disquette : 40 dollars.

Cette même firme propose un très ingénieux disque dur de 40 mégas, le **MacBack**, qui se fixe simplement au dos du Macintosh. Prix : 1300 dollars.

Pour les logiciels, c'est à la rentrée, et particulièrement pour l'Apple Expo d'octobre, qu'il faut s'attendre à des nouveautés. En attendant, voici des rééditions et des V.F.

Nouvelles versions

Une nouvelle version du tableur **Trapeze** (distribué en France par **Alpha Systèmes**). La 2.0 intègre en plus des possibilités de se construire son tableur 'en kit', un grapheur sophistiqué, qui mettent encore davantage ce tableur à part, et le rapprochent d'un programme de mise en page.

Une nouvelle édition de **Sidekick**, l'ensemble d'accessoires de bureau de **Borland**, contient plusieurs accessoires de bureau s'installant dans le menu , et ajoute notamment un mini-tableur, un mini-traitement d'idées, aux accessoires précédents.

Nouvelle version aussi, la 2.1 d'**OverVUE**, 'Gestionnaire Relationnel Multifichier' francisée et distribué par **Soulat Frères S.A.**, plus rapide que la version précédente.

Publication électronique

C'est toujours le 'must'. **Target Software** met la dernière main à **Scoop**, un logiciel de PAO permettant d'entourer le texte autour d'objets irréguliers, de remplir une image de textes, de pratiquer un crénage fractionnel entre les lettres, de réduire les espaces entre les lignes ou entre les mots, d'allonger un caractère, de faire pivoter une image, etc.

Letraset se prépare à sortir la version 3.1 (encore améliorée!) de **Ready Set Go, Mindscape** présente **Graphics Works 1.1**, un outil permettant de peindre, dessiner, écrire, éditer des documents professionnels. **Italsoft** vend en France **Ragtime**, un traitement de textes évolué, avec un tableur (très complet) et des fonctions de micro-édition.

Microsoft a sorti avant les vacances, en France et dans notre langue, **Works** son logiciel intégré (traitement de textes, gestion de fichiers, tableur) ainsi que **Word 3.0**, la toute nouvelle version du célèbre traitement de textes comprenant césure automatique et correcteur orthographique intégré, qui en font un des plus puissants outils du marché, en concurrence directe avec **Writer Plus**, sur lequel il a l'avantage d'une parfaite fiabilité.

Éducation

Il manquait au Macintosh un système auteur. C'est une omission réparée avec **Dr Léo** d'**Apigraph**, qui permet aux enseignants de produire eux mêmes des didacticiels adaptés à leurs élèves, reposant notamment sur l'utilisation d'un livre interactif, de livres annexes, et d'un cahier de l'élève.

Matériel

Du côté des matériels, **P-Ingénierie** vend un boîtier miracle baptisé **Multi Talk**. Ce boîtier permet de bancher et de rendre partageable sous le réseau Apple Talk des périphériques qui n'étaient pas étudiés pour cela. Ainsi, plusieurs utilisateurs nantis chacun d'un Mac peuvent désormais se partager une ImageWriter, ou un modem.

Puisque la vogue de la publication électronique ne faiblit pas, **International Computer** propose des écrans pleine page parmi les moins onéreux et s'installant par 'clipage' et donc sans dommage au Mac : **Dimension 15** (vertical A4 résolution 720 x 900 : 17 671 F), **Dimension 17** (vertical A4 résolution 768 x 1024: 23 600 F) et **Dimension 19** (horizontal A3 résolution 1024 x 915 : 24 800 F).

Transformer son Mac en puissante station de travail ? C'est possible avec **Workstation** vendu par **Abvent**. Pour 25 à 50 000,00 Francs, ces cartes ajoutent un processeur Motorola 68020 tournant à 12 ou 16 mégahertz et un coprocesseur mathématique

Autre périphérique intéressant : **l'Organizer II** de **Psion**, vendu par **KA**. Il s'agit d'un véritable micro-ordinateur miniature de la taille d'une grosse calculette, disposant d'un langage de programmation, permettant de saisir sur le terrain des informations, puis de les transmettre et de les récupérer sur le Macintosh. Un logiciel **File Pak** permet de créer des fichiers et de les manipuler, sans connaître la programmation.

