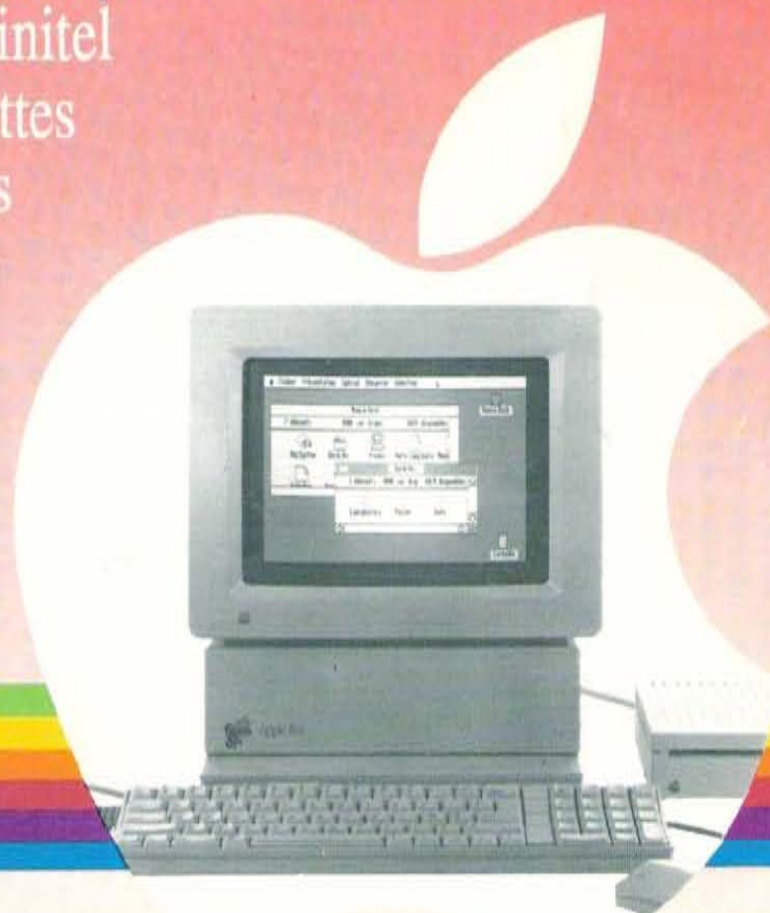


La revue francophone indépendante pour les utilisateurs des  
Apple ][+, //e, //e+, //c, IIGS™ et Macintosh™

# potom's

Apple & MINITEL

- 🍏 Minitel, Apple //, Macintosh : 3 programmes
- 🍏 Communication sous CP/M
- 🍏 Apprentissage du FORTH
- 🍏 Apple IIGS, premiers pas
- 🍏 TéléAlarme par Minitel
- 🍏 Éjection des disquettes
- 🍏 Un carnet d'adresses
- 🍏 Mac : Mise En Page
- 🍏 Crayon optique
- 🍏 Trucs & Essais



NUMERO 27 - PRIX 45 F

ISSN : 0294-6068



## LES BEST SELLERS

### PROGRAMMER APPLE

- **Système ProDOS sur Apple**  
par M. Cottini — 328 pages 190,00 FF
- **Programmation système de l'Apple**  
par M. Cottini — 320 pages 190,00 FF
- **Apple, Modems et serveurs**  
par A. Mariatte — 224 pages 130,00 FF \*
- **Clefs pour Apple // 65C02**  
par N. Bréaud Pouliquen  
172 pages 130,00 FF

### UTILISER APPLE

- **Appleworks au travail**  
par J.M. Jégo et A. Gargadenec  
192 pages 160,00 FF \*
- **50 modèles Multiplan pour gérer sur Apple et IBM/PC**  
par P. Gysel — 200 pages 130,00 FF \*
- **Apple, logique et systèmes experts**  
par R. Descamps — 224 pages 120,00 FF \*
- **Création et animation graphique sur Apple II**  
par G. Fouchard et J.Y. Corre  
Livre-disquette 335,00 FF

### JOUER SUR APPLE

- **Super Jeux Apple**  
par J.F. Sehan — 256 pages 120,00 FF

### UTILISER MACINTOSH

- **Clefs pour Macintosh**  
par F. Blanc et P. Brandeis  
204 pages 150,00 FF
- **Multiplan pour Macintosh**  
par H. Thiriez — 312 pages 110,00 FF \*
- **Le livre de Jazz**  
par J.C. Krust et P. Auchatraire  
312 pages 220,00 FF
- **Basic Microsoft 2.0 sur Macintosh**  
par M. Miller et K. Knecht  
380 pages 250,00 FF

### NOUVEAUTES MACINTOSH

- **Mac Astuces - Mac et Mac Plus**  
Deuxième édition  
par H. Thiriez — 288 pages 160,00 FF \*  
*Des explications claires sur de nombreux aspects du Mac et de multiples astuces sur son système et ses logiciels courants.*
- **Le livre d'Excel sur Macintosh**  
par P. Auchatraire — 384 pages 245,00 FF  
*Découvrez progressivement grâce à des exemples le tableur d'Excel sur Mac et Mac Plus.*

### A PARAÎTRE

- **Macintosh efficace**  
par P. Brandeis et F. Blanc  
168 pages 150,00 FF

ENVOYER CE BON ACCOMPAGNÉ DE VOTRE RÈGLEMENT À P.C.V. DIFFUSION  
BP 86 — 77402 Lagny/Seine-et-Marne Cedex

DESIGNATION	PRIX
Frais de port	10,00 FF
<b>TOTAL</b>	

- Paiement par chèque joint
  - Paiement par Carte Bleue Visa (P.C.V. Diffusion uniquement)  
N° \_\_\_\_\_  
Date d'expiration \_\_\_\_\_
  - Je demande le catalogue P.C.V. gratuit
- Nom \_\_\_\_\_  
Prénom \_\_\_\_\_  
Rue \_\_\_\_\_ N° \_\_\_\_\_  
Ville \_\_\_\_\_ Code Postal \_\_\_\_\_

Signature :



Gagnez du temps :  
Ne recopiez pas tous les programmes du livre, commandez la disquette qui vous les fournit directement utilisables sur votre ordinateur  
Prix : 150,00 FF  
en vente par correspondance uniquement à P.S.I.



Numéro 27  
novembre-décembre 86

## Éditorial

Hervé Thiriez



Page 5

## TéléAlarme Télématique

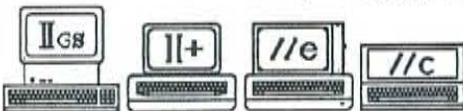
Bernard Hoyez et  
Jean-Luc Nail



Page 6

## Un carnet d'adresses

Boris Fivolewski



Page 11

## Éjectez les disquettes 3,5'

Bruno Fénart



Page 18

## L'apprenti-FORTHier

Fej Nissuorgsky



Page 19

## Carte Super Série et CP/M

Seconde partie

Jean-François  
Rabasse



Page 29

## Mise En Page

Georges Zwingelstein



Page 40

## MacAstuces

Page 42



## Essais

Page 43



## Minitel & Macintosh

Jean-Luc Bazanegue



Page 44

## le Minitel



Page 53

## Minitel & Apple II

Christian Piard



Page 55

## l'Apple IIGS



Page 65

## Un crayon Optique

Christian Piard



Page 67

## Micro-informations



Jean-Michel Gourévitch

Page 70

## Bibliographie

Alexandre Duback



Page 73

Les annonceurs ; Apple : pages 38 et 39. Badaroux C.A. : page 37. Infomag : page 75. PSI : page 76.

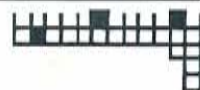
Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39 51 24 43. Directeur de la publication : Hervé Thiriez

**Pom's présente :**

*pour Apple ][+, IIe, IIe+, IIc*

# Ordico



Destiné aux amateurs de mots croisés ou de Scrabble, cette base de données, due à Roland Jost, permet de trouver un mot de longueur donnée dont on ne connaît que quelques lettres.

Ordico contient plus de 15000 mots classés en 70 rubriques.  
Recherches et affichages sont rapides : un fichier de 1500 mots est chargé en moins de 10 secondes et exploité quasi-instantanément.  
Il est bien sûr possible d'ajouter des termes aux divers fichiers, de créer de nouvelles rubriques.

Voici quelques rubriques :

*1ère face :*

*Acteurs, Animaux, Armes/guerres, Auteurs américains, Auteurs anglais, Auteurs français, Chimie, Cinéastes, Coureurs cyclistes, Départements/régions, Dieux/déeses, Familles végétales, Femmes célèbres, Hommes politiques, Iles, Jeux/sports, Minéraux, Montagnes, Musiciens jazz, Musiciens, Parties du corps, Peintres étrangers, Peintres français, Rivières/fleuves, Saints/saintes, Savants/inventeurs, Sculpteurs, végétaux, Vêtements, Villes*

*2ème face :*

*Athlètes, Boxeurs, Cantatrices, Cols, Cosmonautes, Coureurs automobiles, Déserts, Détroits, Doctrines philosophiques, Drogues, Escrimeurs, Explorateurs, Gymnastes, Haltérophiles, Judokas, Lutteurs, Maladies, Maréchaux de France, Médicaments, Nageurs, Patineurs, Poissons, Présidents américains, Skieurs, Ski nordique, Unités, Villes olympiques*

**Exemples :**

Un musicien dont le nom comporte 7 lettres, les 2ème et 5ème sont des 'E'. Tapez :

-E -E- -. Vous obtenez instantanément :

*BENNETT DEBOECK DELEEUW GEVAERT LESUEUR PEDRELL PEETERS WELLESZ*

Dans les acteurs, -A- - - -E vous donnerait :

*CARETTE PALANCE RACETTE RANDONE RAYMONE VALLONE*

et - - - -H- :

*CAUCHY CEECHI ENIGHT VAUGHN WRIGHT*

- - - - - dans les femmes célèbres donnerait 32 noms...

Disquette double face et documentation : 200,00 F franco. Bon de commande page 74

# Éditorial

*Trait d'union entre la série II et le Macintosh ? Il ressort de nos premiers contacts avec le IIGS qu'Apple a réussi à construire un système ouvert, pratique comme le Mac, doté de la philosophie Mac, la couleur en plus, le 68000 en moins. Le côté évolution de l'Apple II est à notre sens un atout décisif pour les adeptes du prédécesseur : il était malaisé de passer du 6502 au 68000 car il s'agit de deux mondes différents. Avec le 65816, on se sent chez soi et, à en croire nos entretiens téléphoniques, nombre d'entre-vous sont déjà séduits. La présentation de la page 65 vous en dira plus.*

*Le numéro 1 de Pom's est sorti en septembre 81, à 35 F ; plus de cinq ans après, nous sommes hélas obligés d'augmenter le prix pour la seconde fois, le numéro passant à 45 F. Le tarif était resté stable depuis le numéro 9, ce qui reste honorable, d'autant plus que la densité d'informations s'accroît de numéro en numéro : la typographie réduite pour les listings nous permet de vous proposer des programmes toujours plus riches. Là où nous pouvions vous proposer une routine à utiliser dans vos programmes, un programme complet (sept présentement), d'utilisation aisée, trouve sa place.*

*Comme vous le découvrez, ces pages sont très orientées vers la communication. Le Minitel prend la dimension d'un phénomène (on parle de deux millions d'appareils en service à la fin de cette année) et Pom's devait s'y consacrer car la gratuité de ce modem nous donne une ouverture vers la communication, ouverture qu'il serait dommage d'ignorer. Un système de Téléalarme et des logiciels d'enregistrement et restitution des écrans sont au programme : l'utilisation de la prise péri-informatique et du modem vous sera bientôt familière.*

*Autre aspect de la communication, le protocole XMODEM sous CP/M qui décrit l'art et la manière de transmettre des informations avec l'assurance qu'elles sont bien comprises.*

*Pour l'Apple II, deux autres programmes d'utilisation immédiate : l'un pour approcher le FORTH, l'autre pour gérer un carnet d'adresses, doté d'une bonne ergonomie. Ce dernier reçoit de façon optionnelle le composeur téléphonique des numéros 22/23 ce qui forme alors un ensemble intéressant.*

*Nous n'abandonnons pas l'aspect utilitaire des routines chères à Pom's : un crayon optique, pratique, sera certainement connecté demain à l'entrée des manettes de jeux de votre IIe ou IIc. Autre utilitaire : comment éjecter des disquettes 3'5. Pratique également, mais sans rapport, l'application 'Mise En Page', œuvre d'un auteur passé récemment de l'Apple II au Macintosh.*

**Hervé Thiriez**

**Ont collaboré à ce numéro :** Alexandre Avrane, Jean-Luc Bazanegue, Bruno Fénart, Boris Fivolewski, Jean-Michel Gourévitch, Olivier Herz, Bernard Hoyez, Gérard Michel, Jean-Luc Nail, Fej Nissuorgsky, Christian Piard, Jean-François Rabasse, Hervé Thiriez, Georges Zwingelstein.

**Directeur de la publication, rédacteur en chef :** Hervé Thiriez.

**Rédacteurs :** Alexandre Avrane, Olivier Herz.

**Siège social :** Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39.51.24.43.

**Publicité :** Éditions MEV.

**Diffusion :** N.M.P.P.

**Impression :** Rosay - 47, avenue de Paris - 94300 Vincennes. Tél. : (1) 43.28.18.63.

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.

# TéléAlarme Télématique

Bernard Hoyez, Jean-Luc Nail

**L**e développement parallèle de la micro-informatique et de la télématique fait souvent voisiner sur une même table micro-ordinateur et Minitel. En dehors de leur air de famille, ils s'ignorent souvent l'un l'autre. Il suffit cependant de peu de chose pour les marier et que naissent des applications nouvelles. Celle que nous vous présentons s'adresse à un Apple II, mais la transposition est réalisable simplement sur un autre micro-ordinateur ayant un port série.

Peut-on apprendre à notre ordinateur à téléphoner tout seul, c'est-à-dire composer un numéro et répondre intelligiblement ? On peut envisager par exemple la connexion du haut-parleur interne sur la ligne téléphonique (par l'intermédiaire d'un transformateur de téléphone). Ce dispositif permettrait à l'Apple de servir de répondeur téléphonique, en utilisant un programme de synthèse vocale par exemple. Malheureusement, la synthèse et la reconnaissance vocales sur un Apple II, sans périphérique spécialisé, n'offre pas de bons résultats. Aussi, notre réalisation n'utilisera pas la parole comme moyen d'échange, mais le texte. Rangez donc votre combiné téléphonique dans le placard et sortez votre Minitel qui s'avèrera plus adapté à la situation.

## Le principe

Plus qu'un simple dispositif d'alarme, cette réalisation est un véritable système de contrôle à distance d'une installation. Le cas choisi ici, un dispositif anticambriolage, nous servira d'exemple, mais vous pourrez l'adapter à la surveillance

d'appareils mécaniques, électriques ou scientifiques.

**Les acteurs de la pièce sont les suivants :**

- d'un côté l'Apple avec pour acolytes des capteurs, des relais et un MINITEL ;
- de l'autre côté, Monsieur X ou un complice, Monsieur Y, comme lui possesseur d'un Minitel ;
- entre les deux, la ligne téléphonique.

**Le scénario se déroule ainsi :**

**Acte 1 :** une maison isolée. Arsène Lupin crochète la serrure. Au moment où la porte s'ouvre, un microcontact se ferme. L'Apple, en chien de garde, détecte l'ouverture. Aussitôt, il compose secrètement le numéro de téléphone de Monsieur X.

**Acte 2 :** la sonnerie du téléphone de Monsieur X retentit. Monsieur X décroche le combiné et, dans l'écouteur, lui parvient un sifflement aigu, quelque chose qui lui rappelle le 1700 Hertz de l'annuaire électronique. Monsieur X a compris et connecte son Minitel. Il voit apparaître sur son écran une supplique le pressant à se manifester. Il frappe sur son clavier la réponse souhaitée. Alors, un message plus explicatif s'offre à ses yeux : le lieu d'effraction, la position exacte du circuit atteint, l'état de certains capteurs. Une question lui est enfin posée : doit-il ordonner à l'Apple une riposte sur le lieu de l'effraction et laquelle ?

**Variante à l'acte 2 :** la sonnerie du téléphone de Monsieur X retentit. Monsieur X est absent. Au bout d'une minute, l'Apple impatient raccroche, puis compose le numéro de Monsieur Y, autre familier qui est heureusement bien chez lui... etc.

(cette variante peut être répétée jusqu'à épuisement de l'agenda).

**Acte 3 :** Monsieur X (ou Y) déclenche la riposte adéquate en tapant le numéro sur son clavier (une sirène, les gaz paralysants, ou le pulvérisateur d'huile de putois, au choix du scénariste). Happy end, le voleur est arrêté.

## Le dispositif

Les éléments nécessaires à la réalisation du système de téléalarme sont les suivants :

- un Apple II+ ou, de préférence, IIe (pour le IIc c'est plus complexe) muni d'une carte super-série (SSC entre nous) ;
- un Minitel réversible (presque tous le sont maintenant) ;
- un petit circuit de composition automatique de numéro semblable à celui proposé dans le numéro 22 de Pom's ;
- un certain nombre de micro-contacts à aimants (autant que de portes ou de fenêtres à protéger). Ces micro-contacts équipent les dispositifs de sécurité. Normalement, en veille, ces contacts sont fermés. Remplacer l'ampoule par une ampoule à contacts ouverts lorsque le plot aimanté est rapproché.

*Les micro-contacts sont groupés, en série, sur une entrée à un bit (boutons-poussoirs). Il y a 3 entrées (PB0, PB1, PB2) sur la prise de jeux, vous pourrez donc avoir trois groupes différents de micro-contacts.*

*Vous reliez l'autre borne de l'interrupteur au +5 volts (broche 1 du connecteur interne), via une résistance d'environ 1 K $\Omega$  pour atténuer le courant.*

Rappelons que les entrées logiques sont accessibles sur les broches 2, 3 et 4 du connecteur interne ou sur les broches 7, 1 et 6 du connecteur arrière du II.

- de 0 à 3 relais, selon les besoins, montés selon le même principe que pour le composeur de numéros. Ces relais sont connectés aux sorties-témoins ou annonceurs. Les sorties logiques sont accessibles sur les broches 15, 14, 13 et 12 du connecteur interne (hélas pas sur le connecteur arrière). Comme l'une de ces sorties est déjà occupée par le composeur de numéros, trois sont encore disponibles (AN1, AN2 et AN3);
- de 0 à 4 capteurs à résistance variable ou potentiomètres de 150 K $\Omega$ . Ces capteurs seront reliés directement aux entrées des manettes de jeux (GC0, GC1, GC2 et GC3). Dans le programme proposé, nous ne ferons que lire la valeur prise par ce capteur, mais vous pouvez bien entendu envisager de déclencher l'alarme au-delà d'un certain seuil.

*Une application classique peut être la détection d'incendie ou de gel, à l'aide d'une thermistance. Une valeur préconisée pour cette thermistance peut être de 15 K $\Omega$  à la température ambiante et un rapport R 25° / R 50° entre 9 et 10. Des essais ont montré la linéarité de la fonction  $R = f(PDL(x))$ . Pour  $PDL(x) = 255$ , la résistance est environ de 115 K $\Omega$ .*

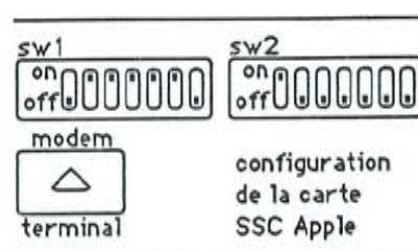
*D'autres applications : adapter une photo-résistance pour déclencher en fonction de la lumière ambiante, une jauge résistive de contrainte en fonction d'un effort ou d'une pression, un accéléromètre en fonction de chocs...*

- une liaison SSC-MINITEL : plusieurs solutions s'offrent à vous. Se reporter à la présentation générale page 53 et au plan de câblage dans l'article Minitel/Apple page 55.

Attention si vous êtes bricoleur, ne réunissez votre Minitel à votre Apple qu'après avoir débranché sa prise secteur. Il ne suffit pas de l'éteindre, mais carrément de l'écartier du 220 volts, car le pauvre n'a qu'une seule tare, il ne possède pas de transformateur d'isolement.

La carte SSC est logée dans le slot 2. Pour une autre implantation, quelques modifications doivent être apportées au programme.

Enfin quelques réglages préliminaires permettent de s'affranchir de lignes de programmes superflues. Voici la configuration :



## Le programme

Trois sortes de considérations sont abordées dans l'écriture du logiciel de téléalarme : la gestion des entrées/sorties sur le connecteur de jeu, la programmation de l'ACIA de la carte super-série et enfin l'utilisation du Minitel.

### La prise de jeu, prise au sérieux

Pourquoi chercher plus compliqué ? Le connecteur de jeu offre une variété de signaux d'entrées/sorties propre à nous satisfaire.