Quant à **Thunderscan**, le numériseur s'installant dans l'imprimante ImageWriter à la place du ruban, il a été sérieusement perfectionné et permet notamment d'entourer les images dans des formes variés, de réaliser des dégradés, de sauvegarder les images en Postscript ou en TIFF, ce qui le rend parfaitement compatible avec PageMaker. À remarquer que ce même Thunderscan est désormais aussi disponible pour l'Apple // et le IIGS.

Ada sur le Mac

Alsys, créateur du langage **Ada**, vient de signer un accord avec Apple Computer pour le développement d'un compilateur Ada destiné au Macintosh II. Rappelons que ce langage, développé principalement par Jean D. Ichbiah, président d'Alsys, a reçu l'agrément du gouvernement américain et a aussi été adopté par la NASA.

L'Apple // ?

Les programmes destinés au GS de **Version Soft** (et particulièrement **GSWrite**) ont connu quelques problèmes de mise au point, épuisant parfois la patience des utilisateurs...

Pour l'Apple //, voici un nouveau disque dur de... 320 mégas, le **Data Exchange de CMS Enhancements**. Une nouveauté importante permettant de se relier également au Mac et de partager des dossiers entre les deux familles Apple.

On peut en ce moment trouver, chez les revendeurs Apple, une plaquette publicitaire donnant une liste de quelques programmes qui 'tournent' sur le GS :

- Graphique
  - GS/Paint
  - VS/Draw
  - GraphicWriter
  - De Luxe Paint
  - Slide Show
  - Chairman
  - Visualizer
  - Démo QuickDraw
- Son
  - Digitalisation sonore
  - Synthé
  - Music Construction Set
  - Démo QuickDraw
- Éducation & Productivité personnelle
  - AppleWorks
  - GS/Paint
  - GS/Write
  - VS/Com
  - VS/Draw
  - GraphicWriter
  - Visualizer
  - De Luxe Paint
- Education & Loisirs
  - Géographie
  - Logo
  - Color Mind
  - Digitalisation sonore
  - Synthé
  - Slide Show
  - Music construction set

- Démo QuickDraw
- Chairman

- Utilitaires & Outils de développement
  - MouseDesk
  - Chooser
  - Memsoft
  - Basic AppleSoft
  - Digitalisation sonore
  - Logo
  - Slide Show
- Gestion
  - Sénior
  - Memsoft Comptabilité
  - Memsoft Ventes
  - Memsoft Paye
  - Memsoft Microbase

Jeux

Sur le front des jeux, voici un nouveau simulateur de vol, **Falcon** de **Spectrume Holobyte**, qui représente le poste de commande d'un F16, avec son viseur tête haute, ses bruits de réacteurs numérisés, la détection radar, etc. Le même éditeur propose un simulateur de vedette lance torpille baptisé **PT 109** (du nom de la vedette où s'illustra pendant la guerre le défunt président Kennedy).

On termine ces micro-informations dans le luxe avec un sac de transport pour Macintosh baptisé **MACROCO**. L'objet, fabriqué en série limitée, est en cuir 1er choix et crocodile. 8 302,00 F (quand même !), chez **Kangourou services**.

Adresses

Jasmine Technologies
555 de Haro St San Francisco
CA 94107

Alpha Systèmes
18, avenue d'Alsace Lorraine
38000 Grenoble
☎ 76 43 19 97

P-Ingénierie
226, bd Raspail
75014 Paris
☎ 43 21 93 36

International Computer
26, rue du Renard
75004 Paris
☎ 42 72 26 26

KA
14, rue Magellan
75008 Paris
☎ 47 23 72 00

Borland
Les Postillons des Bruyères
65, rue de la Garenne
92318 Sèvres

Mindscape
P.O. Box 1167
Northbrook IL 60065

Italsoft
114, bd Blanqui
75013 Paris
☎ 47 07 35 65

CMS Enhancements
3080 A Airway Av.
Costa Mesa CA USA

Apigraph
32 bis, quai JB Simon
69270 Fontaines s/ Saône

Abvent
53, avenue de Breteuil
75007 Paris
☎ 47 34 43 98

BIP
13, rue Duc
75018 Paris
☎ 42 55 44 63

Alsys
29, avenue de Versailles
78170 La Celle St Cloud

INOVATIC
Immeuble Les Quadrants
3, avenue du Centre
78180 Montigny le Bretonneux
☎ 30 57 22 11

Soulat Frères S.A.
Département micro-informatique
45-47, rue de la Division Leclerc
94259 Gentilly
☎ 47 40 00 20

Kangourou services
B.P. 19
54130 Saint Max Cedex
☎ 83 21 25 33

...suite de la page 70

```
7290- AC 1D 70 20 00 C4 60 8D
7298- A3 72 8E A4 72 8C A5 72
72A0- 20 00 BF 00 FF FF 60 A0
72A8- 12 AD 1A 70 20 87 72 60
72B0- AE 03 70 8C 18 70 18 D8
72B8- 6A 6E 18 70 CA D0 F9 AC
72C0- 18 70 60 AE 03 70 8C 18
72C8- 70 18 D8 2E 18 70 2A CA
72D0- D0 F9 AC 18 70 60 84 1F
72D8- 48 AD 1F C0 10 32 A5 1F
72E0- 4A A8 70 16 08 78 AD 55
72E8- C0 90 03 AD 54 C0 B1 28
72F0- A8 AD 54 C0 28 68 98 48
72F8- 50 24 68 48 08 78 48 AD
7300- 55 C0 90 03 AD 54 C0 68
7308- 91 28 AD 54 C0 28 70 0E
7310- A4 1F 70 06 68 B1 28 48
7318- 50 04 68 48 91 28 68 A4
7320- 1F 60 B8 20 D6 72 60 48
7328- 24 32 30 02 49 80 2C 35
7330- 73 20 D6 72 68 60 20 70
7338- 73 20 86 73 98 20 F7 74
7340- AC 08 70 20 22 73 8D 0C
7348- 70 A9 42 AC 08 70 20 27
7350- 73 20 7B 73 60 20 70 73
7358- AD 0A 70 20 F7 74 AD 0C
7360- 70 20 4B 73 60 A5 28 8D
7368- 04 70 A5 29 8D 05 70 60
7370- A5 28 8D 06 70 A5 29 8D
7378- 07 70 60 AD 06 70 85 28
7380- AD 07 70 85 29 60 A2 04
7388- BD 78 05 BC 78 04 20 B0
7390- 72 8D 09 70 8C 08 70 A2
7398- 04 BD F8 05 BC F8 04 20
73A0- B0 72 8C 0A 70 60 9C 0D
73A8- 70 9C 0B 70 60 BD 00 FF
73B0- 60 9D 00 FF 60 20 70 73
73B8- A2 02 AD 0B 70 D0 05 A2
73C0- 01 AD 0A 70 20 F7 74 AC
73C8- 0F 70 20 22 73 C9 5F F0
73D0- 08 C9 44 D0 06 A9 5F D0
73D8- 02 A9 44 20 27 73 AD 0A
73E0- 70 8D 0B 70 CA D0 DD 20
73E8- 7B 73 60 A2 04 BD 78 07
73F0- 2A 2A 2A 29 03 49 03 F0
73F8- 07 3A F0 3F 3A F0 0F 60
7400- AD 0D 70 F0 08 AD 0A 70
7408- F0 03 20 B5 73 60 AD 0D
7410- 70 F0 27 20 34 74 AD 0A
7418- 70 F0 1F 1A CD 11 70 10
7420- 19 0A AA 20 AD 73 8D 31
7428- 74 E8 20 AD 73 8D 32 74
7430- 20 00 00 60 20 C2 74 20
7438- A6 73 60 AD 0D 70 D0 08
7440- AD 0A 70 F0 04 20 A6 73
7448- 60 A2 00 AD 08 70 AC 13
7450- 70 C8 88 30 F0 E8 E8 E8
7458- EE 0D 70 DD 1D 70 F0 1F
7460- 10 F0 20 70 73 A9 00 20
7468- F7 74 AC 08 70 20 22 73
7470- 48 20 7B 73 68 C9 A0 F0
7478- CC CE 0D 70 CA CA CA AD
7480- 0D 70 8D 14 70 BC 1B 70
7488- BD 1C 70 F0 B8 8D B3 73
7490- 8D AF 73 8C B2 73 8C AE
7498- 73 0E 0D 70 0E 0D 70 0E
74A0- 0D 70 0E 0D 70 A2 00 20
74A8- AD 73 8D 0F 70 E8 20 AD
74B0- 73 8D 10 70 E8 20 AD 73
74B8- F0 8B 8D 11 70 0A AA 8E
74C0- 12 70 20 70 73 AE 12 70
74C8- A9 01 48 20 F7 74 AC 0F
74D0- 70 20 22 73 C9 44 D0 02
74D8- A9 5F 48 20 AD 73 20 27
74E0- 73 68 20 B1 73 E8 C8 CC
74E8- 10 70 30 E5 68 1A CD 11
74F0- 70 30 D7 20 7B 73 60 48
74F8- 4A 29 03 09 04 85 29 68
7500- 29 18 90 02 69 7F 85 28
7508- 0A 0A 05 28 85 28 60 48
7510- DA 5A D8 A0 13 20 87 72
7518- B0 16 A0 12 A9 01 20 87
7520- 72 20 55 73 A0 14 20 87
7528- 72 20 EB 73 20 36 73 18
7530- A0 12 AD 1A 70 20 87 72
7538- 7A FA 68 40 A9 00 8D 78
7540- 04 8D 78 05 AC 15 70 AD
7548- 16 70 20 C3 72 8D F8 05
7550- 8C F8 04 8A A0 17 20 87
7558- 72 8A 8D 78 04 8D 78 05
7560- AC 17 70 8A 20 C3 72 8D
7568- F8 05 8C F8 04 8A 1A A0
7570- 17 20 87 72 60 20 00 C3
7578- 20 39 FB 20 58 FC A9 02
7580- 85 22 A9 00 20 F7 74 A0
7588- 01 B9 32 70 F0 06 20 27
7590- 73 C8 D0 F5 A9 01 20 F7
7598- 74 A9 53 A0 4F 20 27 73
75A0- 88 10 FA 20 58 FC A9 0F
75A8- 8D 1A 70 A9 A0 8D 0C 70
75B0- A0 19 20 87 72 20 3C 75
75B8- A0 15 20 87 72 A0 12 AD
75C0- 1A 70 20 87 72 A2 0F A0
75C8- 75 8E FE 03 8C FF 03 A9
75D0- 4C 8D F5 03 A9 00 8D F6
75D8- 03 A9 70 8D F7 03 60 A2
75E0- 00 BD F9 75 9D 00 02 C9
75E8- 8D F0 03 E8 80 F3 20 03
75F0- BE 60 AD 0F BE 20 0C BE
75F8- 60 C3 C1 D4 C1 CC CF C7
7600- 8D
```