Vous pourrez trouver, page 53 de ce numéro, un article sur les problèmes généraux de liaison Apple/Minitel

À chaque entrée logique est associé un commutateur logiciel. Leurs adresses-mémoire sont -16287 à -16285 ou \$C061 à \$C063. C'est le bit de poids fort (bit 7) qui informe de l'état de la commutation alors que le reste de l'octet est indéfini.

En Basic, il suffit de lire le commutateur avec un PEEK et de le comparer avec la valeur 128. Si la valeur est égale ou supérieure à 128, le commutateur est à 1 ou, pour notre application, le micro-contact est fermé (effraction).

Pour les entrées analogiques, l'Applesoft donne un moyen rapide d'évaluation par l'instruction PDL, avec pour argument le numéro (0 à 3) de la manette. Chaque capteur doit évidemment être étalonné.

Pour les sorties logiques (ici les relais), la commutation logicielle utilise cette fois une paire d'adresses, une met l'annonceur à 0 tandis que l'autre le met à 1. Le simple fait d'accéder à une adresse (une lecture par exemple) suffit pour l'activer.

Le composeur automatique de numéros téléphoniques illustre un exemple de leur programmation. Les trois paires d'adresses libres vont de -16294 (\$C05A) à -16289 (\$C05F).

### L'ACIA, du travail en série

La manière la plus simple d'utiliser la SSC est encore de se servir du 'firmware' de la carte. Tapez les commandes PR#2 et IN#2 en mode direct, ou précédées par un CHR\$(4) dans un programme Basic, et le tour est joué. Tout ce que vous tapez sur le clavier de l'Apple est instantanément transféré sur les écrans de l'Apple et du Minitel, et de même si vous tapez sur le clavier du Minitel.

Le revers de cette facilité est, d'une part une certaine lenteur et, d'autre part, un manque de contrôle sur les incidences du logiciel. Pour ces raisons, nous avons préféré programmer direc-

tement les entrées-sorties de l'ACIA (l'ACIA, ou 'Asynchronous Communications Interface Adapter' est le circuit intégré principal chargé de la 'sérialisation' et des transferts de données).

Deux registres sont à connaître : le registre tampon double d'émission/réception et le registre d'état, qui ont respectivement leur image en mémoire aux adresses \$COA8 (TDREG / RDREG) et \$COA9 (STATUS), si la carte est enfichée dans le slot 2 (pour l'activer, faites un saut en \$C200).

Chaque bit du registre d'état, pris isolément, indique l'état de certains registres ou de certains signaux. Dans notre cas, ce sont les bits 3 et 4 qui retiennent notre attention. Le bit 3 est positionné à 1 si le tampon de réception est plein. Attention, c'est l'inverse pour le bit 4 qui est à 1 si le tampon de transmission est vide.

Pour tester si un caractère est entré ou sorti, il suffit de faire respectivement un AND \$8 ou un AND \$10 avec l'octet de status et d'examiner l'indicateur de zéro.

Le caractère reçu ou à émettre est placé dans la mémoire 9 qui intervient dans les transferts entre la routine machine et le programme Basic.

### Le Minitel, agent de liaison

Le Minitel est un Modem bien particulier qui interprète certaines séquences de codes pour son propre compte. L'ensemble de ces mots réservés constitue le vocabulaire de la norme Videotex Télétel détaillée dans la notice technique du Minitel ou dans quelques ouvrages disponibles en librairie. Un, deux, trois, quatre ou cinq octets servent à construire

ces mots. Non seulement, le Minitel comprend ce langage, mais il y répond parfois en émettant une séquence appropriée.

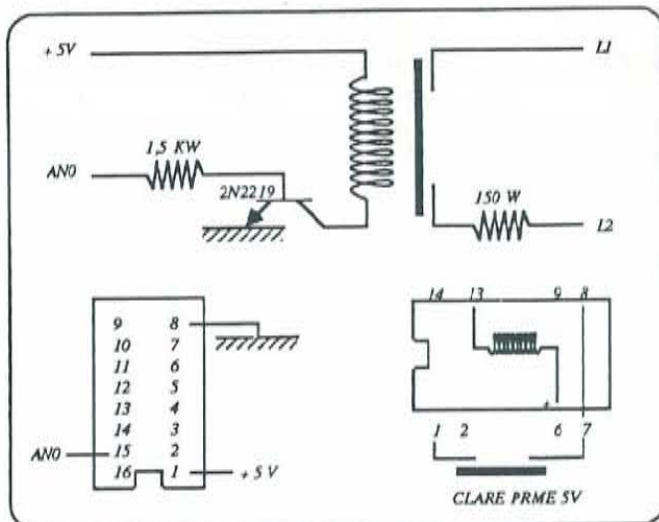
Les principales fonctions du langage sont les suivantes :

- modifier le mode de fonctionnement du terminal (vitesse d'échange avec le périphérique, aiguillage des différents modules, connexion et déconnexion...);
- modifier les caractéristiques de visualisation (passage du jeu alphabétique au jeu semi-graphique, taille et couleur des caractères...);
- générer les séquences de codes correspondant aux touches de fonction (Envoi, Suite, Annulation...);
- indiquer l'état du terminal (vitesse, aiguillages des modules, présence d'une porteuse...);
- mettre en marche la procédure de correction d'erreurs ;

## Programme 'TELEALARME'

```

1 REM *****
2 REM *
3 REM * TELEALARME *
4 REM *
5 REM * Bernard HOYEZ *
6 REM * Jean-Luc NAIL *
7 REM *
8 REM *****
9 :
```



Ce schéma est extrait du numéro 22 de Pom's ("Composition de numéros de téléphone")

```

10 GOSUB 100: REM chargement des routines
    et des variables
20 GOTO 1000: REM programme principal
30 :
100 REM *** COMMANDES VIDEOTEX *****
    *****
110 ES$ = CHR$ (27)
120 EF$ = CHR$ (12): REM EFFACEMENT D'EC
    RAN
130 RC$ = CHR$ (10) + CHR$ (13)
140 DH$ = ES$ + CHR$ (77): REM DOUBLE HA
    UTEUR
150 FL$ = ES$ + CHR$ (72): REM FLASH
160 VT$ = ES$ + CHR$ (82): REM FOND VERT
170 P1$ = ES$ + CHR$ (57): REM PRO1
180 CO$ = P1$ + CHR$ (104): REM CONNEXIO
    N
190 DE$ = P1$ + CHR$ (103): REM DECONNEX
    ION
200 OP$ = P1$ + CHR$ (111): REM OPPOSITI
    ON
210 :
300 REM *** MESSAGES *****
    *****
310 ALERTE$ = EF$ + RC$ + RC$ + VT$ + DH$
    + FL$ + " ALERTE " +
    CHR$ (31) + "KA" + DH$ + "REPONDEZ E
    N TAPANT " + VT$ + " OK "
320 M1$ = EF$ + RC$ + DH$ + "MERCI D'AVOIR
    REPONDU" + RC$ + RC$ + "CECI ETAIT UN
    E SIMULATION D'EFFRACTION" + RC$ + RC$
    + "CHEZ MARTIN .."
```



- télécharger les mémoires vives du Minitel ou identifier au contraire ces mémoires.

La connaissance de la norme Télétel est indispensable pour l'écriture d'un logiciel serveur ou, inversement, pour l'émulation d'un Minitel. Elle présente moins d'importance dans notre application de téléalarme. Les séquences de codes essentielles ou les plus usitées ont été introduites dans des variables chaînes, rassemblées en tête du programme. Pour les transmettre, il suffit de faire appel au sous-programme d'affichage d'une chaîne (ligne 800). Retenez surtout les séquences de connexion (CO\$), de déconnexion (DE\$), d'opposition (OP\$); les autres ne concernent que la mise en page des messages.

Une fois votre dispositif en place, vérifiez son bon fonctionnement :

- en dehors de l'alarme, le dialogue entre l'Apple et le Minitel s'établit-il lorsque vous validez la carte SSC (PR#,IN#) ? ;
- avant d'importuner vos amis, testez "en local" la réponse à une alarme simulée. Pour cela, 'shuntez' la demande de connexion en oblitérant les lignes 1080 et 1090. Supprimez l'appel au sous-programme de composition de numéro téléphonique (ligne 1050). Pour déclencher l'alarme, utilisez indifféremment le micro-contact, la touche  ou le bouton-poussoir. Votre Minitel doit alors afficher les messages d'alerte auxquels vous répondrez sur son clavier. Un 'F', en haut à droite de l'écran précise que vous n'êtes pas connecté.
- enfin, assurez vous que l'appel téléphonique automatique se réalise bien et que le Minitel prend la ligne. Rétablissez

pour cela toutes les lignes du programme et ajoutez en DATA une liste de vos proches amis. Normalement, l'Apple affiche le numéro qu'il compose puis un 'C' prend alors la place du 'F' sur l'écran du Minitel.

Si tous vos essais sont fructueux, vous êtes en possession d'un système d'alarme bon marché et adaptatif dont les performances rivalisent avec celles d'appareils spécialisés.

Plus encore, ces quelques lignes vous auront peut-être fait découvrir les immenses possibilités d'ouverture de votre ordinateur sur le monde extérieur.



```

330 M3$ = "MESURE ANALOGIQUE "
340 M4$ = "NUMERO DE RIPOSTE (1,2,3) ? "
350 :
400 REM *** ROUTINES MACHINE *****
*****
410 SLOT = 49664
420 ENTRE = 768
430 AFFICHE = 779
440 PTT = 792
450 :
460 FOR I = 768 TO 832
470 READ X: POKE I, X
480 NEXT I
490 DATA 173,169,192,41,8,173,168,192,13
3,9,96
500 DATA 173,169,192,41,16,240,249,165,
9,141,168,192,96
510 DATA 165,7,201,176,144,17,201,186,
176,13,41,15,208,2,169,10,170,32,48,3,
202,208,250,96
530 DATA 169,159,141,88,192,32,168,252,1
69,108,141,89,192,32,168,252,96
540 RETURN
550 :
600 REM ***COMPOSITION NUMERO *****
*****
610 HOME
620 READ X$: IF X$ = "FIN" THEN END
630 POKE 49241,0: REM PRISE DE LIGNE
640 FOR I = 1 TO 700: NEXT I
650 HOME
660 PRINT : PRINT "Je compose le numéro "

/
670 FOR I = 1 TO LEN (X$)
680 A = ASC ( MID$ (X$, I, 1)) + 128
690 POKE 7,A
700 CALL PTT
710 PRINT CHR$ (A);
720 FOR T = 1 TO 600: NEXT T
730 NEXT I
740 RETURN
800 REM *** AFFICHAGE D'UNE CHAINE ****
*****
810 FOR J = 1 TO LEN (A$)
820 X = ASC ( MID$ (A$, J, 1))
830 POKE 9,X
840 CALL AFFICHE
850 NEXT J
860 RETURN
870 :
1000 REM *** PROGRAMME PRINCIPAL *****
*****
1010 IF PEEK ( - 16287) > 128 THEN P0% =
1
1020 IF PEEK ( - 16286) > 128 THEN P1% =
1
1030 IF PEEK ( - 16285) < 128 THEN P2% =
1
1040 IF NOT (P0% OR P1% OR P2%) THEN 100
0
1050 GOSUB 600: REM APPEL***
1060 FOR Q = 1 TO 200: NEXT Q
1070 CALL SLOT
1080 A$ = OP$ + CO$

```

```

1090 GOSUB 800: REM AFFICHAGE
1100 F1 = 0:F2 = 0
1110 T = 0
1120 T = T + 1
1130 A$ = ALERTE$
1140 GOSUB 800
1150 FOR I = 1 TO 200
1160 CALL ENTRE
1170 IF PEEK (9) = ASC ("O") THEN F1 =
1
1180 CALL ENTRE
1190 IF PEEK (9) = ASC ("K") THEN F2 =
1
1200 IF F1 AND F2 THEN 1250
1210 NEXT I
1220 IF T > 5 THEN POKE 49240,0
1230 IF T > 10 THEN A$ = DE$: GOSUB 800:
FOR G = 1 TO 500: NEXT G: POKE 49321,1
28: GOTO 1050
1240 GOTO 1120
1250 IF P0% THEN S0$ = " CIRCUIT 1 ACTION
NE" + RC$ + RC$
1260 IF P1% THEN S1$ = " CIRCUIT 2 ACTION
NE" + RC$ + RC$
1270 IF P2% THEN S2$ = " CIRCUIT 3 ACTION
NE" + RC$ + RC$
1280 E0$ = STR$ ( PDL (0)):E1$ = STR$ (
PDL (1)):E2$ = STR$ ( PDL (2)):E3$ =
STR$ ( PDL (3))
1290 A$ = M1$ + CHR$ (31) + "JA" + DH$ +
VT$ + S0$ + S1$ + S2$ + RC$ + RC$ + M3
$ + " 1 = " + E0$ + RC$ + RC$ + M3$ +
" 2 = " + E1$ + RC$ + RC$ + M3$ + " 3
= " + E2$ + RC$ + RC$ + M3$ + " 4 = "
+ E3$
1300 GOSUB 800
1310 A$ = RC$ + RC$ + M4$
1320 GOSUB 800
1330 CALL ENTRE
1340 X% = PEEK (9) - 48
1350 IF (X% < 1 OR X% > 3) THEN 1330
1360 PRINT PEEK (49241 + X%)
1370 A$ = RC$ + RC$ + VT$ + " RIPOSTE " +
STR$ (X%) + " ACTIONNEE "
1380 GOSUB 800
1390 END
1400 DATA 3541: REM MONSIEUR X
1405 DATA 35654321 : REM MONSIEUR Y
1410 DATA FIN

```

## Source 'T.ALARME' Assembleur Big Mac

NE : Ce source, sur la disquette d'accompagnement de la revue, est sauvegardé en format 'TEXT' pour permettre la récupération par votre assembleur.

```

1 *****
2 *
3 * ALARME
4 *
5 *****
6
7 * 9 AVRIL 86
8
9 STATUS = $COA9 ;Registre d'état carte SSC
10 DATA = $COA8 ;Registre entrée/sortie données
11 CHIFFRE = $7
12 CARACT = $9 ;paramètre transfert Basic
13 SETANG = $C059 ;sortie binaire paddle mise à 1
14 CLRANG = $C058 ;sortie binaire paddle mise à 0
15 WAIT = $FCA8 ;routine tempo du moniteur
16
17 ORG $300
18
19 *----> Entree de caractère
20
21 TEST LDA STATUS ;Si le bit 3 du registre d'état est à 1
22 * ;ceci n'a pas d'utilité dans notre cas
23 AND #08 ;un caractère est présent
24 * ; on aurait pu le tester par un BEQ
25 LDA DATA ;lecture de la donnée
26 STA CARACT
27 RTS
28
29 *----> Sortie de caractère
30
31 LOOP LDA STATUS ;Si le bit 4 du registre d'état est à 1
32 AND #$10 ;on peut transmettre la donnée
33 BEQ LOOP
34 LDA CARACT
35 STA DATA ;Sortie du caractère
36 RTS
37
38 *----> Composition numéro
39
40 LDA CHIFFRE
41 CMP #0*
42 BCC SEPRE ;séparateur entre chiffres
43 CMP #'9'+1
44 BCS SEPRE
45 AND #1111 ;filtrage
46 BNE COMPOSE
47 LDA #10 ;chiffre=0 <A>=10
48 COMPOSE TAX
49 BOUCLE = *
50 JSR PULSE
51 DEX
52 BNE BOUCLE
53 SEPRE RTS
54
55 PULSE LDA #159 ;délai de 66 ms
56 STA CLRANG ;coupure de la ligne téléphonique
57 JSR WAIT
58 LDA #108 ;délai de 33 ms
59 STA SETANG ;rétablissement de la ligne téléphonique
60 JSR WAIT
61 RTS

```

## Récapitulation 'ALARME'

Ce code est inclus dans le programme Basic.

Accompagné d'une cinquantaine de pages de documentation, Disk Manager permet de recréer les commandes du Dos, redéfinir l'organisation d'une disquette, grâce à un jeu d'instructions qui en fait un langage simple d'accès à la disquette. Il offre également un programme d'édition à l'aide de commandes évoluées. 4 utilitaires figurent aussi sur la disquette :

- Utili-disque : reconstruction d'une disquette détruite, Vérification, Plan d'occupation
- Ultra-copie : pour un backup particulièrement rapide
- Edicat : Edition du catalogue, classement des fichiers, Titres...
- Multi-disque : pour le classement de tous vos programmes (tri instantané).

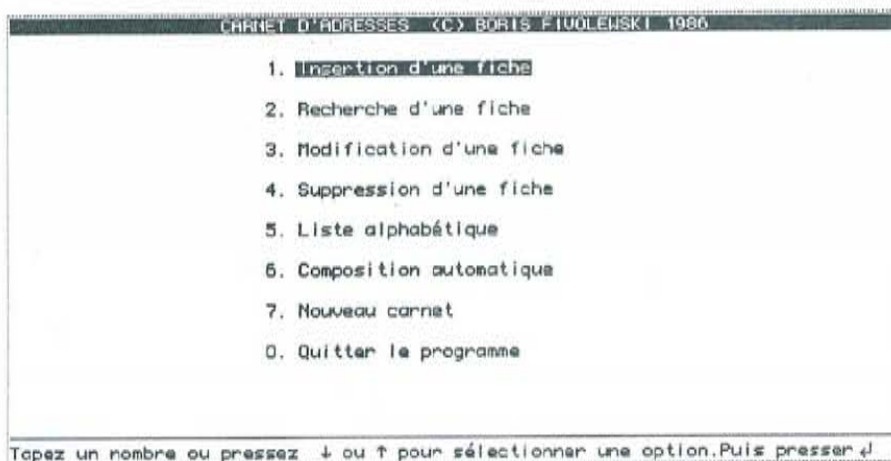
# Disk Manager le Dos en Kit

de Dan Steerey

# Un Carnet d'Adresses

Boris Fivolewski

**S**ans remplacer les carnets d'adresses traditionnels, ni avoir la prétention d'égaliser une gestion de fiches sophistiquée, ou un logiciel de communication, ce programme offre des possibilités intéressantes (du moins nous l'espérons) comme la composition automatique d'un numéro de téléphone. Et surtout, il est écrit essentiellement en Basic ; vous pourrez donc le modifier facilement en fonction de vos besoins.



## Mode d'emploi

La gestion du fichier d'adresses est volontairement simplifiée pour permettre une utilisation du programme aussi bien en DOS 3.3 ou en ProDOS.

Vous devez avoir sur la même disquette le programme CARNET et les deux fichiers ADRESSES et INDEX. S'il n'y a pas de fichier INDEX sur la disquette un nouveau carnet est créé au démarrage. Vous ne devez pas par la suite retirer la disquette du lecteur, les données étant lues au fur et à mesure en fonction des commandes, ce qui permet d'avoir des fichiers très importants (avec un unidisk par exemple) qui ne pourraient tenir entièrement en mémoire vive. Sur une disquette 5" 1/4 la limite est d'environ 400 fiches, mais avec d'autres supports vous pouvez aller jusqu'à 1200 fiches.

Attention : le fichier CARNET de type Basic est facilement convertible en ProDOS à l'aide des *Utilitaires Système*. Il n'en est pas de même pour les fichiers ADRESSES et INDEX qui sont à accès relatifs. Choisissez donc, avant d'enregistrer quoi que ce

soit, le système d'exploitation que vous utiliserez. Le passage de l'un à l'autre étant délicat par la suite.

## Description des commandes

### Introduction en continu

Les différentes rubriques sont les suivantes :

- Nom
- Prénom
- Adresse
- Ville
- Téléphone
- Renseignement

Les rubriques *téléphone* et *renseignement* peuvent être utilisées par exemple pour définir une procédure de connection automatique à un serveur en couplant la commande 6 (voir ci-après) avec un logiciel de communication.

La rubrique *Nom* n'accepte que les lettres majuscules et les caractères apostrophe et tiret ; les minuscules sont automatiquement converties en majuscules ; les autres caractères ne sont pas pris en compte. Pour les autres rubriques, il n'y a pas de limitation excepté pour les caractères de contrôle.

Pour passer à la rubrique suivante utilisez `↵` ou `↓`. Pour revenir à la rubrique (ou champ) précédent utilisez `↑`.

Dans un même champ vous pouvez vous déplacer à l'aide des touches `←` et `→`. Pour effacer le caractère que vous venez de taper, utilisez la touche "Delete".

Quand le dernier champ est complété, la fiche est enregistrée sur le disque. Si vous voulez retourner au menu principal sans enregistrer la fiche, pressez la touche 'Escape'. Sinon, complétez l'enregistrement puis pressez `↵` ou 'Escape'.