Disquettes Apple][

CLV_Pom's - 140Ko ou 800Ko	200,00 F ☐
Éditeur Vidéotex - 140Ko	200,00 F ☐
E.P.E. 5.1 - 140Ko	200,00 F ☐
E.P.E. 5.1 - 800Ko	200,00 F ☐
Max (moniteur étendu) - 140Ko	150,00 F ☐
BananaSoft - 140Ko	200,00 F ☐
Pascal - 140Ko	80,00 F ☐
Max (moniteur étendu) - 140Ko	150,00 F ☐
Dominos - 140Ko	80,00 F ☐
COGO - 140Ko	200,00 F ☐
Ludologic - 140Ko	80,00 F ☐
Ordico - 140Ko	200,00 F ☐

Recueils Pom's

Numéro 1 (Revues 1 à 4)	140,00 F ☐
Disquettes 1 à 4	200,00 F ☐
Numéro 2 (Revues 5 à 8)	140,00 F ☐
Disquettes 5 à 8	200,00 F ☐
Numéro 3 (Revues 9 à 12)	140,00 F ☐
Disquettes 9 à 12	200,00 F ☐

_____ **reliures toilées**
pour 6 numéros, soit un an 60,00 F ☐

Apple][800Ko 3'5 numéro **29**	80,00 F ☐
Apple][800Ko 3'5 numéro **30**	80,00 F ☐
Apple][800Ko 3'5 numéro **31**	80,00 F ☐

Macintosh

CLV_Pom's	200,00 F ☐
Mac Raccourci	200,00 F ☐
MacAstuces	200,00 F ☐
Mac 'A'	80,00 F ☐
Mac 'B'	80,00 F ☐
Mac 'C'	80,00 F ☐
Mac 'D'	80,00 F ☐
Mac 'E'	80,00 F ☐
Mac 'F'	80,00 F ☐
Mac 'G'	80,00 F ☐
Mac 'H'	80,00 F ☐
Mac 'I'	80,00 F ☐
Mac 14/15/16	150,00 F ☐
Mac 17	80,00 F ☐
Mac 18	80,00 F ☐
Mac 19	80,00 F ☐
Mac 20	80,00 F ☐
Mac 21	80,00 F ☐
Mac 22	80,00 F ☐
Mac 23	80,00 F ☐
Mac 24	80,00 F ☐
Mac 25	80,00 F ☐
Mac 26	80,00 F ☐
Mac 27	80,00 F ☐
Mac 28	80,00 F ☐
Mac 29	80,00 F ☐
Mac 30	80,00 F ☐
Mac 31	80,00 F ☐

Accompagnement

A][140Ko 3	60,00 F ☐
A][140Ko 4	60,00 F ☐
A][140Ko 5	60,00 F ☐
A][140Ko 6	60,00 F ☐
A][140Ko 7	60,00 F ☐
A][140Ko 8	60,00 F ☐
A][140Ko 9	60,00 F ☐
A][140Ko 10	60,00 F ☐
A][140Ko 11	60,00 F ☐
A][140Ko 12	60,00 F ☐
A][140Ko 13	60,00 F ☐
A][140Ko 14	60,00 F ☐
A][140Ko 15	60,00 F ☐
A][140Ko 16	60,00 F ☐
A][140Ko 17	60,00 F ☐
A][140Ko 18	60,00 F ☐
A][140Ko 19	60,00 F ☐
A][140Ko 20	60,00 F ☐
A][140Ko 21	60,00 F ☐
A][140Ko 22	60,00 F ☐
A][140Ko 23	60,00 F ☐
A][140Ko 24	60,00 F ☐
A][140Ko 25	60,00 F ☐
A][140Ko 26	60,00 F ☐
A][140Ko 27	60,00 F ☐
A][140Ko 28	60,00 F ☐
A][140Ko 29	60,00 F ☐
A][140Ko 30	60,00 F ☐
A][140Ko 31	60,00 F ☐

Revue n° 7	35,00 F ☐	**Revue n° 8**	35,00 F ☐	**Revue n° 10**	40,00 F ☐
Revue n° 11	40,00 F ☐	**Revue n° 12**	40,00 F ☐	**Revue n° 13**	40,00 F ☐
Revue n° 14	40,00 F ☐	**Revue n° 15**	40,00 F ☐	**Revue n° 16**	40,00 F ☐
Revue n° 17	40,00 F ☐	**Revue n° 18**	40,00 F ☐	**Revue n° 19**	40,00 F ☐
Revue n° 20	40,00 F ☐	**Revue n° 21**	40,00 F ☐	**Revue n° 22**	40,00 F ☐
Revue n° 23	40,00 F ☐	**Revue n° 24**	40,00 F ☐	**Revue n° 25**	40,00 F ☐
Revue n° 26	40,00 F ☐	**Revue n° 27**	45,00 F ☐	**Revue n° 28**	45,00 F ☐
Revue n° 29	45,00 F ☐	**Revue n° 30**	45,00 F ☐	**Revue n° 31**	45,00 F ☐

Abonnements pour six numéros à partir du _____, à :

la **revue** seule	225,00 F ☐
la **revue** et les **disquettes Apple][** 140Ko – 5' 1/4	525,00 F ☐
la **revue** et les **disquettes Apple][** 800Ko – 3' 1/2	625,00 F ☐
la **revue** et les **disquettes Macintosh**	625,00 F ☐
la **revue**, les **disquettes Apple][** 140Ko – 5' 1/4 et les **disquettes Mac**	925,00 F ☐
la **revue**, les **disquettes Apple][** 800Ko – 3' 1/2 et les **disquettes Mac**	1025,00 F ☐

* pour les envois par avion, ajoutez 15 F par numéro et/ou par disquette soit, par exemple, 90 F pour un abonnement avec ou sans disquettes.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par : Carte Bleue/VISA ☐ ➘ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ date d'expiration _ _ _ _ _ _