### Recherche d'un nom

Vous pouvez entrer soit le nom en entier, soit le nom abrégé suivi par un point. Dans le premier cas le programme affichera toutes les fiches portant le même nom, dans le deuxième cas toutes celles commençant par le nom abrégé quelle que soit la longueur de ces noms.

La recherche se fait dans un tableau en mémoire vive, elle est donc relativement rapide. Quand

les numéros des fiches recherchées sont trouvés, le programme lit leurs différents champs sur le disque. La clef utilisée est de 12 lettres ; au delà, les noms sont considérés comme identiques.

Quand vous avez trouvé la fiche, vous pouvez l'imprimer.

### Modification d'une fiche

La recherche de la fiche se fait de la même façon que ci-dessus, mais vous avez ensuite accès aux différents champs de la fiche afin de les modifier. Les possibilités sont les mêmes que pour l'enregistrement en continu.

### Suppression d'une fiche

La recherche de la fiche se fait de la même façon que ci-dessus. Avant de supprimer la fiche, le programme vous demande de confirmer.

### Liste alphabétique à partir d'un nom

La liste commence au nom que vous choisissez. Si vous voulez lister à partir du début, tapez directement  $\downarrow$ . Vous pouvez interrompre momentanément l'affichage à l'aide de la barre d'espace. N'importe quelle touche permet de reprendre la liste. Pour interrompre définitivement, pressez 'Escape'.

### Composition automatique

Vous avez sans doute trouvé que le programme qui accompagnait le composeur automatique de numéro de téléphone - Pom's numéro 22 pour le //e, 23 pour le //c - était un peu sommaire. L'option 6 permettra d'utiliser pleinement le circuit que vous aurez construit à cette occasion.

Si le numéro comprend des parenthèses le composeur considère les chiffres entrés comme un

indicatif et il vous demande s'il faut le composer ou non. De plus vous pouvez ajouter des temporisations pour attendre une tonalité après un indicatif. Le symbole '-' correspond à une pause d'une seconde, le symbole '+' à une pause de dix seconde. Exemples :

- numéro en région parisienne : (16 -- 1) 12 34 56 78 (Si l'indicatif est composé, la pause entre l'indicatif et le numéro est de 2 secondes)
- numéro en province : (16 +-) 12 34 56 78 (Dans ce cas la pause est de 11 secondes)

*Note : le programme, contrairement à ceux des numéros 22 et 23 de Pom's, détecte si vous utilisez un Apple IIe ou IIc ; vous pouvez donc l'utiliser sur une machine ou sur l'autre à condition d'avoir réalisé le circuit correspondant.*

### Création d'un nouveau carnet

Cette commande permet d'effacer totalement un carnet pour en créer un nouveau. A utiliser avec prudence.

### Description technique

#### Les caractères spéciaux avec la carte 80 colonnes

Le programme utilise largement les possibilités des divers caractères de contrôle qu'autorise la carte 80 colonnes. Les caractères de contrôle qui suivent sont utilisables avec la fonction PRINT dès que l'on a fait PR#3.

#### Tabulation :

- BS\$ = CHR\$(8) : curseur un cran à gauche ;
- FS\$ = CHR\$(28) : curseur un cran à droite ;
- VT\$ = CHR\$(31) : curseur une ligne au dessus (même position horizontale) ;
- LF\$ = CHR\$(10) : curseur une ligne au dessous (idem) .

#### Effacement :

- CHR\$(26) : effacement de la ligne ;
- CHR\$(11) : effacement jusqu'en bas de l'écran ;
- CHR\$(29) : effacement jusqu'en fin de ligne.

#### Mode d'affichage :

- CHR\$(18) : affichage en 80 colonnes ;
- CHR\$(17) : affichage en 40 colonnes (mais gestion par la carte 80 colonnes) ;
- CHR\$(21) : retour à l'affichage normal en 40 colonnes.

#### Les icônes souris :

- IS = CHR\$(15) : mode inverse ;
- N\$ = CHR\$(14) : Mode normal ;
- IC\$ = CHR\$(27) + IS : mode icône puis mode inverse ;
- NI\$ = CHR\$(24) + N\$ : fin du mode icône puis mode normal.

Ces caractères permettent de modifier plusieurs fois le mode d'affichage pendant le même PRINT. Sur les anciens //e, les icônes n'étant pas disponibles, IC\$ et NI\$ n'ont aucun effet.

Exemple d'utilisation : l'affichage de la variable CR\$ = IC\$ + "M" + NI\$ donnera l'icône  $\downarrow$ .

*Note : pour que le programme puisse tourner sur un //e non mis à niveau, un test est effectué au début afin de déterminer si l'affichage utilisera ou non les possibilités graphiques des icônes souris.*

### Les programmes en langage machine

Ces programmes, au nombre de quatre, sont chargés par le programme en Basic sous forme de DATA qui sont 'POKés' en page 3. Ils sont tous relogeables, peu importe donc leur ordre, pourvu qu'ils ne se chevauchent pas. Les deux premiers sont déjà parus dans Pom's, donc seuls les sources des deux derniers sont listés.

GER est la routine classique qui

accompagne un traitement d'erreur, il est inutile de la présenter.

**COMP** assure la même fonction que les programmes 'COMPOSEUR' déjà cités. Il est par contre relogeable et fonctionne aussi bien sur le //e ou le //c.

**IN** est un input généralisé qui permet d'accepter tous les caractères. Il est utilisé pour lire le fichier 'ADRESSE' sans avoir de problème avec les virgules ou autres.

**G80** est un programme d'entrée de caractères en 80 colonnes. Il est surtout utile pour éviter les inconvénients de la fonction GET en 80 colonnes sur un Apple //e, en particulier l'interception de la touche 'Escape' ou la confusion entre la flèche droite et le caractère espace (ces inconvénients n'existent pas sur un //c). De plus, il permet de redéfinir le curseur.

## Les fichiers

**CARNET**  
programme  
**ADRESSES** et **INDEX**  
fichiers de données créés à l'initialisation  
**T.G80** et **T.IN**  
sources de G80 et de IN  
(ProCODE ou Big Mac)



## Programme 'CARNET'

NB : Les 'puces' symbolisent les espaces pour simplifier la saisie.

```

0 REM CARNET D'ADRESSES (C) BORIS FIVOLEW
  SKI 1986
1 REM =====
10 ONERR GOTO 20000
20 GOTO 10000
99 :
100 REM Entrée d'un mot
101 REM -----
110 MA = 1: GOTO 130
120 MA = 0: GOTO 130
130 I = 1:C$ = MID$(C$,1,LM):P = PEEK (
  1403): PRINT C$;: POKE 1403,P
140 CALL G80:K = PEEK (- 16384):K = K -
  128 * (K > 127)
150 IF K = 27 THEN : POP : RETURN
160 IF K = 10 OR K = 11 OR K = 13 GOTO 26
  0
170 IF (K = 8 OR K = 127) AND I > 1 THEN
  I = I - 1: PRINT BS$;: IF K = 8 GOTO 1
  40
180 IF K = 127 THEN P = PEEK (1403):K$ =
  MID$(C$,I + 1,LM + 1 - I):C$ = MID
  $(C$,1,I - 1) + K$: PRINT K$;" ";: PO
  KE 1403,P: GOTO 140
190 IF K = 21 AND I < LM THEN I = I + 1:
  PRINT FS$;: GOTO 140
200 IF MA AND (96 < K AND K < 123) THEN K
  = K - 32
210 IF MA AND ((K < 65 OR K > 90) AND K <
  > 32 AND K < > 45 AND K < > 39 AND
  K < > 46) THEN PRINT BL$;: GOTO 140
220 IF K < 32 THEN PRINT BL$;: GOTO 140
230 K$ = CHR$(K):C$ = MID$(C$ + BB$,1,
  I - 1) + K$ + MID$(C$,I + 1,LM - I)
240 I = I + 1: IF I > LM THEN K$ = K$ + BL
  $ + BS$:I = I - 1
250 PRINT K$;: GOTO 140
260 L = LEN (C$): IF L = 0 GOTO 290
270 FOR I = L TO 1 STEP - 1: IF MID$(C
  $,I,1) < > " " THEN L = I:I = 1
280 NEXT I
290 C$ = MID$(C$,1,L): RETURN
399 :
499 :
500 REM Recherche du code d'un nom entre
  deux limites
501 REM -----
510 N = 0:J = 0: FOR I = 1 TO NF
520 J = J + 1:Y0 = X(J,0):Y1 = X(J,1): IF
  Y0 = 0 GOTO 520
530 IF (Z0 < Y0 OR Z0 = Y0 AND Z1 < = Y1
  ) AND (Y0 < X0 OR Y0 = X0 AND Y1 < X1)
  THEN N = N + 1:N(N) = J
540 NEXT I
550 RETURN
599 :
600 REM Codage d'un nom sur 12 lettres
601 REM -----
610 L = 0:J = 0: FOR I = 1 TO 12
620 IF I = 7 THEN X0 = J:J = 0
630 L = L + 1: REM Ignore blanc,tiret et a
  postrophe
640 K$ = MID$(C$,L,1): IF K$ = "" THEN K
  $ = "à"
650 IF ASC (K$) - 64 < 0 GOTO 630
660 J = J * 27 + ASC (K$) - 64
670 NEXT I:X1 = J
680 RETURN
699 :
700 REM Lecture d'une fiche
701 REM -----
710 CALL IN,NO$: CALL IN,PR$: CALL IN,AD$
  : CALL IN,VI$: CALL IN,TE$: CALL IN,RE
  $: PRINT D$
720 PRINT TM$;
730 PRINT "NOM: ";NO$;: POKE 36,41: PRINT
  "Prénom: ";PR$: PRINT "Adresse: ";AD$
  : PRINT "Ville: ";VI$: POKE 36,46: PR
  INT "Téléphone: ";TE$: PRINT "Renseign
  ement: ";RE$
740 RETURN
799 :
999 :
1000 REM Insertion d'une fiche
1001 REM =====
1010 NO$ = "":PR$ = "":AD$ = "":VI$ = "":T
  E$ = "":RE$ = ""

```

```

1020 REM Recherche point d'insertion
1021 REM -----
1030 IF PF = MF + 1 GOTO 1070
1040 FOR I = PF TO MF
1050 IF X(I,0) = 0 THEN PF = I: GOTO 1080
1060 NEXT I:PF = MF + 1
1070 PRINT : PRINT BL$;"Attention le fichier est plein, vous ne pouvez plus insérer de fiche! ";: GET K$: RETURN
1080 FF = PF
1090 P = PEEK (37): VTAB 3: HTAB 26: PRINT "Nombre de fiche(s): ";NF: VTAB P + 1
1100 REM Entrée d'une fiche
1101 REM -----
1110 KK = 1: GOTO 1170
1120 IF K = 11 GOTO 1150
1130 IF KK < > 1 AND KK < > 4 THEN PRINT
1140 KK = KK + 1: GOTO 1170
1150 KK = KK - 1: IF KK = 0 THEN KK = 1: PRINT BL$;
1160 IF KK < > 1 AND KK < > 4 THEN PRINT VT$;
1170 HTAB 2: PRINT BS$;
1180 ON KK GOTO 1510,1540,1550,1560,1570,1580
1190 PRINT TM$;VT$
1200 REM Enregistrement d'une fiche
1201 REM -----
1210 PRINT D$"OPEN INDEX,L20"
1220 PRINT D$"OPEN ADRESSES,L256"
1230 PRINT D$"WRITE INDEX,R"FF
1240 PRINT X0: PRINT X1
1250 IF FF < > PF GOTO 1270
1260 PRINT D$"WRITE INDEX,R0": PRINT NF + 1:NF = NF + 1
1270 PRINT D$"WRITE ADRESSES,R"FF
1280 PRINT NO$: PRINT PR$: PRINT AD$: PRINT VI$: PRINT TE$: PRINT RE$
1290 PRINT D$"CLOSE"
1300 X(FF,0) = X0:X(FF,1) = X1
1310 IF FF = PF GOTO 1000: REM Insertion en continu
1320 RETURN
1500 REM Rubriques
1501 REM -----
1510 PRINT "NOM: ";:LM = 35:C$ = NO$: GOSUB 110:NO$ = C$
1520 GOSUB 600: IF X0 = 0 THEN RETURN
1530 GOTO 1120
1540 POKE 36,41: PRINT "Prénom: ";:LM = 30:C$ = PR$: GOSUB 120:PR$ = C$: GOTO 1120
1550 PRINT "Adresse: ";:LM = 70:C$ = AD$: GOSUB 120:AD$ = C$: GOTO 1120
1560 PRINT "Ville: ";:LM = 38:C$ = VI$: GOSUB 120:VI$ = C$: GOTO 1120
1570 POKE 36,46: PRINT "Téléphone: ";:LM = 22:C$ = TE$: GOSUB 120:TE$ = C$: GOTO 1120
1580 LM = 250 - LEN (NO$) - LEN (PR$) - LEN (AD$) - LEN (VI$) - LEN (TE$):
LM = 64 * (LM > 64) + LM * (LM < 65):
IF LEN (RE$) > LM THEN PRINT CHR$ (29);
1590 PRINT "Renseignement: ";:C$ = RE$: GOSUB 120:RE$ = C$: GOTO 1120
1999 :
2000 REM Recherche et impression d'une fiche
2001 REM -----
2010 GOSUB 2500
2020 IF K = 27 OR N = 0 THEN RETURN
2030 PRINT : PRINT "Impression de cette fiche (O/N) ? ";: CALL G80: PRINT
2040 K$ = CHR$ ( PEEK ( - 16384)): IF K$ < > "O" AND K$ < > "o" THEN RETURN
2050 PRINT D$;"PR#1":I = 10
2060 PRINT SPC( I);"M. ";PR$;" ";NO$
2070 PRINT SPC( I);AD$: PRINT SPC( I);VI$
2080 PRINT D$"PR#3"
2090 RETURN
2210 PRINT VT$;VT$
2499 :
2500 REM Recherche des fiches
2501 REM -----
2510 N = 0: PRINT "NOM: ";:LM = 35:MA = 1:C$ = NO$: GOSUB 110: PRINT
2520 GOSUB 600: IF X0 = 0 THEN PRINT "Nom incorrect!": GOTO 2510
2530 Z0 = X0:Z1 = X1: IF RIGHT$ (C$,1) < > "." THEN X1 = X1 + 1:C$ = "portant le nom: " + C$: GOTO 2560
2540 C$ = C$ + "[": GOSUB 600: REM Astuce: Code ascii suivant celui de Z
2550 C$ = "commençant par: " + LEFT$ (C$, LEN (C$) - 1)
2560 GOSUB 500: IF N = 0 GOTO 2670
2570 VTAB 3: PRINT "Il y a ";N;" fiche(s) ";C$: HOME
2580 PRINT D$"OPEN ADRESSES,L256"
2590 FOR I = 1 TO N
2600 PRINT D$"READ ADRESSES,R"N(I)
2610 GOSUB 700
2620 PRINT : PRINT "Est-ce la fiche que vous recherchez (O/N) ? ";
2630 CALL G80:K = PEEK ( - 16384):K$ = CHR$ (K): IF K = 27 GOTO 2680
2640 IF K$ = "O" OR K$ = "o" GOTO 2680
2650 IF K$ < > "N" AND K$ < > "n" GOTO 2630
2660 PRINT VT$;VT$: NEXT I
2670 PRINT : PRINT "La fiche que vous recherchez n'est pas dans ce fichier.": PRINT " Appuyez sur une touche pour revenir au menu. ";: CALL G80:N = 0
2680 PRINT : PRINT D$"CLOSE"
2690 RETURN
2999 :
3000 REM Modification d'une fiche
3001 REM -----
3010 GOSUB 2500
3020 IF K = 27 OR N = 0 THEN RETURN

```

```

3030 FF = N(I)
3040 HOME : GOSUB 720
3050 VTAB 5: GOTO 1100
3999 :
4000 REM Suppression d'une fiche
4001 REM =====
4010 GOSUB 2500
4020 IF K = 27 OR N = 0 THEN RETURN
4030 PRINT : PRINT "D'accord pour supprimer cette fiche (O/N) ? " : CALL G80: PRINT
4040 K$ = CHR$ ( PEEK ( - 16384)): IF K$ < > "O" AND K$ < > "o" THEN RETURN

4050 PRINT D$"OPEN INDEX,L20"
4060 PRINT D$"WRITE INDEX,R"N(I)
4070 PRINT "0": PRINT "0"
4080 PRINT D$"WRITE INDEX,R0"
4090 PRINT NF - 1
4100 PRINT D$"CLOSE"
4110 X(N(I),0) = 0:NF = NF - 1:PF = N(I)
4120 RETURN
4999 :
5000 REM Liste à partir d'un nom
5001 REM =====
5100 REM Initialisation
5101 REM -----
5110 PRINT "Nom d'origine: ";LM = 35:C$ = NO$: GOSUB 110: PRINT : GOSUB 600
5120 K = 0:FF = 0: FOR I = 1 TO NF
5130 K = K + 1:Y0 = X(K,0):Y1 = X(K,1)
5140 N(K) = 0: IF Y0 = 0 GOTO 5130
5150 IF Y0 > X0 OR Y0 = X0 AND Y1 > = X1 THEN FF = FF + 1:N(K) = 1
5160 NEXT I: IF FF = 0 THEN PRINT : PRINT "Il n'y a aucune fiche après ce nom !!!": GOTO 5300
5170 C$ = "de " + C$: IF X0 = 0 THEN C$ = "du début"
5180 VTAB 3: PRINT "Liste de ";FF;" fiche(s) à partir ";C$: HOME
5190 PRINT D$"OPEN ADRESSES,L256"
5200 REM Lecture
5201 REM -----
5210 FOR I = FF TO 1 STEP - 1
5220 X0 = 987654321:K = 0: FOR J = 1 TO I
5230 K = K + 1: IF N(K) = 0 GOTO 5230
5240 Y0 = X(K,0):Y1 = X(K,1)
5250 IF Y0 < X0 OR Y0 = X0 AND Y1 < X1 THEN X0 = Y0:X1 = Y1:N = K
5260 NEXT J: PRINT D$"READ ADRESSES,R"N: GOSUB 700
5270 POKE - 16364,0: IF PEEK ( - 16384) = 32 THEN CALL G80
5280 IF PEEK ( - 16384) = 27 GOTO 5500
5290 N(N) = 0: NEXT I: PRINT : PRINT "Dernière fiche. ";
5300 PRINT " Appuyez sur une touche pour revenir au menu: "; CALL G80
5500 REM Fin de la liste
5501 REM -----
5510 PRINT : PRINT D$"CLOSE"
5520 RETURN

```

```

5999 :
6000 REM Composition automatique
6001 REM =====
6010 GOSUB 2500
6020 IF K = 27 OR N = 0 THEN RETURN
6030 PRINT VT$: CHR$ (29)
6040 L = LEN (TE$)
6050 IF L = 0 THEN PRINT "Pas de numéro de téléphone.": GOTO 6500
6060 PRINT "Je compose le numéro: "; REM Prise de ligne
6070 IF IC THEN POKE 49322,8
6080 IF NOT IC THEN POKE 49241,0
6090 FOR J = 1 TO 1000: NEXT
6100 REM Composition
6110 FOR I = 1 TO L
6120 K$ = MID$ (TE$,I,1):K = ASC (K$): PRINT K$;
6130 IF 47 < K AND K < 58 THEN POKE 6,K - 48 + 10 * (K = 48): CALL COMP:K$ = "-": REM Ajoute un délai d'1 seconde après chaque chiffre
6140 IF K$ = "-" THEN FOR J = 1 TO 1000: NEXT : REM Environ 1 seconde
6150 IF K$ = "+" THEN FOR J = 1 TO 10000: NEXT : REM Environ 10 secondes
6160 IF K$ < > "(" GOTO 6290
6170 P = PEEK (1403)
6180 PRINT : PRINT "Indicatif (O/N) ? ";
6190 CALL G80:K = PEEK ( - 16384):K$ = CHR$ (K)
6200 REM Indicatif ?
6210 IF K = 27 GOTO 6550
6220 IF K$ = "O" OR K$ = "o" GOTO 6280
6230 IF K$ < > "N" AND K$ < > "n" GOTO 6190
6240 FOR J = I + 1 TO L
6250 K$ = MID$ (TE$,J,1):K = ASC (K$)
6260 IF K$ = ")" THEN J = L:I = I - 1
6270 I = I + 1: NEXT J
6280 POKE 1403,P: PRINT VT$: CHR$ (11);
6290 NEXT I
6400 REM Décrochage
6410 PRINT : PRINT CHR$ (7)
6420 PRINT "Débranchez le combiné avant .. secondes";
6430 FOR I = 10 TO 1 STEP - 1
6440 HTAB 28: PRINT SPC( I < 10);I;
6450 FOR J = 1 TO 1000: NEXT
6460 NEXT I
6470 PRINT VT$
6480 IF IC THEN POKE 49322,0
6490 IF NOT IC THEN POKE 49240,0
6500 REM Fin
6510 PRINT "Appuyez sur une touche pour revenir au menu principal. "; CALL G80
6520 RETURN
6550 REM Escape
6560 IF IC THEN POKE 49322,0
6570 IF NOT IC THEN POKE 49240,0
6580 RETURN
6999 :
7000 REM Créer un nouveau carnet

```

```

7001 REM =====
7010 HOME : PRINT " Attention en créant
un nouveau carnet vous effacer le pré
cédent s'il existe.": PRINT " Voulez
-vous vraiment créer un nouveau carne
t (O/N): ";
7020 CALL G80:K$ = CHR$ ( PEEK ( - 16384
))
7030 IF K$ < > "O" AND K$ < > "o" THEN
RETURN
7040 PRINT
7100 REM Destruction éventuelle
7110 IF LI = 7130 GOTO 7140
7120 IF LI = 7140 GOTO 7150
7130 PRINT D$"DELETE INDEX"
7140 PRINT D$"DELETE ADRESSES"
7150 PRINT D$"OPEN INDEX,L20"
7160 PRINT D$"WRITE INDEX,R0"
7170 PRINT "0": REM Zéro pas "o"
7180 PRINT D$"OPEN ADRESSES,L256"
7190 PRINT D$"CLOSE"
7200 RUN
7999 :
9999 :
10000 REM *****
10001 REM ** DEBUT DU PROGRAMME **
10002 REM *****
10010 TEXT : HOME :D$ = CHR$ (4): PRINT
D$;"PR#3": REM 80 colonnes
10020 PRINT : PRINT "POM'S présente:"
10030 POKE 32,20: PRINT : PRINT
10035 PRINT "*****
*****"
10040 PRINT "*****
*****"
10045 PRINT "*****
*****"
10050 PRINT "*****
*****"
10055 PRINT "*****
*****"
10060 POKE 32,9: PRINT : PRINT
10065 PRINT "*****
*****"
10070 PRINT "*****
*****"
10075 PRINT "*****
*****"
10080 PRINT "*****
*****"
10085 PRINT "*****
*****"
10090 POKE 32,0: PRINT : PRINT
10095 PRINT "Veuillez patienter, merci."
10099 :
10100 REM Variables principales & caractè
res spéciaux
10101 REM =====
10110 I = J:X0 = X1:Y0 = Y1:Z0 = Z1:MF = 5
00: DIM X(MF,1),N(MF)
10120 BS$ = CHR$ (8):FS$ = CHR$ (28):VT$
= CHR$ (31):BL$ = CHR$ (7)
10130 I$ = CHR$ (15):N$ = CHR$ (14):IC$
= CHR$ (27) + I$:NI$ = N$ + CHR$ (
24)
10140 IC = PEEK (64448):IC = (IC = 0) + (
IC = 224)
10150 BB$ = "
":BB$ = BB$ + BB$
10160 TB$ = "
":TB$ = TB$ + TB$
10170 TM$ = "SSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SSSSSSSSSSSS":TM$ = IC$ + TM$ + TM$ +
NI$ + CHR$ (26)
10180 IF IC GOTO 10200
10190 TM$ = "-----
-----":TM$ = TM$ + TM$
10199 :
10200 REM Sous programmes en binaire (re
logeable)
10201 REM =====
10210 GER = 768: REM Gestion d'erreurs
10220 DATA 104,168,104,166,223,154,72,152
,72,96
10230 IN = GER + 10: REM Input généralisé
10240 DATA 32,190,222,32,227,223,32,44,21
3,162,255,232,189,0,2,208,250,138,16
0,0,145,131,32,82,228,72,152,160,2,1
45,131,136,138,145,131,162,0,200,104
,32,226,229,96
10250 G80 = IN + 43: REM GET clignotant en
80 colonnes
10260 DATA 165,37,32,193,251,165,36,205,1
23,4,208,3,173,123,5,74,168,141,84,1
92,176,3,141,85,192,177,40,170,138,2
09,40,208,2,169
10270 DATA 223,145,40,169,0,24,72,44,0,19
2,48,11,233,1,208,247,104,233,1,208,
241,240,227,104,138,145,40,141,16,19
2,141,84,192,96
10280 COMP = G80 + 68: REM Composition a
utomatique
10285 DATA 166,6,160,0,169,160,140,88,192
,32,168,252,160,8,169,108,140,89,192
,32,168,252,202,208,233,96
10290 FIN = COMP + 25: FOR I = GER TO FIN:
READ J: POKE I,J: NEXT I
10295 IF IC THEN POKE COMP + 7,170: POKE
COMP + 17,170
10299 :
10300 REM Menu
10301 REM =====
10310 DATA Insertion d'une fiche
10320 DATA Recherche d'une fiche
10330 DATA Modification d'une fiche
10340 DATA Suppression d'une fiche
10350 DATA Liste alphabétique
10360 DATA Composition automatique
10370 DATA Nouveau carnet
10380 DATA Quitter le programme
10390 NC = 7: DIM CHOIX$(NC): FOR I = 0 TO
NC: READ CHOIX$(I): NEXT I
10399 :
10400 REM Lecture du fichier index
10401 REM -----
10410 PRINT D$"UNLOCK INDEX"

```



```

10420 PRINT D$"UNLOCK ADRESSES"
10430 PRINT D$"OPEN INDEX,L20": PRINT D$"
      READ INDEX,R0": INPUT NF
10440 IF NF = 0 GOTO 10490
10450 K = 0: FOR I = 1 TO NF
10460 K = K + 1: PRINT D$"READ INDEX,R"K:
      INPUT X(K,0),X(K,1)
10470 IF X(K,0) = 0 AND X(K,1) = 0 GOTO 1
      0460
10480 NEXT I
10490 PRINT D$"CLOSE":PF = 1
10499 :
10500 REM Affichage du menu
10501 REM -----
10510 TEXT : HOME :HP = 26:VP = 11 - NC:O
      LD = PT + 1
10520 PRINT TB$;: PRINT I$;BB$: VTAB 2: H
      TAB 20: PRINT "CARNET D'ADRESSES (C
      ) BORIS FIVOLEWSKI 1986";N$
10530 VTAB VP + 1: FOR I = 0 TO NC - 1: H
      TAB HP - 2: PRINT I + 1;". ";CHOIX$(
      I): PRINT : NEXT I
10540 HTAB HP - 2: PRINT "0. ";CHOIX$(NC)
10550 VTAB 23: PRINT TM$;
10560 IF IC GOTO 10590
10570 PRINT "Tapez un nombre ou utilisez
      les flèches pour choisir une option.
      Puis RETURN.";
10580 GOTO 10600
10590 PRINT "Tapez un nombre ou pressez
      ";IC$;"J";NI$;" ou ";IC$;"K";NI$;" p
      our sélectionner une option.Puis pre
      sser ";IC$;"M";NI$;
10600 REM *** Entrée du choix ***
10610 IF PT = OLD THEN GOTO 10630
10620 VTAB OLD * 2 + VP: HTAB HP: PRINT
      SPC( 26): PRINT : HTAB HP: PRINT " "
      ;CHOIX$(OLD);" ": HTAB HP: VTAB PT *
      2 + VP: PRINT LEFT$(TB$, LEN (CHO
      IX$(PT)) + 2): HTAB HP: PRINT I$;" "
      ;CHOIX$(PT);" ";NI$
10630 WAIT - 16384,128: POKE - 16368,0:
      K = PEEK ( - 16384)
10640 OLD = PT
10650 IF K = 10 THEN PT = PT + 1 - (NC +
      1) * (PT = NC)
10660 IF K = 11 THEN PT = PT - 1 + (NC +
      1) * (PT = 0)
10670 IF 47 < K AND K < 49 + NC THEN PT =
      K - 49: IF PT < 0 THEN PT = NC
10690 IF K < > 13 THEN GOTO 10600
10700 REM Exécution de la commande
10701 REM -----
10710 IF PT = NC GOTO 11000
10720 IF NF = 0 AND PT < > 0 THEN VTAB
      22: PRINT I$;" Commande impossible:
      fichier vide. ";N$;:OLD = PT:PT = 0:
      GET K$: PRINT CHR$( 26): GOTO 1060
      0
10730 HOME : PRINT I$;BB$: VTAB 1: HTAB 2
      : PRINT CHOIX$(PT);: POKE 36,51: PRI
      NT "ESC : Retour au menu général";N$
      : POKE 34,3: PRINT : PRINT
10740 ON PT + 1 GOSUB 1000,2000,3000,4000
      ,5000,6000,7000
10750 GOTO 10500
11000 HOME : PRINT CHR$( 21): END
19999 :
20000 REM Traitement des erreurs
20001 REM -----
20100 ER = PEEK (222):LI = PEEK (218) +
      PEEK (219) * 256
20200 :
20210 CALL GER
20220 IF LI = 20230 GOTO 20240
20230 PRINT D$;"CLOSE".
20240 PRINT CHR$( 7)
20250 IF ER = 255 THEN PRINT "BREAK EN L
      IGNE ";LI: END
20300 REM Erreurs initiales
20301 REM -----
20310 IF (ER = 6 OR ER = 7) AND LI = 1041
      0 GOTO 10420
20320 IF (ER = 6 OR ER = 7) AND LI = 1042
      0 GOTO 7100
20330 IF ER = 5 AND LI = 10430 GOTO 7100
20340 IF (ER = 6 OR ER = 7) AND (LI > 700
      0 AND LI < 8000) GOTO 7100
20350 IF ER = 4 AND LI = 10410 GOTO 10430
20390 IF LI > 10000 AND LI < 10500 THEN
      PRINT "Erreur pendant l'initialisati
      on. Recharger le programme.": END
20400 REM Autres erreurs
20401 REM -----
20410 IF ER = 6 OR ER = 7 OR ER = 8 THEN
      PRINT "Carnet d'adresses non trouvé
      .": PRINT "Mettez la bonne disquette
      ou créez un nouveau carnet (option
      7)"
20420 IF ER = 9 THEN PRINT "Disque plein
      .": PRINT "Créez un nouveau carnet s
      ur une autre disquette": GOTO 20500
20430 IF ER = 17 THEN PRINT "Répertoire
      plein.": PRINT "Utilisez une autre d
      isquette": GOTO 20500
20440 IF ER = 10 THEN PRINT "Fichiers ve
      rrouillés."
20450 IF ER = 4 THEN PRINT "Disquette pr
      otégée."
20460 IF ER = 4 OR ER = 10 THEN PRINT "
      Vous ne pouvez pas modifier les fich
      es": GOTO 20500
20470 IF ER = 5 THEN PRINT " Fichiers dé
      tériorés ou disquettes échangées": G
      OTO 20500
20480 PRINT "erreur. ";
20500 PRINT "Pressez une touche pour reve
      nir au menu";: CALL G80
20600 GOTO 10500

```

## Récapitulation

Les quatre fichiers binaires utilisés se trouvent sous forme de DATAs dans le programme Basic.

## Source 'IN' Assembleur Procode ou BigMac

```

1
2      ORG $0300
3
4  VARPNT  EQU $83
5  IN      EQU $0200
6  INLIN   EQU $D52C
7  CHKCLS  EQU $DEB8
8  CHKOPN  EQU $DEBB
9  PTRGET  EQU $DFE3
10 GETSPA  EQU $E452
11 MOVSTR  EQU $E5E2
12
13 *** Début du code ***
14
15      JSR  CHKOPN
16      JSR  PTRGET
17      JSR  CHKCLS
18      JSR  INLIN
19      LDX  #$FF
20 BCL    INX
21      LDA  IN,X
22      BNE  BCL
23      TXA
24      LDY  #$00
25      STA  (VARPNT),Y
26      JSR  GETSPA
27      PHA
28      TYA
29      LDY  #$02
30      STA  (VARPNT),Y
    
```

```

31      DEY
32      TXA
33      STA  (VARPNT),Y
34      LDX  #$00
35      INY
36      PLA
37      JSR  MOVSTR
38      RTS
    
```

## Source 'T.G80' Assembleur Procode ou BigMac

```

1 *      GET
2
3      ORG  $380
4
5  CH    =    $24
6  CV    =    $25
7  BASL  =    $28
8  OLDCH =    $47B
9  OURCH =    $57B
10
11 KED    =    $C000
12 KBDSTRB = $C010
13 TXTPAGE1 = $C054
14 TXTPAGE2 = $C055
15
16 BASCALC = $FBC1
17
18      LDA  CV
19      JSR  BASCALC
20      LDA  CH
21      CMP  OLDCH
    
```

```

22      BNE  old
23      LDA  OURCH
24 old   LSR
25      TAY
26      STA  TXTPAGE1
27      BCS  page
28      STA  TXTPAGE2
29 page  EQU  *
30
31      LDA  (BASL),Y
32      TAX
33 boucle TXA
34      CMP  (BASL),Y
35      BNE  curseur
36      LDA  #"_ "
37 curseur STA (BASL),Y
38
39      LDA  #0
40      CLC
41 wait0  PHA
42 wait1  BIT  KBD
43      BMI  touche
44      SBC  #1
45      BNE  wait1
46      PLA
47      SBC  #1
48      BNE  wait0
49      BEQ  boucle
50
51 touche PLA
52      TXA
53      STA  (BASL),Y
54      STA  KBDSTRB
55      STA  TXTPAGE1
56      RTS
57      DS   $3D0-*
    
```

# Éjectez les disquettes 3'5

Bruno Fémarit

Bien qu'un bouton permette de le faire manuellement, il peut-être intéressant d'éjecter une disquette de l'Unidisk 3.5 par programme à la manière d'un Macintosh. La petite routine ci-contre permet de faire l'opération.

La commande d'éjection n'étant pas gérée par ProDOS, il faut faire appel directement à la carte interface. Le programme de démonstration donne les quelques lignes à inclure dans votre programme Basic.



## Démonstration

```

10  DATA 32,0,0,4,10,3,141,17,3,
    96,3,0,15,3,4,0,0
20  FOR I = 768 TO 784: READ J:
    POKE I,J: NEXT
30  INPUT "SLOT: ";S: INPUT "DRIVE: ";D:S = (S+192) * 256
40  IF PEEK (S + 7) THEN PRINT
    "CE N'EST PAS UN UNIDISK": E
    ND
50  POKE 769, PEEK (S + 255) + 3
    : POKE 770,S / 256: POKE 779
    ,D
60  CALL 768:ER = PEEK (785)
70  IF ER = 40 THEN PRINT "DRIVE
    NON CONNECTE": END
80  IF ER THEN PRINT "ERREUR"
    
```

## Source Éjection

```

                                ORG  $300
                                OBJ  $300
ENTRY  EQU  0
DEVICE EQU  1
20 00 00      JSR  ENTRY
04          DFB  4
0A 03          DA  PARLIST
8D 11 03      STA  ERROR
60          RTS
03          PARLIST DFB  3
00          DFB  DEVICE
0F 03          DW  CTRLST
04          DFB  4
00 00          CTRLST DW  0
ERROR      EQU  *
    
```

# L'apprenti-FORTHier

Fej Nissuorgsky

L'"Apprenti-FORTHier" (APF) est un système FORTH (interpréteur+compilateur) écrit en Applesoft sous DOS 3.3 et prévu pour fonctionner en 80 colonnes. Il possède de plus l'intérêt de visualiser le contenu de la pile de DATA et l'état du dictionnaire, et permet donc de saisir le fonctionnement du langage Forth

## Le langage Forth

Il fut inventé à la fin des années 60 et implémenté sur IBM 1130 par Charles H. Moore. Ses premières applications furent consacrées à l'astronomie. Forth est également un outil privilégié dans le développement de la ludotique.

Forth possède un grand nombre d'avantages :

- facile à utiliser grâce à une syntaxe simple et peu contraignante ; l'utilisation crée son propre vocabulaire de dialogue avec la machine ;
- efficace car semi-compilé ; il est interprété et permet donc une programmation agréable ; les définitions du vocabulaire sont compilées d'où une exécution très rapide (sauf pour APF car interpréteur et compilateur sont écrits et compilés en Applesoft, ce qui est le pendant de son caractère didactique) ;
- puissant car c'est un langage de haut niveau qui permet cependant également de manipuler les données au niveau des octets en mémoire.

Forth ne fonctionne pas avec des instructions (comme le Basic, Pascal ou les assembleurs). Au contraire, l'utilisateur peut enseigner son langage à la machine en

créant des mots, ce vocabulaire étant placé dans un dictionnaire. Il est très simple de définir un mot, qui ne sera reconnu que s'il ne fait appel qu'à des mots déjà connus :

: Exemple 1 2 + . ;

Exemple affiche le résultat de 1+2 (à noter que, en Forth, tous les mots doivent être séparés par un espace). Bien entendu, un mot peut être effacé avec l'instruction FORGET

Les données manipulées en Forth peuvent être des variables ou tableaux des constantes, ou encore des constantes immédiates.

Forth est construit autour de deux

pires : la pile de données ( la seule visualuée par APF) et la pile de retour, toutes deux bâties selon la structure LIFO (Last In First Out).

Forth fonctionne avec la notation polonaise inversée, comme certaines calculatrices :

1 2 + 3 \* donne 9.

La pratique d'APF permet de visualiser la pile tout au cours des opérations. Conventionnellement, on représente entre parenthèses et séparés par --- les paramètres sur la pile avant et après exécution du mot :

+ ( a b --- a + b )

L'approche interprétée est représentée par le dialogue au clavier :

## Help !

*Voici la marche à suivre pour utiliser ce programme d'approche du langage Forth.*

- 1 Vous êtes le bienheureux possesseur de la disquette d'accompagnement de la revue ? Passez à l'étape 7.
- 2 Saisir et sauvegarder le fichier binaire DICO qui contient les ordres Forth de base, le noyau.
- 3 Saisir CREDEMO, le sauvegarder, faire RUN CREDEMO pour créer le fichier TEXT DEMO.APF.
- 4 Saisir CREMESS, le sauvegarder, faire RUN CREMESS pour créer le fichier MESSAPF.
- 5 Saisir et sauvegarder GARDE.APF et APF 22-4.
- 6 Vous devez disposer sur la même disquette des fichiers suivants (ce sont les seuls nécessaires) :

```
A GARDE.APF
A APF 22-4
B DICO
T DEMOAPF
T MESSAPF
```

- 7 Faites RUN GARDE.APF.

1 2 + 3 \*

alors que l'approche compilée est représentée par la définition de nouveaux mots :

: CARRE DUP \* ;

## Les programmes fournis

### CREMESS

crée et modifie le fichier direct MESS APF. L'utilisateur n'a pas en principe à travailler avec, sauf s'il veut adapter le programme APF.

### CREDEMO

crée et modifie le fichier DEMO APF qui contient des exemples de séquences. Outre l'aspect démonstratif, on peut s'en servir comme base de petits "programmes" à charger.

### GARDE.APF

affiche la page de garde et lance APF.

## Comment se servir d'APF

Il y a deux modes :

- normal
- pas à pas (pause après chaque mot d'une même ligne.

## Vocabulaire d'APF

DEMO charge un des exemples du fichier DEMO.APF ;  
 INIT réinitialise le Forth (doit être en début de ligne) ;  
 RAZ réinitialise les piles ;  
 CLS efface l'écran ;  
 BELL émet un 'beep' ;  
 LISTE MOTS liste les dictionnaires ;  
 MODE commute les modes normal et pas à pas ;  
 QUITTE arrêt du programme ;  
 AIDE affiche un écran d'aide ;  
 CARRAY tableau d'octets (*n* CARRAY nom crée un tableau de *n*+1 octets auquel on peut accéder par C@ et C!).