Montant _ _ _ _ , _ _ F Signature :

Câble-interface de communication Apple/Minitel

Cette liaison – décrite dans les numéros 27 et 28 de Pom's – est indispensable pour faire fonctionner les programmes suivants :

- **MinBas** pour Apple][+, //e, //e+, //c et //GS* : programme permettant l'enregistrement des écrans Minitel, la restitution à loisir hors réseau, le stockage et/ou l'impression de copies d'écran du Minitel, et aussi l'envoide textes ou messages sur un serveur, ces messages étant préparés à l'avance avec n'importe quel programme de traitement de textes. Programme publié dans le numéro 27 de Pom's.
- **Minitel/1** pour Macintosh : programme identique à MinBas pour Apple][, avec en plus un mini-éditeur de texte pour la préparation et le stockage des messages sans sortir du programme. Programme publié dansle numéro 27 de Pom's.
- **InterPom's** pour Apple][+, //e, //e+, //c et //GS* : programme de téléchargement entre Apple][et/ou Apple][et Macintosh. Ce logiciel autorise la transmission de n'importe quel type de fichier (système, texte, binaire, Basic...) en utilisant le Modem du Minitel. Programme publié dans le numéro 28 de Pom's.
- **InterPom's** pour Macintosh : même programme, mais version Macintosh. Programme publié dans le numéro 28 de Pom's.
- **TPom's** pour Apple][+, //e, //e+, //c et //GS** : programme de récupération de l'annuaire téléphonique sous la forme de fichiers texte. Voir page 53 du numéro 30.
- **TPom's** pour Macintosh : Identique à la version Apple][. Voir page 42 du numéro 30.

* ce programme est prévu pour fonctionner avec une carte Super Série Apple ; le port série intégré de l'Apple IIGS ne convient pas. Toutefois, si vous désirez utiliser le port intégré afin d'y connecter le Minitel pour d'autres applications, commandez un câble pour Macintosh Plus.

** sur un Apple IIGS, TPom's fonctionne indifféremment avec le port série intégré ou la carte Super Série Apple. Pour connecter le port intégré du IIGS, utilisez un câble pour Macintosh Plus.

Je désire recevoir :

câble Minitel/Apple][+, //e, //e+, IIGS avec SSC	____	à	225,00 F	_____
câble Minitel/Apple //c	____	à	225,00 F	_____
câble Minitel/Macintosh 128 & 512Ko	____	à	225,00 F	_____
câble Minitel/Macintosh Plus, IIGS port intégré	____	à	225,00 F	_____
câble Minitel/IBM PC™	____	à	225,00 F	_____
câble de liaison locale Apple II/Mac/IBM™ *	____	à	225,00 F	_____

*(préciser le type des deux machines à relier : Mac 512, Mac Plus, Apple //e //c, IIGS, IBM PC™)

* pour les envois par avion, ajoutez 15 F par câble.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par : Carte Bleue/VISA ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ date d'expiration _ _ _ _ _ _

Montant _ _ _ _ , _ _ F Signature :

Programme de transmission InterPom's

Ce programme décrit pages 53 et 54 vous donne la possibilité d'échanger à distance via Minitel et en local des fichiers de texte entre Apple //, Macintosh et IBM PC™. Entre deux machines de même type, tous les fichiers sont transférables : Applications, polices de caractères, accessoires de bureau, commandes, documents graphiques etc.

Je désire recevoir :

InterPom's pour Apple //, disquette 140Ko, 5'25	____	à	450,00 F	_____
InterPom's pour Apple //, disquette 800Ko, 3'25	____	à	450,00 F	_____
InterPom's pour Macintosh, disquette 3'25	____	à	450,00 F	_____
InterPom's pour IBM PC™	____	à	450,00 F	_____

* pour les envois par avion, ajoutez 15 F par disquette.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