## Quelques clefs pour l'utilisation du programme

Se reporter à la bibliographie

ORDRE	PILE AVANT	PILE APRES	COMMENTAIRE
*	n1 n2	n3	Multiplication de n1 par n2 Laisse n3 sur la pile
+	n1 n2	n3	Somme de n1 et de n2 Laisse n3 sur la pile
+LOOP	n		Incrémente l'indice de boucle de n, L'exécution reprend au DO correspondant tant que l'indice est inférieur à la limite
-	n1 n2	n3	Soustrait n2 de n1 Laisse n3 sur la pile
.	n		Affiche le nombre n suivi d'un espace
:	"texte"		Affiche du texte jusqu'à "
/	n1 n2	n3	Divise n1 par n2 (division entière) Laisse n3 sur la pile
:	MOT définition		MOT est créé dans le dictionnaire, l'interpréteur compile la définition, chaque ordre de la définition doit exister dans le vocabulaire connu
.			Indique la fin d'une définition
.			Exemple de définition : : CARRE DUP * ;
<	n1 n2	drapeau	Compare n1 à n2. Sur la pile drapeau sera à 1 si n1<n2 et 0 sinon
=	n1 n2	drapeau	Compare n1 à n2. Sur la pile drapeau sera à 1 si n1=n2 et 0 sinon
>	n1 n2	drapeau	Compare n1 à n2. Sur la pile drapeau sera à -1 si n1>n2 et 0 sinon
>	n1 n2	drapeau	Compare n1 à n2. Sur la pile drapeau sera à 1 si n1>n2 et 0 sinon
?	adr		Affiche contenu de la mémoire ADR
!	n adr		Mémoire n à l'adresse adr (n sur 2 octets)
@	adr	n	Empile le contenu de l'adresse adr (n sur 2 octets)
AND	n1 n2	n3	ET logique
C!	octet adr		Mémoire l'octet à l'adresse adr
C@	adr	octet	Empile le contenu de adr
CR			Affiche un retour chariot
DROP	n		Supprime le sommet de la pile
DUP	n	n n	Duplique le sommet de la pile
ELSE			De la forme <Cond> IF <si vrai> ELSE <si faux> THEN
EMIT	n		Affiche le caractère dont le code ASCII est n
FORGET			Supprime un nom du dictionnaire et ceux définis depuis. ex : FORGET CARRE
I		n	Dans les boucles. Empile l'indice ex : DO...I...LOOP"
IF	drapeau		Si drapeau est vrai les mots qui suivent sont exécutés, sinon ceux qui suivent ELSE le sont.
KEY		n	Empile code ASCII de la touche frappée
NOT	drapeau1	drapeau2	Inverse le drapeau
OR	n1 n2	n3	OU logique
OVER	n1 n2	n1 n2 n1	Empile copie de l'avant dernier nombre
ROT	n1 n2 n3	n2 n3 n1	Rotation de la pile (3 cellules)
SWAP	n1 n2	n2 n1	Échange des deux cellules en haut de la pile
THEN			Symbolise la fin d'une condition

Les nombres sont sur deux octets (-32768 à +32767), les mots sont limités à 16 caractères (154 mots maxi) et les phrases à 159 caractères.

Les autres mots sont des classiques du Forth 79 (voir tableau ci-joint)

## Bibliographie

- FIG Forth, distribué par le Forth Interest Group  
 Forth 79  
 TransForth, avec des nombres réels, de Paul Lutus  
 GraForth, avec de puissantes possibilités graphiques, de Paul Lutus

# Liste des mots du dictionnaire de DICO

(1<sup>ère</sup> colonne, puis 2<sup>ème</sup> colonne).

```

+      KEY
-      CLS
*      :
/      ;
<      I
=      DO
>      +LOOP
<>     IF
AND    THEN
OR     ELSE
NOT    BEGIN
DUP    WHILE
DROP   REPEAT
OVER   CONSTANT
CG     VARIABLE
C!     CARRY
ROT    FORGET
SWAP   AIDE
CR     QUITTE
EMIT   MODE
!      RAZ
.      BELL
. "    LISTEMOTS
?      INIT
@      DEMO
    
```

# Fichier DICO (créé par CREDICO)

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par BSAVE DICO,A\$8601,L\$19A

```

8601- 2B 20 00 01 00 00 2D
8608- 20 00 02 86 01 2A 20 00
8610- 03 86 07 2F 20 00 04 86
8618- 0D 3C 20 00 05 86 13 3D
8620- 20 00 06 86 19 3E 20 00
8628- 07 86 1F 3C 3E 20 00 08
8630- 86 25 41 4E 44 20 00 09
8638- 86 2E 4F 52 20 00 0A 86
8640- 32 4E 4F 54 20 00 0B 86
8648- 3A 44 55 50 20 00 0C 86
8650- 41 44 52 4F 50 20 00 0D
8658- 86 49 4F 56 45 52 20 00
8660- 0E 86 51 43 40 20 00 0F
8668- 86 5A 43 21 20 00 10 86
8670- 63 52 4F 54 20 00 11 86
8678- 6A 53 57 41 50 20 00 12
8680- 86 71 43 52 20 00 13 86
8688- 79 45 4D 49 54 20 00 14
8690- 86 82 21 20 00 15 86 89
8698- 2E 20 00 16 86 92 2E 22
86A0- 20 00 17 86 98 3F 20 00
86A8- 18 86 9E 40 20 00 19 86
    
```

```

86B0- A5 4B 45 59 20 00 1A 86
86B8- AB 43 4C 53 20 00 1B 86
86C0- B1 3A 20 00 1C 86 B9 3B
86C8- 20 00 1D 86 C1 49 20 00
86D0- 1E 86 C7 44 4F 20 00 1F
86D8- 86 CD 2B 4C 4F 4F 50 20
86E0- 00 20 86 D3 49 46 20 00
86E8- 21 86 DA 54 48 45 4E 20
86F0- 00 22 86 E4 45 4C 53 45
86F8- 20 00 23 86 EB 42 45 47
8700- 49 4E 20 00 24 86 F4 57
8708- 48 49 4C 45 20 00 25 86
8710- FD 52 45 50 45 41 54 20
8718- 00 26 87 07 43 4F 4E 53
8720- 54 41 4E 54 20 00 27 87
8728- 11 56 41 52 49 41 42 4C
8730- 45 20 00 28 87 1C 43 41
8738- 52 52 41 59 20 00 29 87
8740- 29 46 4F 52 47 45 54 20
8748- 00 2A 87 36 41 49 44 45
8750- 20 00 2B 87 41 51 55 49
8758- 54 54 45 20 00 2C 87 4C
8760- 4D 4F 44 45 20 00 2D 87
8768- 55 52 41 5A 20 00 2E 87
8770- 60 42 45 4C 4C 20 00 2F
8778- 87 69 4C 49 53 54 45 4D
8780- 4F 54 53 20 00 30 87 71
8788- 49 4E 49 54 20 00 31 87
8790- 7A 44 45 4D 4F 20 00 32
8798- 87 88 44
    
```

# Programme 'CREDEMO'

```

1  REM *****
2  REM * Creation des phrases pour la commande DEMO *
3  REM * du programme APF *
4  REM * Fej Nissuorgsky (c) 1985 CHANCELADE 24650 *
5  REM * Le fichier s'appelle DEMOAPF *
6  REM * Ce programme s'appelle CREDEMO *
9  REM *****
99 REM
100 REM Programme Principal
101 REM
110 GOSUB 1000: GOSUB 56000
120 HOME : PRINT "Ce programme cree le fichier DEMOAPF qu
i contient les phrases de demonstration": PRINT "utili
sables par le programme APF (l'APPRENTI FORTHier) sous
la commande DEMO.": PRINT "Les phrases contiennent au
maximum 159 caracteres."
121 PRINT "Le nombre de phrases est limite a 99."
122 PRINT : PRINT "Taper <C> pour CREER une phrase ou <V>
pour VOIR une phrase": PRINT "<G> pour quitter."
130 GET X$:X = ASC (X$): IF X = ASC ("G") THEN END
140 IF X = 67 OR X = 99 THEN GOSUB 500: GOTO 120
141 IF X = 86 OR X = 118 THEN GOSUB 700: GOTO 120
142 GOTO 130
500 REM CREATION
510 HOME : PRINT "CREATION d'une PHRASE": PRINT : INPUT "
Numero : ";NUS:NU = INT ( VAL (NUS)): IF NU < 1 OR NU
> 99 THEN 510
520 GOSUB 31000:P$ = ""
530 FOR I = 1 TO LEN (E$)
540 IF MIDS (E$,I,1) > CHR$ (96) AND MIDS (E$,I,1) <
CHR$ (123) THEN P$ = P$ + CHR$ ( ASC ( MIDS (E$,I,1)
- 32): GOTO 550
545 P$ = P$ + MIDS (E$,I,1)
550 NEXT :E$ = P$
560 PRINT D$"OPENDEMOAPF,L160"
570 FOR I = 1 TO LEN (E$): PRINT D$"WRITEDEMOAPF,R";NU;
B"I - 1: PRINT MIDS (E$,I,1): NEXT
580 PRINT D$"CLOSE": RETURN
699 REM VOIR
700 HOME : PRINT "VOIR une PHRASE": PRINT : INPUT "Numero
    
```

```

: ";NUS:NU = INT ( VAL (NUS)): IF NU < 1 OR NU > 99
THEN 700
705 VTAB 6: POKE 36,0
710 PRINT : PRINT D$"OPENDEMOAPF,L160":VT$ = ""
720 PRINT D$"READDEMOAPF,R";NU
730 GET X$: IF X$ = CHR$ (13) THEN 740
735 VT$ = VT$ + X$: GOTO 730
740 PRINT D$"CLOSE": PRINT : PRINT VT$
750 PRINT : PRINT "Autre taper <RETURN> Arret taper <G>
": & KEY: IF PEEK (768) < > 13 THEN RETURN
760 GOTO 700
999 END
1000 Q = 45: REM IMPLANTATION DE LA ROUTINE &KEY
1010 FOR I = 0 TO Q - 1: READ Z$: POKE 769 + I,Z$: NEXT
1020 POKE 1014,1: POKE 1015,3
1190 RETURN
1200 DATA 44,0,192,16,251,173,0,192,44,16,192,41,127,141,
0,3,201,32,144,14,165,184,24,105,3
1210 DATA 133,184,165,185,105,0,133,185,96,201,13,240,238
,201,8,240,234,76,1,3
30997 REM
30998 REM ENTREE d'une ligne de 159 caracteres
30999 REM
31000 VTAB 2: PRINT : VTAB 19: POKE 36,0: CALL CW: PRINT
PM$: INVERSE : PRINT " "; NORMAL :NC = 0:E$ = "":X
= FRE (0):X = Z
31010 IF NC = 160 OR X = 13 THEN 31050
31015 VTAB 2: PRINT : VTAB 20 + INT (NC / 80): POKE 36,N
C - INT (NC / 80) * 80
31020 & KEY:X = PEEK (768):X$ = CHR$ (X): IF X < 123 AN
D X > 96 THEN X = X - 32:X$ = CHR$ (X)
31030 IF X = 8 THEN GOSUB 31950
31040 IF X < > 8 THEN GOSUB 31900
31041 GOTO 31010
31050 IF X < > 13 THEN 31000
31060 NC = NC - 1:E$ = MIDS (E$,1,NC): IF RIGHTS (E$,1)
= " " THEN 31050
31061 RETURN
31897 REM
31898 REM on n'a pas tape CTRL-H
31899 REM
31900 IF NOT (NC = 0 AND (X = 32 OR X = 13)) THEN 31910
31906 PRINT BIPS: IF X = 13 THEN X = 0
    
```

```

31907 RETURN
31910 IF X < 32 AND X > 13 THEN PRINT BIPS;: RETURN
31911 NC = NC + U; ES = ES + X$: IF X > 13 THEN PRINT X
$: INVERSE : PRINT " ";: NORMAL
31930 RETURN
31947 REM
31948 REM on a tape CTRL-H
31949 REM
31950 IF NC = 0 THEN PRINT BIPS;: RETURN
31956 GOSUB 31970
31960 RETURN
31967 REM
31968 REM Recul
31969 REM
31970 VTAB 2: PRINT : VTAB 20 + INT (NC / 80): POKE 36,N
C - 80 * INT (NC / 80): PRINT " "; CHR$(8); CHR$(
8);: INVERSE : PRINT " ";: NORMAL :NC = NC - 1:ES =
MID$(ES,1,NC): RETURN
56000 REM INITIALISATION
56010 D$ = CHR$(13) + CHR$(4): PMS$ = "Votre phrase ": B
IP$ = CHR$(7): CW = - 958
56020 Z = 0: U = 1
56099 RETURN
57000 REM CREATION du fichier DEMOAPF la 1ere fois
57010 D$ = CHR$(13) + CHR$(4): PRINT D$"OPENDEMOAPF":
PRINT D$"DELETEDEMOAPF": PRINT D$"OPENDEMOAPF,L160"
57020 F$ = "": FOR I = 1 TO 159: F$ = F$ + "$": NEXT
57030 FOR I = 0 TO 99: PRINT D$"WRITEDEMOAPF,R" I: PRINT F
$: NEXT
57040 PRINT D$"CLOSE": END
58000 REM LISTE IMPRIMEE DES PHRASES DE DEMO
58001 REM LA FIN DU FICHIER EST INDIQUEE PAR LA PHRASE :
FIN
58010 TEXT : HOME : PRINT "LISTE IMPRIMEE DES PHRASES DE
DEMO": PRINT : PRINT "Preparer l'IMPRIMANTE puis tap
er une TOUCHE ";: GET X$: PRINT X$
58015 D$ = CHR$(13) + CHR$(4): NU = 1
58016 PRINT D$"PR#1": PRINT "Phrases de la commande DEMO"
: PRINT : PRINT D$"PR#3"
58020 PRINT : PRINT D$"OPENDEMOAPF,L160": VTS = ""
58030 PRINT D$"READDEMOAPF,R" NU
58040 GET X$: IF X$ = CHR$(13) THEN 58050
58045 VTS = VTS + X$: GOTO 58040
58050 PRINT D$"CLOSE": IF VTS = "FIN" THEN PRINT D$"PR#3
": END
58060 PRINT D$"PR#1"
58065 PRINT : PRINT "Phrase " NU ":": PRINT VTS
58070 PRINT D$"PR#3"
58080 NU = NU + 1: GOTO 58020

```

## Programme 'APF 22-4'

```

1 REM *****
2 REM * L'APPRENTI FORTHier Version 3.0 *
3 REM * *
4 REM * Fej Nissuorgsky (c) 1985 CHANCELADE 24650 *
5 REM *****
99 REM
100 REM Programme Principal
101 REM
102 GOSUB 56000: REM Initialisation
104 GOSUB 57200: GOSUB 1350: REM Etablit l'ecran
110 GOSUB 11000: GOSUB 51000: REM Met le DICO
120 SU = VR
130 IF NOT SU THEN 180
135 X = FRE (0)
140 GOSUB 31000: IF MID$(ES,1,4) = "DEMO" THEN GOSUB 1
6000: GOTO 150
141 IF MID$(ES,1,4) = "INIT" THEN GOSUB 350: GOTC 130
150 IF MID$(ES,1,2) = " ": THEN GOSUB 20000: GOTC 160
151 GOSUB 2000
160 IF ER < > 0 THEN GOSUB 10000
170 GOTO 130
180 IF MID$(ES,1,4) = "INIT" THEN RUN
199 END
297 REM
298 REM ENTREE d'une Touche au CLAVIER
299 REM
300 INVERSE : PRINT " ";: NORMAL : & KEY: X = PEEK (768):
X$ = CHR$(X): PRINT CHR$(8): CHR$(32): CHR$(8)::
RETURN
350 VTAB 2: PRINT : VTAB 23: PRINT "Confirmer par 'C' ";:
GOSUB 300: IF X$ = "C" OR X$ = "c" THEN SU = FX: RETU
RN

```

```

355 VTAB 2: PRINT : VTAB 23: CALL CW: RETURN
1297 REM
1298 REM Suite Taper Return
1299 REM
1300 PRINT "Pour continuer Taper une Touche ";: GOSUB 300
:T9 = PEEK (37): VTAB 2: PRINT : VTAB T9 + 1: POKE
36,0: CALL CL: RETURN
1347 REM
1348 REM Mode NORMAL
1349 REM
1350 VTAB 2: PRINT : VTAB 4: POKE 36,62: PRINT "NORMAL
": MF = FX: RETURN
1357 REM
1358 REM Mode PAS A PAS
1359 REM
1360 VTAB 2: PRINT : VTAB 4: POKE 36,62: PRINT "PAS A PAS
": MF = VR: RETURN
1997 REM
1998 REM EXECUTION
1999 REM
2000 PR = U
2011 GOSUB 15900: GOSUB 15920: CO = 4: GOSUB 15000
2015 IF NOT (PR < LEN (E$) AND ER = 0) THEN 2060
2020 GOSUB 32000: GOSUB 32090
2030 IF ER < > Z THEN RETURN
2031 GOSUB 32200: IF NU < > Z THEN 2034
2032 GOSUB 32100: IF NU = 99 THEN GOSUB 3000: GOTO 2039
2033 ER = 18: RETURN
2034 IF NU > 29 AND NU < 39 THEN ER = 6: RETURN
2035 IF NU = 28 THEN ER = 28: RETURN
2036 IF NU = 29 THEN ER = 27: RETURN
2037 GOSUB 2200
2039 IF ER < > Z THEN RETURN
2040 IF MF THEN GOSUB 15920: CO = 42: GOSUB 15000: VTAB 2
: PRINT : VTAB 24: GOSUB 1300
2045 IF MF THEN GOSUB 15900: GOSUB 15920: CO = 4: GOSUB 1
5000
2050 GOTO 2015
2060 IF NOT MF AND ER = Z THEN GOSUB 15920: CO = 42: GOS
UB 15000: VTAB 2: PRINT : VTAB 24: POKE 36,5: GOSUB
1300
2065 GOSUB 15900: GOSUB 15920: CO = 4: GOSUB 15000
2070 RETURN
2197 REM
2198 REM Exécution d'un mot
2199 REM
2200 IF NU > 99 THEN 2300
2210 ON NU GOSUB 3100,3200,3300,3400,3500,3600,3700,3800,
3900,4000,4100,4200,4300,4400,4500,4600,4700,4800,49
00,5000,5100,5200,5300,5400,5500,5600,5700,5800,5900
,6000
2215 IF NU < 31 THEN RETURN
2220 ON NU - 30 GOSUB 6100,6200,6300,6400,6500,6600,6700,
6800,6900,7000,7100,7200,7300,7400,7500,7600,7700,78
00,7900,8000
2240 RETURN
2297 REM
2298 REM Exécution d'un mot créé
2299 REM
2300 ON PEEK (AD + LM + 3) GOTO 2400,2400,2600,2700
2310 ER = 18: RETURN
2397 REM
2398 REM le mot trouvé est une CONSTANTE
2399 REM
2400 GOSUB 15200: IF ER < > Z THEN RETURN
2410 POKE FN VP (PM%), PEEK (AD + LM + 4): POKE FN VP (PM
%) + 1, PEEK (AD + LM + 5): RETURN
2497 REM
2498 REM le mot trouvé est une VARIABLE
2499 REM
2500 GOSUB 15200: IF ER < > Z THEN RETURN
2510 T9 = FN VP (PM%): POKE T9, FN RQ (AD + LM + 4)
2520 POKE T9 + 1, INT ((AD + LM + 4) / 256): RETURN
2597 REM
2598 REM le mot trouvé est un TABLEAU (CARRAY)
2599 REM
2600 PD = 1: GOSUB 15210: IF ER = Z THEN RETURN
2605 TP% = PM% (PM%): IF TP% < Z OR TP% > PEEK (AD + LM +
4) - 1 THEN ER = 33: RETURN
2610 T9 = FN VP (PM%): POKE T9, PEEK (AD + LM + 5 + TP%):
POKE T9 + U, Z
2620 RETURN
2697 REM
2698 REM Exécution d'un MOT CREE
2699 REM
2700 PR% = 1: PR (PR%) = AD + LM + 3