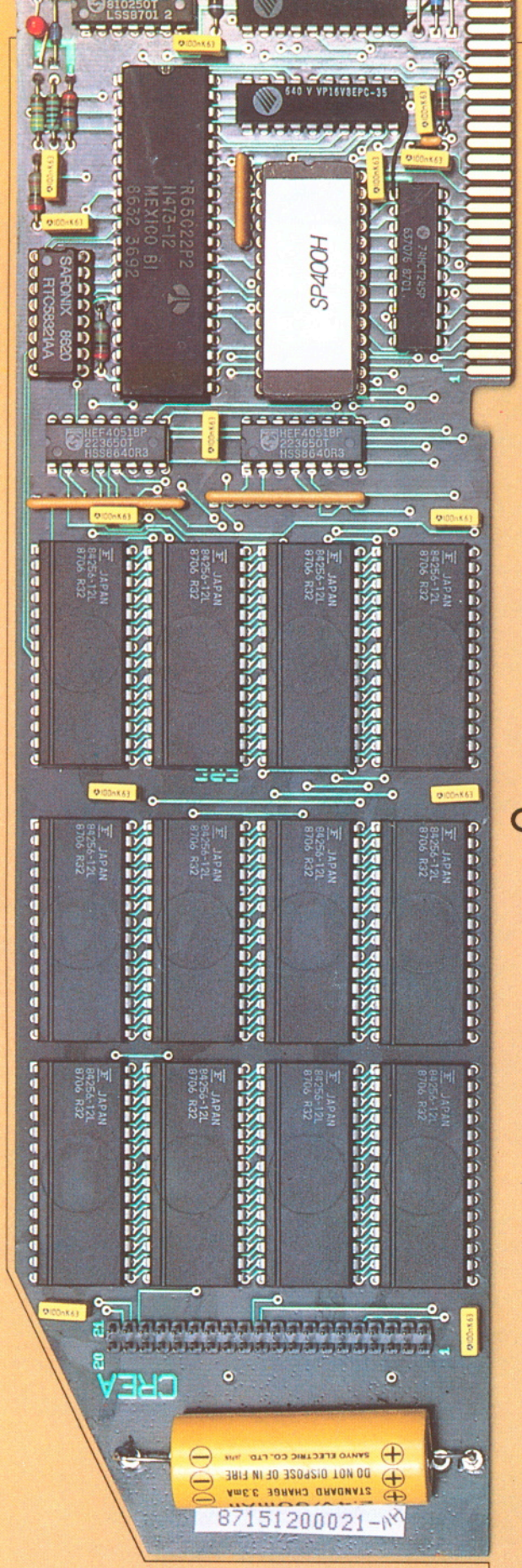
Nom : _____

Adresse : _____

Règlement par : Carte Bleue/VISA ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ date d'expiration _ _ _ _ _ _

Montant _ _ _ _ , _ _ F Signature :



L'anti-disque

Speedisk, la RAM Card à mémoire permanente

Rapidité Temps d'accès à l'information : 0,2 ms (100 fois plus rapide que les disques durs...).
Par exemple :
démarrage sur Basic.System en 3 s,
AppleWriter disponible en à peine 1 s.

Fiabilité Constituée de circuit CMOS à très faible consommation, Speedisk est aussi fiable que l'ordinateur lui-même. Elle est insensible à l'environnement.

Capacité Speedisk est proposée en quatre versions :
1 Mo (1 048 576 octets)
384Ko extensible à 1Mo
avec horloge compatible ProDOS (pour les Apple)(+, //e) ou sans (IIGS)

Compatibilité 100% compatible avec ProDOS (c'est un volume), Speedisk fonctionne sur Apple)(+*, sur Apple //e et sur Apple IIGS.

Prix Lecteur de Pom's, vous bénéficiez d'une remise de 10 % :

SP400		
384Ko	~~3 990,00~~	**3 591,00**
SP1000		
1 Mo	~~5 990,00~~	**5 391,00**
SP400H		
384Ko horloge	~~4 580,00~~	**4 122,00**
SP1000H		
1 Mo horloge	~~6 580,00~~	**5 922,00**

Garantie Speedisk est une fabrication française garantie un an par échange de la carte.

* Sur Apple)(+, ProDOS nécessite la carte langage.

Vente par correspondance, documentation, renseignements
LOGMA S.A.
12, rue d'Anjou
78000 Versailles
Tél : (1) 39 51 24 43

Je désire recevoir – sans engagement – votre documentation sur les cartes Speedisk

Nom.....

Adresse.....

.....

.....

Speedisk