```

```

2730 IF PR% = Z OR ER < > Z THEN RETURN
2735 PT = PR(PR%):PR% = PR% - 1
2740 IF PEEK (PT) = 1 OR PEEK (PT) = 2 OR PEEK (PT) =
3 THEN ON PT GOSUB 2400,2500,2600: GOTO 2730
2750 IF PEEK (PT) < > 4 THEN 2760
2751 IF PEEK (PT + 1) = 255 THEN 2730
2752 GOSUB 15220: IF ER < > Z THEN 2730
2753 PR(PR%) = PT + 1: GOTO 2730
2760 IF PEEK (PT) = 128 THEN N = NU:NU = PEEK (PT + 1):
GOSUB 2820:NU = N: GOTO 2730
2770 IF PEEK (PT) = 129 THEN GOSUB 2950
2800 GOTO 2730
2820 IF NU > 99 OR NU = 23 THEN 2825
2821 GOSUB 2210: IF ER < > Z THEN RETURN
2822 PT = PT + 2: GOSUB 15220: IF ER = Z THEN PR(PR%) = PT
: RETURN
2825 IF ER < > Z THEN RETURN
2830 IF NU = 23 THEN 2900
2835 GOSUB 15220: IF ER < > Z THEN RETURN
2840 PR(PR%) = PT + 2
2845 GOSUB 9000: REM cherche l'adresse du mot cree NU
2850 GOSUB 15220: IF ER < > Z THEN RETURN
2855 PR(PR%) = AD: RETURN
2900 TP = PEEK (PT + 2): GOSUB 15200: IF ER < > Z THEN
RETURN
2905 IF TP < > Z THEN FOR I = U TO TP:PM%(PM%) = PEEK
(PT + 2 + I): GOSUB 5000:PM% = PM% + U: NEXT
2906 PM% = PM% - U: GOSUB 15220: IF ER < > Z THEN RETURN

2910 PR(PR%) = PT + 3 + TP: RETURN
2950 GOSUB 15200: IF ER < > Z THEN RETURN
2955 T9 = FN VP(PM%): POKE T9, PEEK (PT + U): POKE T9 + U
, PEEK (PT + 2)
2965 GOSUB 15220: IF ER < > Z THEN RETURN
2970 PR(PR%) = PT + 3: RETURN
2997 REM
2998 REM **** 99 (immediat) ****
2999 REM
3000 GOSUB 15200: IF ER = Z THEN PM%(PM%) = VN
3005 RETURN
3099 REM **** + ****
3100 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3105 TP = PM%(PM%) + PM%(PM% - U): GOSUB 15400: IF ER < >
Z THEN RETURN
3110 PM%(PM% - U) = TP:PM% = PM% - U: RETURN
3199 REM **** - ****
3200 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3205 TP = PM%(PM% - U) - PM%(PM%): GOSUB 15400: IF ER < >
Z THEN RETURN
3210 PM%(PM% - U) = TP:PM% = PM% - U: RETURN
3299 REM **** * ****
3300 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3305 TP = PM%(PM% - U) * PM%(PM%): GOSUB 15400: IF ER < >
Z THEN RETURN
3310 PM%(PM% - U) = TP:PM% = PM% - U: RETURN
3399 REM **** / ****
3400 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3402 IF PM%(PM%) = Z THEN ER = 2: RETURN
3405 TP = INT (PM%(PM% - U) / PM%(PM%)): GOSUB 15400: IF
ER < > Z THEN RETURN
3410 PM%(PM% - U) = TP:PM% = PM% - U: RETURN
3499 REM **** < ****
3500 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3510 IF PM%(PM% - U) < PM%(PM%) THEN TP = - 1: GOTO 3820
3520 TP = 2: GOTO 3820
3599 REM **** = ****
3600 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3610 IF PM%(PM% - U) = PM%(PM%) THEN TP = - 1: GOTO 3820
3620 TP = 2: GOTO 3820
3699 REM **** > ****
3700 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3710 IF PM%(PM% - U) > PM%(PM%) THEN TP = - 1: GOTO 3820
3720 TP = 2: GOTO 3820
3799 REM **** <> ****
3800 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3810 IF PM%(PM% - U) < > PM%(PM%) THEN TP = - 1: GOTO 3
820
3815 TP = Z
3820 PM% = PM% - U:PM%(PM%) = TP: RETURN
3899 REM **** AND ****
3900 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
3910 IF PM%(PM% - U) = Z OR PM%(PM%) = Z THEN TP = Z: GOT
O 3820
3920 TP = - 1: GOTO 3820
3999 REM **** OR ****
4000 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN

```

```

4010 IF PM%(PM% - U) < > Z OR PM%(PM%) < > Z THEN TP =
- 1: GOTO 3820
4020 TP = Z: GOTO 3820
4099 REM **** NOT ****
4100 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
4110 IF PM%(PM%) = Z THEN PM%(PM%) = - U: RETURN
4120 PM%(PM%) = Z: RETURN
4199 REM **** DUP ****
4200 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
4210 GOSUB 15200: IF ER < > Z THEN RETURN
4220 PM%(PM%) = PM%(PM% - U): RETURN
4299 REM **** DROP ****
4300 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
4310 PM% = PM% - U: RETURN
4399 REM **** OVER ****
4400 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
4410 GOSUB 15200: IF ER < > Z THEN RETURN
4420 PM%(PM%) = PM%(PM% - 2): RETURN
4499 REM **** Ca ****
4500 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
4510 PM%(PM%) = PEEK (PM%(PM%)): RETURN
4599 REM **** C! ****
4600 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
4610 IF PM%(PM% - U) < Z OR PM%(PM% - U) > 255 THEN ER =
30: RETURN
4615 POKE PM%(PM%),PM%(PM% - U):PM% = PM% - 2: RETURN
4699 REM **** ROT ****
4700 PD = 3: GOSUB 15210: IF ER < > Z THEN RETURN
4710 TP = PM%(PM% - 2):PM%(PM% - 2) = PM%(PM% - U):PM%(PM%
- U) = PM%(PM%):PM%(PM%) = TP: RETURN
4799 REM **** SWAP ****
4800 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
4810 TP = PM%(PM% - U):PM%(PM% - U) = PM%(PM%):PM%(PM%) =
TP: RETURN
4899 REM **** CR ****
4900 EL = EL + U:EC = U: IF EL < > 6 THEN RETURN
4910 FOR I = U TO 4:EC$(I) = EC$(I + U): NEXT :EC$(5) =
":EL = 5
4920 GOSUB 50055: RETURN
4999 REM **** EMIT ****
5000 PD = 1: GOSUB 15210: IF ER < > Z THEN RETURN
5010 IF PM%(PM%) > 255 OR (PM%(PM%) < 32 AND PM%(PM%) <
> 13) THEN ER = 30: RETURN
5015 IF PM%(PM%) = 13 THEN GOSUB 4900: RETURN
5020 EC$(EL) = EC$(EL) + CHR$(PM%(PM%)): VTAB 2: PRINT :
VTAB 4 + EL: POKE 36,17 + EC: PRINT CHR$(PM%(PM%
)):EC = EC + 1: IF EC = 17 THEN GOSUB 4900
5030 PM% = PM% - U: RETURN
5099 REM **** ! STORE ****
5100 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
5110 POKE PM%(PM%), PEEK (FN VP(PM% - U) + U): POKE PM%(
PM%) + U, PEEK (FN VP(PM% - U)):PM% = PM% - 2: RETU
RN
5199 REM **** . ****
5200 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
5210 VTS = STR$(PM%(PM%)): GOSUB 4900:EC$(EL) = VTS:EC =
EC + LEN (VTS)
5220 VTAB 2: PRINT : VTAB 4 + EL: POKE 36,18: PRINT EC$(E
L):PM% = PM% - U: RETURN
5299 REM **** " ****
5300 IF MID$(ES,PR,U) = " " THEN PR = PR + U: GOTO 5300
5305 IF PM% > = 11 THEN ER = 21: RETURN
5310 IF MID$(ES,PR,U) = CHR$(34) OR PR > = NC THEN 5
330
5315 PM% = PM% + U:PM%(PM%) = ASC (MID$(ES,PR,U)): GOSU
B 5000: IF ER < > Z THEN RETURN
5320 PR = PR + U: GOTO 5310
5330 IF MID$(ES,PR,U) < > CHR$(34) THEN ER = 31: RET
URN
5340 PR = PR + U: RETURN
5399 REM **** ? ****
5400 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
5410 GOSUB 5510: GOTO 5210
5499 REM **** à AROBAS FETCH ****
5500 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
5510 T9 = PEEK (PM%(PM%)) + PEEK (PM%(PM%) + 1) * 256
5520 IF T9 > NX THEN T9 = T9 - 65536
5530 PM%(PM%) = T9: RETURN
5599 REM **** KEY ****
5600 GOSUB 15200: IF ER < > Z THEN RETURN
5605 VTAB 2: PRINT : VTAB 4 + EL: POKE 36,EC + 17: GOSUB
300: VTAB 2: PRINT :PM%(PM%) = ASC (X$): RETURN
5699 REM **** CLS ****
5700 GOSUB 50005: RETURN
5799 REM **** : ****
5800 ER = 23: RETURN

```

```

5899 REM **** ; ****
5900 ER = 27: RETURN
5999 REM **** I ****
6000 IF PO = Z THEN ER = 24: RETURN
6020 GOSUB 15200: IF ER < > Z THEN RETURN
6030 PM%(PM%) = ID: RETURN
6099 REM **** DO ****
6100 PD = 2: GOSUB 15210: IF ER < > Z THEN RETURN
6110 ID = PM%(PM%):II = PM%(PM% - U):PM% = PM% - 2:PO = PT
: RETURN
6199 REM **** +LOOP ****
6200 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
6210 IF PO = Z THEN ER = 11: RETURN
6215 ID = ID + PM%(PM%):PM% = PM% - U
6220 IF ID < II THEN PT = PO
6230 RETURN
6299 REM **** IF ****
6300 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
6310 IF PM%(PM%) < > Z THEN PM% = PM% - U: RETURN
6320 PM% = PM% - U
6330 ND = 35: GOSUB 8800: IF PT < MX THEN RETURN
6340 ND = 34: GOSUB 8800: RETURN
6399 REM **** THEN ****
6400 RETURN
6499 REM **** ELSE ****
6500 ND = 34: GOSUB 8800: RETURN
6599 REM **** BEGIN ****
6600 PW = PT: RETURN
6699 REM **** WHILE ****
6700 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
6710 IF PM%(PM%) < > Z THEN PM% = PM% - U: RETURN
6720 PM% = PM% - U:ND = 36: GOSUB 8800: RETURN
6799 REM **** REPEAT ****
6800 PT = PW: RETURN
6899 REM **** CONSTANT ****
6900 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
6910 T1 = U: GOTO 7005
6999 REM **** VARIABLE ****
7000 T1 = 2: IF MID$(ES,1, LEN(MOS)) < > MOS THEN ER =
28: RETURN
7005 GOSUB 32000: IF ER < > Z THEN RETURN
7006 IF MOS = CHR$(13) THEN ER = 29: RETURN
7010 GOSUB 32200: IF NU < > Z THEN ER = 20: RETURN
7012 GOSUB 32090
7015 IF FD + LM + 9 > MX OR MC = 255 THEN ER = 4: RETURN
7020 FOR I = Z TO LM - U: POKE FD + I, ASC (MID$(MOS,I
+ U,U)): NEXT
7025 POKE FD + I,32: POKE FD + I + U,2: POKE FD + I + 2,M
C: POKE FD + I + 3,T1
7030 IF T1 = 2 THEN POKE FD + I + 4,2: POKE FD + I + 5,2
: POKE FD + I + 6,255
7035 IF T1 = U THEN POKE FD + I + 4, PEEK (FN VP(PM%)):
POKE FD + I + 5, PEEK (FN VP(PM%) + U): POKE FD +
I + 6,255:PM% = PM% - U
7040 POKE FD + I + 8, FN RQ(DM): POKE FD + I + 7, INT (DM
/ 256)
7050 DM = FD:FD = FD + LM + 9:MC = MC + U
7060 GOSUB 51000: RETURN
7099 REM **** CARRAY ****
7100 PD = U: GOSUB 15210: IF ER < > Z THEN RETURN
7105 TP% = PM%(PM%) + U: IF TP% < U OR TP% > 127 THEN ER =
33: RETURN
7110 GOSUB 32000: IF ER < > Z THEN RETURN
7115 IF MOS = CHR$(13) THEN ER = 29: RETURN
7116 GOSUB 32200: IF NU < > Z THEN ER = 20: RETURN
7120 IF FD + LM + 7 + TP% > MX OR MC = 255 THEN ER = 4: R
ETURN
7125 FOR I = Z TO LM - U: POKE FD + I, ASC (MID$(MOS,I
+ U,U)): NEXT
7130 POKE FD + I,32: POKE FD + I + U,2: POKE FD + I + 2,M
C: POKE FD + I + 3,3
7140 POKE FD + I + 4,TP%: POKE FD + I + 5 + TP%,255
7145 POKE FD + I + 7 + TP%, FN RQ(DM): POKE FD + I + 6 +
TP%, INT (DM / 256)
7150 DM = FD:FD = FD + LM + 8 + TP%:MC = MC + U: GOSUB 510
00:PM% = PM% - U: RETURN
7199 REM **** FORGET ****
7200 GOSUB 32000: IF ER < > Z THEN RETURN
7205 IF MOS = CHR$(13) THEN ER = 29: RETURN
7210 VTAB 2: PRINT : VTAB 23: CALL CW: PRINT "Etes-vous s
ur ? O/N "; GOSUB 300: IF XS < > "O" AND XS < > "
o" AND XS < > "N" AND XS < > "n" THEN 7210
7215 IF XS = "n" OR XS = "N" THEN PR = NC: RETURN
7216 VTAB 2: PRINT
7220 GOSUB 32200: IF MOS = CHR$(13) THEN ER = 29: RETUR
N
7230 IF NU = Z THEN ER = 18: RETURN
7240 IF NU < NM + U THEN ER = 22: RETURN
7250 FD = AD:AD = AD + TP: GOSUB 51800:DM = PEEK (AD) * 2
56 + PEEK (AD + U)
7260 MC = NU: GOSUB 51000: RETURN
7299 REM **** AIDE ****
7300 GOSUB 57000: GOSUB 57200:CO = 4: GOSUB 15000: GOSUB
32090
7310 GOSUB 50050: GOSUB 1350: GOSUB 51000
7320 VTAB 2: PRINT : VTAB 19: PRINT PM$: PRINT E$: RETURN
7399 REM **** QUITTE ****
7400 VTAB 2: PRINT : VTAB 23: PRINT "Confirmer par 'C' ";
: GOSUB 300: IF XS = "C" OR XS = "c" THEN TEXT : HO
ME : PRINT "Au revoir .....": END
7410 VTAB 2: PRINT : VTAB 23: CALL CW: RETURN
7499 REM **** MODE ****
7500 IF NOT MF THEN GOSUB 1360: RETURN
7510 GOSUB 1350: RETURN
7599 REM **** RAZ ****
7600 GOSUB 15300: GOSUB 15900: GOSUB 15920:PR = NC: RETUR
N
7699 REM **** BELL ****
7700 PRINT BIP$;: RETURN
7799 REM **** LISTEMOTS ****
7800 AD = DM: GOSUB 51900:TP = 2: TEXT
7810 IF AD = Z THEN 7870
7815 VTAB 2: PRINT : VTAB 13 + TP: POKE 36,54
7820 IF PEEK (AD) = 32 THEN 7850
7830 PRINT CHR$(PEEK (AD));AD = AD + U: GOTO 7820
7850 TP = TP + U: GOSUB 51800:AD = PEEK (AD) * 256 + PEE
K (AD + U): IF TP = 5 THEN TP = Z: VTAB 2: PRINT : V
TAB 23: GOSUB 1300: VTAB 2: PRINT : VTAB 23: CALL CW
: GOSUB 51900: TEXT
7860 GOTO 7810
7870 IF TP < > Z THEN VTAB 2: PRINT : VTAB 23: GOSUB 13
00: VTAB 2: PRINT : VTAB 23: CALL CW
7880 GOSUB 51000: RETURN
7899 REM **** DEMO ****
7900 ER = 26: RETURN
7999 REM **** INIT ****
8000 ER = 26: RETURN
8798 REM Recherche d'un mot de numero ND dans un mot cr
ee
8799 REM (utilise pour ELSE THEN et REPEAT)
8800 PT = PT + 2
8820 IF PEEK (PT + U) = ND OR PT > = MX THEN RETURN
8830 IF PEEK (PT) = 129 THEN PT = PT + 3: GOTO 8820
8840 IF PEEK (PT) = 128 AND PEEK (PT + U) < > 23 THEN
PT = PT + 2: GOTO 8820
8850 PT = PT + 3 + PEEK (PT + 2): GOTO 8820
8997 REM
8998 REM Cherche l'adresse du mot cree NU
8999 REM
9000 I = DM
9005 IF I = Z THEN RETURN
9010 IF PEEK (I) < > 32 THEN I = I + U: GOTO 9010
9015 IF PEEK (I + 2) = NU THEN AD = I + 4:I = Z: GOTO 90
05
9020 AD = I: GOSUB 51800:I = PEEK (AD) * 256 + PEEK (AD
+ U): GOTO 9005
9997 REM
9998 REM Affichage des messages d'ERREUR
9999 REM
10000 IF ER = 34 THEN MSS = "Erreur d'E/S sur Disquette."
: GOTO 10010
10004 VTAB 2: PRINT : VTAB 2
10005 PRINT D$"OPEN MESSAGE",L39": PRINT D$"READ MESSAGE",R
"ER: INPUT MSS: PRINT D$"CLOSE"
10010 VTAB 2: PRINT : VTAB 23: CALL CW: PRINT MSS: VTAB 2
0:CO = PR: IF PR > 80 THEN CO = PR - 80: VTAB 21
10020 POKE 36,CO - 1: INVERSE : PRINT " ": NORMAL : VTAB
2: PRINT : VTAB 24: GOSUB 1300: VTAB 2: PRINT : VTA
B 23: CALL CW
10030 ER = Z: GOSUB 15900: GOSUB 15920: GOSUB 15300: RETUR
N
10997 REM
10998 REM Mise en place du DICO
10999 REM
11000 VTAB 20: POKE 36,10: PRINT "Preparation du Diction
naire en COURS ..."
11005 PRINT D$"BLOAD DICO"
11010 FD = 34714: REM FD=FINDICO
11020 DM = 34705: REM DM=LASTMOT (DERNIER MOT)
11025 AY = FD: REM FIN DU DICO DE BASE
11030 I = DM:K = NM

```



```

11040 IF I = Z THEN RETURN
11050 IF PEEK (1) < > 32 THEN TS(K) = TS(K) + CHR$ ( P
EEK (1)):I = I + U: GOTO 11050
11060 AD = I: GOSUB 51800:K = K - U: VTAB 20: POKE 36,51:
PRINT CHR$ (42 + K - (INT (K / 2) * 2))
11070 I = PEEK (AD) * 256 + PEEK (AD + U): GOTO 11040
11080 VTAB 20: POKE 36,0: CALL CL
11090 RETURN
11997 REM
11998 REM Examen du DICO (pas utilise par le programme)
11999 REM
12000 I = DM
12006 IF I = Z THEN END
12007 PRINT I;" ";
12010 IF PEEK (1) < > 32 THEN PRINT CHR$ ( PEEK (1));
:I = I + U: GOTO 12010
12020 PRINT " "; PEEK (I + U) * 256 + PEEK (I + 2);"
";AD = I: GOSUB 51800:I = PEEK (AD) * 256 + PEEK
(AD + U): PRINT I: GOTO 12006
14997 REM
14998 REM Affichage Pile Ecran
14999 REM
15000 IF PM% = Z THEN RETURN
15010 FOR I = 1 TO PM%: VTAB 2: PRINT : VTAB 15 - I: POKE
36,CO: PRINT RIGHTS (" " + STR$(PM%(I)),6)
15020 NEXT I: RETURN
15199 REM Pile saturee ??
15200 IF PM% < 11 THEN PM% = PM% + 1: RETURN
15201 ER = 21: RETURN
15209 REM Y a-t-il assez d'elements sur la pile ??
15210 IF PM% < PD THEN ER = 1
15211 RETURN
15219 REM Pile de retour saturee ??
15220 IF PR% < 255 THEN PR% = PR% + 1: RETURN
15221 ER = 21: RETURN
15297 REM
15298 REM Initialisation des Piles
15299 REM
15300 PM% = Z:PR% = Z:MOS = "": GOSUB 32090: RETURN
15397 REM
15398 REM Test si resultat entre NN et NX
15399 REM
15400 IF TP < NN OR TP > NX THEN ER = 3
15401 RETURN
15897 REM
15898 REM Efface Pile AVANT
15899 REM
15900 POKE 33,7: POKE 32,4: POKE 34,3: POKE 35,14: HOME :
TEXT : RETURN
15919 REM Efface Pile APRES
15920 POKE 33,7: POKE 32,42: POKE 34,3: POKE 35,14: HOME
: TEXT : RETURN
15997 REM
15998 REM **** DEMO ****
15999 REM
16000 GOSUB 15900: GOSUB 15920: GOSUB 50005: GOSUB 15300:
GOSUB 1360
16005 VTAB 2: PRINT : VTAB 23: PRINT "Numero de DEMO (XX)
:"; CALL CW: GET X$:X = ASC (X$) - 48: IF X < 0
OR X > 9 THEN PRINT BIP$;: GOTO 16005
16010 NE = X: PRINT X$;: GET X$: IF X$ = CHR$ (13) THEN 1
6025
16015 X = ASC (X$) - 48: IF X < 0 OR X > 9 THEN PRINT BI
P$;: GOTO 16005
16020 NE = NE * 10 + X: PRINT X$;: GET X$: IF X$ < > CHR
$ (13) THEN PRINT BIP$;: GOTO 16005
16025 VTAB 2: PRINT : PRINT D$"OPENDEMOAPP,L160": VTAB 2:
PRINT : PRINT D$"READDEMOAPP,R"NE:ES = ""
16040 GET X$: IF X$ = CHR$ (13) THEN 16050
16045 ES = ES + X$: GOTO 16040
16050 VTAB 2: PRINT : PRINT D$"CLOSE": IF ES = "FIN" OR E
S = "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
THEN ER = 36: RETURN
16060 NC = LEN (ES)
16070 IF MIDS (ES,NC,1) = " " THEN NC = NC - 1: GOTO 160
70
16080 ES = MIDS (ES,1,NC) + CHR$ (13):NC = LEN (ES)
16090 VTAB 2: PRINT : VTAB 19: CALL CW: PRINT PMS: PRINT
ES: RETURN
19997 REM
19998 REM **** COMPILATION ****
19999 REM
20000 GOSUB 15900: GOSUB 15920:CO = 4: GOSUB 15000: GOSUB
20900:CP = VR:PR = 3:FC = 2

```

```

20010 IF MC = 255 THEN ER = 4: RETURN
20020 GOSUB 32000: GOSUB 32090: IF MOS = CHR$ (13) THEN
ER = 19: RETURN
20030 IF ER < > Z THEN RETURN
20040 GOSUB 32200: IF NU < > Z THEN ER = 20: RETURN
20045 A1 = FD
20050 IF A1 + LM + 3 > MX THEN ER = 4: RETURN
20060 FOR I = Z TO LM - 1: POKE A1 + I, ASC ( MIDS (MOS,I
+ U,U)): NEXT
20070 POKE A1 + LM,32:A1 = A1 + LM + U: POKE A1,Z
20080 A1 = A1 + U: POKE A1,MC:A1 = A1 + U: POKE A1,4:A1 =
A1 + U
20100 GOSUB 32000: GOSUB 32090: GOSUB 32200: IF NU = Z TH
EN GOSUB 32100
20105 IF MOS = CHR$ (13) THEN ER = 19: RETURN
20106 IF NU = Z THEN ER = 18: RETURN
20110 IF ER < > Z OR NOT CP THEN 20310
20112 IF NU = 49 OR NU = 50 THEN ER = 5: GOTO 20300
20120 IF NU = 28 THEN ER = 23: GOTO 20300
20130 IF NU > 38 AND NU < 51 AND NU < > 47 THEN ER = 5:
GOTO 20300
20140 IF NU = 23 THEN GOSUB 32000: IF MOS = CHR$ (13) O
R MIDS (MOS,LM,1) < > CHR$ (34) THEN ER = 31: GOT
O 20300
20150 IF NU < > 23 THEN 20160
20151 IF A1 + 3 + LM - U > MX THEN ER = 4: GOTO 20300
20152 POKE A1,128:A1 = A1 + U: POKE A1,23:A1 = A1 + U: PO
KE A1,LM - U: FOR I = U TO LM - U: POKE A1 + I, ASC
( MIDS (MOS,I,U)): NEXT :A1 = A1 + LM: GOTO 20210
20160 IF NU < > 99 THEN 20170
20161 IF A1 + 3 > MX THEN ER = 4: GOTO 20300
20162 POKE A1,129:A1 = A1 + U:TP% = VN: POKE A1, INT (TP%
/ 256):A1 = A1 + U: POKE A1, FN RQ(TP%):A1 = A1 + U
: GOTO 20210
20170 IF NU = 29 THEN CP = FX: GOTO 20300
20180 IF NU > 29 AND NU < 39 THEN GOSUB 21000: IF ER <
> Z THEN 20300
20190 IF A1 + 2 > MX THEN ER = 4: GOTO 20300
20200 POKE A1,128:A1 = A1 + U: POKE A1,NU:A1 = A1 + U
20210 GOSUB 32000: GOSUB 32090: GOSUB 32200: IF NU < > Z
THEN 20300
20211 GOSUB 32100: IF NU < > Z THEN 20300
20212 IF MOS = CHR$ (13) THEN ER = 19: GOTO 20300
20213 ER = 18
20300 GOTO 20110
20310 IF ER < > Z OR CP THEN RETURN
20315 IF FC = 31 THEN ER = 10: RETURN
20316 IF FC = 33 OR FC = 35 THEN ER = 7: RETURN
20317 IF FC = 36 THEN ER = 14: RETURN
20318 IF FC = 37 THEN ER = 13
20319 IF ER < > Z THEN RETURN
20320 IF A1 + 3 > MX THEN ER = 4: RETURN
20330 POKE A1,255:A1 = A1 + U: POKE A1, INT (DM / 256):A1
= A1 + U: POKE A1, FN RQ(DM):A1 = A1 + U:DM = FD
20340 FD = A1:MC = MC + U: GOSUB 51000
20345 MOS = "": GOSUB 32090: VTAB 2: PRINT : VTAB 24: POKE
36,Z
20350 RETURN
20899 REM
20900 VTAB 2: PRINT : VTAB 24: POKE 36,Z: PRINT "Compilat
ion en cours .... Soyez patients.": RETURN
20997 REM
20998 REM **** Analyse SYNTAXIQUE des structures de co
ntrole ****
20999 REM
21000 IF FC = Z THEN GOSUB 21050: RETURN
21010 GOSUB 21100: RETURN
21050 IF NU = 31 OR NU = 33 OR NU = 36 THEN FC = NU: RETU
RN
21055 IF NU = 30 THEN ER = 24: RETURN
21060 IF NU = 32 THEN ER = 11: RETURN
21065 IF NU = 34 THEN ER = 8: RETURN
21070 IF NU = 35 THEN ER = 9: RETURN
21075 IF NU = 37 THEN ER = 12: RETURN
21080 ER = 25: RETURN
21100 IF FC = 31 THEN GOSUB 21150: RETURN
21101 GOSUB 21200: RETURN
21150 IF NU = 31 OR (NU > 32 AND NU < 39) THEN ER = 26: R
ETURN
21151 IF NU = 32 THEN FC = Z
21152 RETURN
21200 IF NU = 30 THEN ER = 24: RETURN
21205 IF FC = 33 THEN GOSUB 21250: RETURN
21210 GOSUB 21300: RETURN
21250 IF NU = 34 THEN FC = Z: RETURN
21251 IF NU = 35 THEN FC = 35: RETURN

```

```

21252 ER = 26: RETURN
21300 IF FC = 35 THEN GOSUB 21350: RETURN
21301 GOSUB 21400: RETURN
21350 IF NU = 34 THEN FC = Z: RETURN
21351 ER = 26: RETURN
21400 IF FC = 36 THEN GOSUB 21450: RETURN
21401 GOTO 21500
21450 IF NU = 37 THEN FC = 37: RETURN
21451 ER = 14: RETURN
21500 IF FC < > 37 THEN ER = 15: RETURN
21505 IF NU = 38 THEN FC = Z: RETURN
21510 ER = 26: RETURN
30997 REM
30998 REM ENTREE d'une ligne de 159 caracteres
30999 REM
31000 VTAB 2: PRINT : VTAB 19: POKE 36,0: CALL CW: PRINT
PMS:NC = 0:ES = "":X = FRE (0):X = Z:MOS = "": GOSUB
B 32090
31010 IF NOT (NC < > 159 AND X < > 13) THEN 31050
31015 VTAB 2: PRINT : VTAB 20 + INT (NC / 80): POKE 36,N
C - INT (NC / 80) * 80
31020 GOSUB 300: IF X < 123 AND X > 96 THEN X = X - 32:X$
= CHR$ (X)
31025 IF X = 27 THEN TEXT : HOME : GOTO 31000
31030 IF X = 8 THEN GOSUB 31950
31040 IF X < > 8 THEN GOSUB 31900
31041 GOTO 31010
31050 IF X < > 13 THEN ER = 16: GOSUB 10000: GOTO 31000
31060 NC = NC - 1:ES = MID$ (ES,1,NC): IF RIGHT$ (ES,1)
= " " THEN 31060
31061 NC = NC + 1:ES = ES + CHR$ (13): RETURN
31897 REM
31898 REM on n'a pas tape CTRL-H
31899 REM
31900 IF NOT (NC = 0 AND (X = 32 OR X = 13)) THEN 31910
31906 PRINT BIP$;: IF X = 13 THEN X = 0
31907 RETURN
31910 IF X < 32 AND X < > 13 THEN PRINT BIP$;: RETURN
31911 NC = NC + U:ES = ES + X$: IF X < > 13 THEN PRINT X
$;
31930 RETURN
31947 REM
31948 REM on a tape CTRL-H
31949 REM
31950 IF NC = 0 THEN PRINT BIP$;: RETURN
31956 NC = NC - 1:ES = MID$ (ES,1,NC): GOSUB 31970
31960 RETURN
31967 REM
31968 REM Recul
31969 REM
31970 VTAB 2: PRINT : VTAB 20 + INT (NC / 80): POKE 36,N
C - 80 * INT (NC / 80): PRINT " " + CHR$ (8);: RET
URN
31997 REM
31998 REM Recherche d'un mot dans l'ENTREE
31999 REM
32000 MOS = "":LM = 0
32010 IF MID$ (ES,PR,1) = " " THEN PR = PR + 1: GOTO 320
10
32040 IF MID$ (ES,PR,1) = " " OR MID$ (ES,PR,1) = CHR$
(13) THEN 32055
32050 MOS = MOS + MID$ (ES,PR,1):PR = PR + 1:LM = LM + 1:
GOTO 32040
32055 IF LM > 16 THEN ER = 17: RETURN
32060 IF MID$ (ES,PR,1) < > CHR$ (13) THEN RETURN
32065 IF LM = 0 THEN MOS = CHR$ (13):PR = PR + 1
32070 RETURN
32087 REM
32088 REM Affiche le MOT
32089 REM
32090 IF MCT$ = CHR$ (13) THEN RETURN
32093 VTAB 2: PRINT : VTAB 14: POKE 36,17: PRINT LEFT$ (
MOS + " " ,16)
32095 RETURN
32097 REM
32098 REM MCS est-il un nombre ??
32099 REM
32100 ER = Z: FOR I = U TO LEN (MOS)
32110 X = ASC ( MID$ (MOS,I,U)): IF (X < > 43 AND X < >
45 AND (X < 48 OR X > 57)) OR (X = 43 AND I < > U)
OR (X = 45 AND I < > U) THEN ER = 18:I = LEN (MOS
)
32120 NEXT : IF ER < > Z THEN RETURN
32130 VN = VAL (MOS):NU = 99: IF VN < NN OR VN > NX THEN
ER = 3
32140 RETURN
32197 REM
32198 REM Recherche du numero d'un mot dans le dico
32199 REM
32200 NU = Z: FOR I = U TO NM
32201 IF T$(I) = MOS THEN NU = I:I = NM
32202 NEXT : IF NU < > Z THEN RETURN
32203 IF MC = 101 THEN RETURN
32209 AD = DM
32210 IF AD < AY OR NU < > Z THEN 32245
32212 TP = Z
32213 IF PEEK (AD + TP) < > 32 THEN TP = TP + U: GOTO 3
2213
32214 IF TP < > LM THEN AD = AD + TP: GOSUB 51800:AD =
PEEK (AD) * 256 + PEEK (AD + U): GOTO 32210
32215 FG = VR
32220 FOR I = Z TO LM - U
32225 IF PEEK (AD + I) < > ASC ( MID$ (MOS,I + U,U)) T
HEN FG = FX:I = LM
32230 NEXT
32235 IF FG THEN NU = PEEK (AD + TP + 2): GOTO 32210
32236 AD = AD + TP: GOSUB 51800:AD = PEEK (AD) * 256 + P
EEK (AD + U): GOTO 32210
32245 IF AD = Z AND NU = Z THEN IF MOS = "+" THEN NU = U
32250 RETURN
39997 REM
39998 REM Gestion des ERREURS
39999 REM
40000 ER = PEEK (222): IF ER > 0 AND ER < 16 THEN ER = 34
: GOSUB 10000: RESUME
40060 PRINT "ERREUR NON TRAITEE N( "ER" EN " PEEK (218) +
256 * PEEK (219): END
50004 REM Efface Ecran du Forth
50005 FOR I = U TO 5:EC$(I) = "": NEXT :EC = U:EL = U: GO
SUB 50055: RETURN
50050 REM Affiche ECRAN du Forth
50055 TEXT : POKE 33,16: POKE 32,18: POKE 34,4: POKE 35,0
9: HOME : TEXT : FOR I = U TO 5: VTAB 4 + I: POKE 36
,18: PRINT EC$(I);: NEXT : RETURN
50897 REM
50898 REM Affichage Fin du DICO
50899 REM
51000 GOSUB 51900:AD = DM
51010 FOR I = 1 TO 5
51020 IF PEEK (AD) = 32 THEN 51040
51030 PRINT CHR$ ( PEEK (AD));:AD = AD + 1: GOTO 51020
51040 IF I = 5 THEN 51050
51045 PRINT : GOSUB 51800:AD = PEEK (AD) * 256 + PEEK (
AD + 1)
51050 NEXT : TEXT : RETURN
51797 REM
51798 REM Calcul l'OFFSET pour trouver l'adresse suivant
e
51799 REM
51800 IF PEEK (AD + 2) < 100 THEN AD = AD + 3: RETURN
51805 IF PEEK (AD + 3) = 1 OR PEEK (AD + 3) = 2 THEN AD
= AD + 7: RETURN
51810 IF PEEK (AD + 3) = 3 THEN AD = AD + PEEK (AD + 4)
+ 6: RETURN
51815 OS = 4
51820 IF PEEK (AD + OS) = 255 THEN 51860
51825 IF PEEK (AD + OS) = 128 AND PEEK (AD + OS + 1) =
23 THEN OS = OS + 3 + PEEK (AD + OS + 2): GOTO 5182
0
51830 IF PEEK (AD + OS) = 129 THEN OS = OS + 3: GOTO 518
20
51835 OS = OS + 2: GOTO 51820
51860 AD = AD + OS + 1: RETURN
51897 REM
51898 REM Efface ecran fin du DICO
51899 REM
51900 TEXT : POKE 33,16: POKE 32,55: POKE 34,12: POKE 35,
17: HOME : RETURN
55999 REM
56000 REM INITIALISATION
56002 HIMEM: 34304: REM $8600
56003 Z = 0:U = 1:ER = Z:I = Z:J = Z:TP = Z:VR = U:FX = Z:
D$ = CHR$ (4):CL = - 868: CW = - 958:X$ = "":X = Z
:MF = FX:PR = U:NU = Z:PM% = Z:PR% = Z:ES = " "
56004 FD = Z:LM = Z:MOS = "":NC = Z:LM = Z:VN = Z:FG = Z:N
X = 32767:NN = - NX:SU = VR:MSS = "":VTS = "":NS =
"":ADS = "":AD = Z:MS = "":MX = 9 * 4096 + 6 * 256:D
M = Z:PD = Z
56005 DD = 8 * 4096 + 6 * 256 + 1:MC = 101:TP% = Z:PMS = "
Pret":NE = U:OS = Z:CP = VR:FC = Z:N = Z:PT = Z:PO =
Z:ID = Z:EC = U:EL = U:CO = Z
56006 II = Z:PW = Z:Tl = Z:T9 = Z:X1 = Z:X2 = Z:ND = Z:DEM

```

```

OS = "":TIS = "L'APPRENTI FORTHier":BIPS = CHR$ (7)
56010 NM = 50: DIM PM$(11),PR(255),EC$(5),T$(NM)
56015 DEF FN RQ(X) = X - INT (X / 256) * 256
56016 DEF FN VP(X) = PEEK (107) + PEEK (108) * 256 + 7
+ 2 * X
56020 ONERR GOTO 40000
56030 RETURN
56999 REM
57000 REM ECRAN d'AIDE
57001 REM
57005 GOSUB 57700: VTAB 1: POKE 36,48: PRINT "...": INV
ERSE : PRINT "Ecran d'AIDE a l'utilisateur": NORMAL
57010 VTAB 3: POKE 36,0: PRINT "DEMO      acces a un f
ichier de demonstration(doit etre en debut de ligne)
"
57020 PRINT "INIT      reinitialise le FORTH (doit etr
e en debut de ligne)"
57030 PRINT "LISTEMOTS liste le dictionnaire a l'ecran
"
57060 PRINT "MODE      commute mode NCRMAL et mode PAS
A PAS"
57070 PRINT "QUITTE    arret du programme"
57080 PRINT "AIDE      affiche cet ecran"
57085 PRINT "RAZ      initialise les piles"
57090 PRINT "CARRAY    tableau d'octets (25 CARRAY XZ
donne un tableau XZ de 26 octets)"
57100 PRINT "Ca et C!  pour utiliser les elements d'un
tableau"
57110 PRINT "+LOOP    il n'y a pas de boucle LOOP, don
c toujours preciser le pas"
57120 PRINT "Nombres  entiers de -32767 a 32767"
57130 PRINT "Autres mots ils sont classiques du FORTH 79
"
57140 PRINT : PRINT "Longueur d'une phrase : limitee a 15
9 caracteres."
57145 PRINT "MOTS CREES limites a 16 caracteres pour le n
om (154 mots maxl.)"
57150 PRINT "L'APPRENTI FORTHier n'admet pas les structur
es de controle imbriquees."
57151 PRINT " Attention aux mots 'a' et 'Ca' (écriture d
ans la memoire)"
57152 PRINT " La recherche des erreurs prend du temps...
Soyez PATIENTS"
57155 PRINT " Ne jamais taper CTRL-C"
57160 VTAB 24: POKE 36,12: GOSUB 1300
57180 RETURN
57199 REM
57200 REM Ecran de travail
57201 REM
57205 GOSUB 57700: INVERSE : FOR I = 4 TO 14: VTAB I: POK
E 36,1: PRINT " "; POKE 36,11: PRINT " "; POKE 3
6,39: PRINT " "; POKE 36,49: PRINT " "; NEXT
57210 VTAB 15: POKE 36,1: PRINT " "; POKE 36,
39: PRINT " "; VTAB 16: POKE 36,2: PRINT
"PILE AVANT"; POKE 36,40: PRINT "PILE APRES"
57215 VTAB 4: POKE 36,16: PRINT " E C R A N   ": V
TAB 10: POKE 36,16: PRINT " "
57220 FOR I = 5 TO 9: VTAB I: POKE 36,16: PRINT " "; PO
KE 36,34: PRINT " "; NEXT
57225 NORMAL : VTAB 12: POKE 36,16: PRINT "MOT en COURS :
": VTAB 4: POKE 36,54: PRINT "MODE :": VTAB 11: POKE
36,53: PRINT "FIN du DICTIONNAIRE"
57299 RETURN
57699 REM
57700 REM **** TITRE
57701 REM
57705 HOME : PRINT TAB( 30);TIS: RETURN
57799 REM

```

## Fichier 'DEMOAPF'

créé par le programme CREDEMO

```

Phrase 1 :
123 58 + 78 SWAP DUP * AND 1 <>
Phrase 2 :
CLS ." FORTH" 42 78 - 555 + 4 *
Phrase 3 :
AIDE LISTEMOTS 12 45 78 96 * + DUP DROP SWAP 45 OVER 78 ROT
Phrase 4 :
: BOOM CR ESSAI ." BOOM" ;
Phrase 5 :
: VALIDE KEY BEGIN 13 <> WHILE KEY REPEAT ." RT" ;

```

```

Phrase 6 :
: ESSAI ." OK" VALIDE CR ." FIN" ;
Phrase 7 :
VARIABLE XYZ
Phrase 8 :
24 CONSTANT CST
Phrase 9 :
: ENTR KEY DUP BEGIN 32 = WHILE KEY DUP REPEAT EMIT ;

```

## Programme 'CREMESS'

```

1 REM *****
2 REM Programme CREMESS
3 REM Creation des Messages d'ERREUR
4 REM pour programme APF
5 REM Le fichier cree est MESSAPF
6 REM C'est un fichier direct
9 REM Fej Nissuorgsky
10 REM *****
99 DS = CHR$ (4): PRINT DS"PR#3"
100 TEXT : HOME : PRINT "CREATION des messages d'erreur d
u programme APF": PRINT "Le fichier s'appelle MESSAPF.
"
105 PRINT : PRINT "Taper <C> pour creer un message ou <V>
pour VOIR un message <G> pour quitter ": GET X$
106 PRINT X$:X = ASC (X$): IF X = 38 THEN END
107 IF X = 67 OR X = 99 THEN GOSUB 110: GOTO 100
108 IF X = 86 OR X = 118 THEN GOSUB 2000: GOTO 100
109 GOTO 105
110 HOME : PRINT "CREATION des Messages (36 Caracteres ma
xi.) <G> Pour Quitter": PRINT : INPUT "Numero : ":NUS:
NU = INT ( VAL (NUS)): IF NU < 1 OR NU > 99 THEN 110
120 X = ASC (X$): IF X = ASC ("G") THEN END
150 M$ = "": PRINT "Votre Message (<G>=Quitte) : ";
152 GET X$:X = ASC (X$): IF X = 13 AND LEN (M$) = 0 THE
N 152
153 IF X = 8 AND LEN (M$) < > 0 THEN GOSUB 1000: GOTO
152
154 IF X = 13 THEN 160
155 IF X = ASC ("G") THEN N = N - 1: RETURN
156 IF X < 32 OR X > 125 THEN 152
157 PRINT X$:M$ = M$ + X$: IF LEN (M$) > 36 THEN INVER
SE : PRINT "Trop long": NORMAL : PRINT : GOTO 150
158 GOTO 152
160 PRINT : PRINT DS"OPEN MESSAPF,L39": PRINT DS"WRITE ME
SSAPF,R"NU
170 PRINT CHR$ (34) + M$ + CHR$ (34): PRINT DS"CLOSE"
180 RETURN
1000 REM On a tape <-- (CTRL-H)
1010 M$ = MID$ (M$,1, LEN (M$) - 1): PRINT : VTAB PEEK (
37): POKE 36,29 + LEN (M$): CALL - 868: RETURN
2000 HOME : PRINT "VOIR un message": PRINT : INPUT "Numer
o : ":NUS:NU = INT ( VAL (NUS)): IF NU < 1 OR NU > 9
9 THEN 2000
2010 PRINT DS"OPEN MESSAPF,L39": PRINT DS"READ MESSAPF,R"
NU: INPUT M$: PRINT DS"CLOSE"
2020 PRINT : PRINT M$: PRINT "Taper une Touche ":
: GET X$: PRINT X$: RETURN
3000 REM Liste IMPRIMEE des MESSAGES
3010 HOME : PRINT "ASSUREZ-VOUS QUE L'IMPRIMANTE EST BRAN
CHEE !!!!!": PRINT : PRINT "VOUS ALLEZ LISTER SUR L'I
MPRIMANTE LES MESSAGES"
3020 GOSUB 3500
3030 NM = 35:DS = CHR$ (4): PRINT DS"PR#1": PRINT "Liste
des Messages d'Erreur du programme L'APPRENTI FORTHie
r": PRINT : PRINT
3035 PRINT DS"OPEN MESSAPF,L39"
3040 FOR I = 1 TO 35: PRINT DS"READ MESSAPF,R"i: INPUT M$
: PRINT RIGHT$ (" " + STR$ (I),2); " ";M$: NEXT
3050 PRINT DS"CLOSE": PRINT DS"PR#0": END
3500 REM ABANDON ?
3510 PRINT : PRINT "POUR ABANDONNER TAPER 'A' SINON <RETU
RN> ": GET X$: IF X$ = "a" OR X$ = "A" THEN END
3520 IF X$ < > CHR$ (13) THEN 3510
3530 PRINT X$: RETURN
5000 REM Creation du Fichier MESSAPF la lere fois
5005 HOME : PRINT "CREATION du FICHIER MESSAPF la lere fo
is": GOSUB 3500
5010 DS = CHR$ (4):M$ = "": FOR I = 1 TO 38:M$ = M$ + ""
: NEXT : PRINT DS"OPEN MESSAPF,L39"
5020 FOR I = 1 TO 50: PRINT DS"WRITE MESSAPF,R"i: PRINT M
$: NEXT
5030 PRINT DS"CLOSE"

```

## Programme 'GARDE.APF'

```

1 REM *****
2 REM * Page de Garde du programme APF V3.0 *
3 REM * Nom de ce programme : GARDE.APF *
4 REM * *
8 REM * Fej Nissourgsky (c) 1985 CHANCELADE 24650 *
9 REM *****
10 PRINT CHR$ (4);"PR#3": TEXT : HOME
20 GOSUB 1000: REM Implante l'AMPERSAND &KEY
25 VTAB 2: PRINT
30 VTAB 1: POKE 36,11: PRINT "PROGRAMME D'ACQUISITION DES
  CONCEPTS DU LANGAGE FORTH"
35 INVERSE
40 VTAB 3: POKE 36,15: PRINT ".....L'APPRENTI FO
  RTHier....."
45 NORMAL
50 VTAB 5: POKE 36,15: PRINT "(c) Fej Nissourgsky 24650
  CHANCELADE"
60 VTAB 8: POKE 36,9: PRINT "Ce logiciel permet de simule
  r un FORTH. Il n'en a pas la vitesse"
70 VTAB 10: POKE 36,6: PRINT "mais il comporte de nombreu
  x messages d'erreur tres explicites"
80 VTAB 12: POKE 36,6: PRINT "et l'utilisateur peut 'voir
  ' l'effet de chacun des 'mots' FORTH sur la"
90 VTAB 14: POKE 36,6: PRINT "pile, l'ecran ou le diction
  naire(mode PAS A PAS). Le mode NORMAL ne montre"
100 VTAB 16: POKE 36,6: PRINT "pas les etats intermediair
  es de la pile."
110 VTAB 18: POKE 36,6: PRINT "Il est possible d'utiliser
  : constantes, variables, tableaux et"
120 VTAB 20: POKE 36,6: PRINT "de creer des mots nouveaux
  . DEMO donne acces a un fichier de demonstration."
130 VTAB 22: POKE 36,6: PRINT "Il s'agit d'un programme d
  'apprentissage, il n'est pas rapide..."
140 VTAB 24: POKE 36,6: PRINT "Pour continuer taper une t
  ouche ";: & KEY
200 VTAB 2: PRINT : PRINT CHR$ (4)"RUN APF 22-4,D1"
999 END
1000 Q = 45
1010 FOR I = 0 TO Q - 1: READ Z#: POKE 769 + I,Z#: NEXT
1020 POKE 1014,1: POKE 1015,3
1190 RETURN
  
```

```

1200 DATA 44,0,192,16,251,173,0,192,44,16,192,41,127,141,
  0,3,201,32,144,14,165,184,24,105,3
1210 DATA 133,184,165,185,105,0,133,185,96,201,13,240,238
  ,201,8,240,234,76,1,3
  
```

## Fichier 'MESSAPF'

créé par le programme 'CREMESS'

```

1 Insuffisance d'éléments sur la pile.
2 Division par ZERO.
3 Dépassement (numérique).
4 Plus de place en MEMOIRE.
5 Mot IMMEDIAT.
6 Mot interdit en mode Immédiat.
7 IF sans THEN.
8 THEN sans IF.
9 ELSE sans IF.
10 DO sans +LOOP.
11 +LOOP sans DO.
12 WHILE sans BEGIN.
13 WHILE sans REPEAT.
14 BEGIN sans WHILE.
15 REPEAT sans WHILE.
16 ENTREE trop LONGUE (>159 caract.).
17 MOT trop LONG (>16).
18 MOT INCONNU.
19 : sans ;
20 MOT non unique.
21 Pile SATURÉE.
22 Dictionnaire PROTEGE.
23 Répétition de ':'.
24 I hors d'une boucle.
25 REPEAT sans BEGIN.
26 Structures de controle imbriquées.
27 ; sans :
28 Le mot doit etre en DEBUT de ligne.
29 Il manque un NOM.
30 Sommet de pile INVALIDE.
31 Il manque un ''.
32 Structure INCCOMPLETE.
33 Hors limites (Tableaux).
34 Erreur sur DISQUE.
35 Absence de ELSE.
  
```

## Éditeur Plein Écran

Le Pacha

# EPE

Apple //e, //e+, //c

Listez vos programmes Basic en avant et en arrière.  
Modifiez, insérez, effacez des caractères en plein écran sans  
relire les lignes.

Recherchez toute chaîne de caractères.  
Choisissez vous-même les codes de contrôle d'EPE.  
Modifiez EPE : le fichier source est sur la disquette.

DOS, ProDOS, 40, 80 colonnes

200,00 F TTC franco (bon de commande page 74).

# Carte SSC & CP/M

J-F Babasse

(2)

La première partie de cet article proposait un programme sous forme de *commande* CP/M permettant de configurer la carte Super Série d'Apple. Après configuration, des fichiers pouvaient être transférés d'un ordinateur à l'autre par la commande CP/M PIP. Une transmission plus évoluée, tel est l'objet de ce deuxième volet.

## XMODEM

(protocole de Ward Christensen)

Le protocole XMODEM a été choisi car il est très courant, et c'est le plus simple des protocoles "hautes performances". Il est en effet beaucoup plus puissant que les protocoles type D1/D3 ou ENQ/ACK, et reste relativement simple à implanter comparé à des "monstres" comme KERMIT.

Dans un protocole logiciel, la gestion de la transmission (appelée aussi Contrôle de flux) n'est plus assurée par les signaux de la RS232 mais par des codes échangés entre émetteur et récepteur. De ceci découlent plusieurs avantages :

- la liaison électrique est plus simple. En local, un câble pour XMODEM se ramène à trois fils seulement : Masse, TXD, RXD. (TXD et RXD étant bien sûr croisés) ;
- le dialogue entre le programme émetteur et le programme récepteur autorise une automatisation complète de la transmission. Il y a envoi d'un bloc de données, contrôle par le récepteur, et envoi d'un accusé de réception ou au contraire d'une demande de renvoi en cas d'erreur. En cas de non réponse

(coupure de ligne momentanée, ou perte du code retour), il y a réémission automatique du bloc. Enfin, les programmes émetteur et récepteur se déconnectent au bout d'un temps maximum, ou d'un nombre donné de renvois sans succès. On peut donc lancer le transfert d'un gros fichier et vaquer à ses occupations en toute sérénité, le programme ne "plantera" pas. En cas d'avarie, le récepteur est assuré d'avoir un fichier peut être partiel, mais valide et fermé correctement.

XMODEM transmet sur 8 bits, ce qui autorise des échanges de fichiers ASCII ou binaires, et avec somme de contrôle, ce qui assure une bonne fiabilité de la transmission.

Enfin, la mise en route de XMODEM émetteur et XMODEM récepteur est contrôlée par les programmes, qui doivent se *rencontrer* dans un délai d'une minute maximum. Ceci permet des transferts téléphoniques avec un équipement rustique, un simple Modem manuel est suffisant. Les correspondants peuvent s'appeler, se mettre d'accord, puis chacun lance son programme de transmission et commute son Modem sur la ligne. Tous les signaux parasites, rattachement de combinés et autres sont sans importance.

## Le protocole

Le format de la transmission est 8 bits de données, 1 bit d'arrêt, pas de parité, vitesse quelconque. Les données entre émetteur et récepteur sont transmises sous forme de blocs de 132 octets, le format des Blocs étant le suivant :

- un octet d'en-tête, SOH (ASCII 1) ;

- un octet donnant le numéro du bloc modulo 256. Le premier bloc est le numéro 1 ;
- un octet donnant le complément à 255 du numéro de bloc, pour vérification ;
- 128 octets de données, soit (au hasard ?) un secteur CP/M ;
- un octet "Checksum" qui est la somme sans retenue des 128 octets de données.

Le contrôle de transmission utilise les codes de contrôle suivants :

- EOT (ASCII 4) Fin de transmission ;
- ACK (ASCII 6) Bloc reçu et correct ;
- NAK (ASCII 21) Refus du Bloc reçu, erreur ;
- CAN (ASCII 24) Annulation de la transmission.

## Déroulement de la transmission côté émetteur

- 1) Attendre pendant une minute un des deux codes suivants :  
CAN abandonner  
NAK démarrer la transmission.  
Au bout d'une minute sans NAK, abandonner.

Ceci est la seconde partie d'un article sur la carte SSC et le CP/M.

Cependant, le contenu de ce numéro (textes et programmes) est utilisable seul.

## 2) Transmission.

- Envoyer un Bloc de 132 octets.
  - Attendre pendant 10 secondes maximum un des codes suivants :  
CAN abandonner  
ACK correct, passer au Bloc suivant.  
NAK refusé, renvoyer le même bloc.
- Au bout de 10 secondes sans une de ces trois réponses, renvoyer le même bloc.
- Un Bloc mauvais n'est réémis que dix fois au maximum, après, on abandonne de la transmission.

## 3) Fin du fichier.

Le dernier Bloc doit comporter des Ctrl-Z de remplissage. Si ce n'était pas le cas (fichier de données multiple exact de 128) il faut renvoyer un Bloc complet de Ctrl-Z.

Après envoi du dernier Bloc, l'émetteur envoie le code EOT, avec la même procédure que pour les envois de Blocs (10 essais maxi, 10 secondes maxi pour recevoir CAN, ACK ou NAK).

### Déroulement de la transmission côté récepteur

1) Attendre quelque chose en provenance de l'émetteur, en envoyant NAK toutes les dix secondes, ceci pendant une minute maximum.

### Help !

*Vous avez la disquette d'accompagnement :*

*Utilisez un programme du type Universal File Conversion pour transférer les fichiers sur votre disquette CP/M puis tapez selon le cas :  
XMODR ou XMODE.*

*Vous n'avez pas la disquette d'accompagnement :*

*Saisissez les codes objet à l'aide de DDT, puis sauvegardez par  
SAVE 6 XMODR.COM  
et SAVE 5 XMODE.COM.  
Enfin exécutez l'une ou l'autre des commandes*

## 2) Réception de quelque chose :

- C'est un bloc de 132 octets. Vérifier l'en-tête (SOH), et le Checksum. En cas d'erreur envoyer NAK. Si c'est correct, vérifier le numéro de bloc :
  - Bloc attendu, envoyer ACK.
  - Bloc précédent (l'émetteur n'a pas reçu le ACK correspondant), laisser passer et envoyer ACK.
  - Tout autre Bloc, le transfert est perdu, envoyer CAN et abandonner.
- C'est le code EOT. Envoyer ACK, fermer le fichier, terminé.

• Ce n'est ni l'un ni l'autre (bloc incomplet, ou "trou" dans la transmission), faire une "purge" de 5 secondes et envoyer NAK. En lecture, tout délai de plus d'une seconde entre deux caractères signifie un problème grave de ligne (même à basse vitesse, 300 bauds, cela fait au moins 30 caractères perdus). L'information sera de toutes façons dégradée, la purge de 5 secondes sert à laisser l'émetteur finir l'envoi du bloc en cours.

**Remarque :** une fois la transmission commencée, le récepteur doit abandonner s'il reste plus de 10 secondes sans rien recevoir (l'émetteur faisant des renvois automatiques, cela indique une ligne complètement coupée).

Lorsqu'on abandonne la transmission, il est recommandé d'envoyer un ou deux codes CAN, afin de libérer éventuellement l'émetteur.

## Le programme

Le programme est en fait découpé en deux commandes CP/M, une pour l'émission, XMODE.COM et l'autre pour la réception, XMODR.COM.

Ces deux programmes sont assez voisins, assurent la gestion du fichier à transmettre ou à recevoir, et le déroulement du protocole.

La carte SSC est initialisée avec 8 bits de données, 1 stop bit et pas de parité. La vitesse n'est pas

modifiée. Il faudra la configurer au préalable, ou bien à l'aide des switches, ou bien grâce à un utilitaire de configuration (SETSSC.COM -voir n° 26- par exemple !).

Pour lancer un envoi de fichier, on entrera la commande :

*XMODE Nomdufichier*

Pour lancer une réception :

*XMODR Nomdufichier*

Il suffit que les deux programmes soient lancés dans la même minute, chaque programme commente l'état de la transmission en indiquant le numéro du bloc en cours, éventuellement le nombre de réémission d'un même bloc en cas de problème, et le résultat de l'envoi de Bloc (correct, erreur etc.)

La gestion de la carte série est réduite au minimum puisqu'il n'y a aucun traitement des signaux de contrôle. Il suffit de tester l'état des registres émission ou réception. A noter que, comme pour SETSSC.COM, la routine de lecture du port série peut retourner au programme appelant un caractère frappé au clavier. Ceci donne donc un moyen éventuel d'interrompre une transmission en frappant un code CAN (Ctrl-X), si vous constatez, par exemple, que votre programme réémet constamment le même bloc, indiquant manifestement un problème chez votre vis-a-vis.

La mise en œuvre est donc très simple, le choix de la vitesse dépendra des conditions d'utilisation :

- en liaison locale par fil entre deux machines, la grande fiabilité de ce protocole autorise des transferts à la vitesse maximale de la carte SSC, 19200 bauds. Ceci permet des échanges de fichiers importants avec une rapidité spectaculaire ;

- en liaison téléphonique, cela dépendra évidemment du standard utilisé : 300 bauds pour Transpac V21, et 1200/75 pour V23. Sous XMODEM, le volume d'information échangé est de 132 octets dans le sens émetteur vers

récepteur, pour 1 octet en sens inverse. La transmission en V23 n'est donc intéressante que si c'est le programme d'émission qui utilise le canal 1200 bauds, les codes du programme récepteur revenant par le canal 75 bauds. Ceci suppose donc l'utilisation d'un Modem répondeur (Mode 2). Si votre Modem ne comporte que le mode Appel, utilisez plutôt Transpac 300 bauds, ce sera plus rapide.

Enfin, dernière remarque, la carte SSC est monovitesse. Pour transmettre en V23, il vous faudra un Modem capable de symétriser ses entrées (celui du Minitel par exemple), ou la solution à deux cartes série, une émission, une réception. Ceci étant, XMODEM rend envisageable la technique consistant à utiliser un port monovitesse, en changeant la vitesse de transmission selon que l'on lit ou écrit, puisque les opérations de lecture et d'écriture sont alternées. Dans ce cas, il faut initialiser avec la vitesse de réception, reconfigurer l'ACIA en vitesse de transmission uniquement au moment d'émettre, et repasser sur la vitesse de réception tout de suite après. ATTENTION, ne jamais changer la vitesse lorsque l'ACIA est en train de faire une conversion

parallèle/série. Il faudra écrire une routine d'émission qui charge le registre de sortie avec l'octet à émettre, et qui attende que ce registre soit vide (bit 4 du registre d'état) avant de modifier la vitesse.

### Aux CP/Mistes télémaniques

Il faut mentionner ici un point qui sort du cadre de cet article, car utilisant un matériel trop spécifique, la Premium Softcard de Microsoft, équipement qui représente l'arme absolue pour l'inconditionnel de CP/M.

Cette carte qui utilise le slot 3 auxiliaire de l'Apple //e rassemble une carte 80 colonnes étendue et une carte Z80, avec toutefois une différence fondamentale avec la Softcard classique : elle utilise un Z80 B à 6 Mhz, et d'autre part les deux processeurs, 6502 et Z80, sont découplés et chacun fonctionne sur ses 64ko de RAM simultanément, alors que sur la Softcard traditionnelle, ils ne travaillent qu'à tour de rôle, sur les mêmes bus, en se mettant alternativement en DMA. Le programme d'application tourne sur le Z80 tandis que le 6502 joue le rôle d'un processeur d'entrées-sorties.

En plus de l'implantation CP/M, Microsoft a donc développé un O.S. particulier, le 65BIOS, qui est en gros une boucle d'attente, avec gestion des entrées-sorties de l'Apple, d'un buffer de clavier et d'un buffer d'imprimante.

L'auteur a réalisé une modification de ce 65BIOS qui permet de gérer quatre buffers : clavier, imprimante, réception RS232 et émission. Ce programme peut utiliser une ou deux cartes SSC (par exemple une réception 1200 bauds et une émission 75 bauds, valeurs au hasard !).

Ceci permet de créer des applications de transmissions très efficaces. La gestion de la transmission des données étant assurée par le 6502, en temps réel, on peut utiliser le Z80 pour faire tourner le programme principal sans impératifs de rapidité (20Ko de buffer réception série), ce qui autorise l'utilisation de langages évolués (Turbo Pascal, Forth, Basic). A titre indicatif, un émulateur de terminal Minitel, compilé en Forth, occupe moins de 4Ko de mémoire, et avec une facilité de mise au point et de modifications très supérieure.

Les CP/Mistes télémaniques intéressés peuvent me contacter par l'intermédiaire de la revue.



### Source 'XMODR.SOURCE'

```

TITLE XMODR
;-----
; Réception de fichiers au protocole XMODEM
;
; JF-R                               1986
;-----
;
; Adresses CP/M
;-----
BOOT      EQU    0
BDOS      EQU    5
FCB       EQU    5CH
BUFFER    EQU    80H
;
.PHASE    103H
JP       START
;
; Routines CP/M
;-----
; Affiche le message pointé par DE
;-----
; 2 points d'entrée, avec et sans saut de ligne

```

```

; selon Flag DEBUT (IY+5)
;
PRMSG:    BIT    0, (IY+5)
          LD    C, 9
          JP    Z, BDOS
          PUSH DE
          LD    DE, MSG4
          RES   0, (IY+5)
          CALL PRMSG1
          POP  DE
PRMSG1:   LD    C, 9
          JP    BDOS
;
; Affiche le caractère dans E
;-----
PRCAR:    LD    C, 2
          JP    BDOS
;
; Messages du programme
;-----
;
MSGG:     DB    13, 10
          ASC   'Bad Command...$'
MSG1:     DB    27, '*'
          ASC   '-----'
          DB    13, 10
          ASC   'XMODEM  Downloading Utility'
          DB    13, 10

```

```

ASC ' JF-R 1986'
DB 13,10
ASC '-----'
MSG2: ASC 'Can't open File'
DB 13,10,'$'
MSG3: ASC 'Prompting Sender $'
MSG4: DB 13,10,13,10,'$'
MSG5: ASC ' Received Bloc $'
NUMBLOC: ASC 'xxx Ok'
DB 13,10,13,10,'$'
MSG6: ASC ' Header Error'
DB 13,10,'$'
MSG7: ASC ' Checksum Error'
DB 13,10,'$'
MSG8: ASC ' Fatal Error'
DB 13,10,'$'
MSG9: ASC ' Timeout'
DB 13,10,'$'
MSG10: ASC ' Disk Error'
DB 13,10,'$'
MSG14: ASC ' Got something, but ? ? ?'
DB 13,10,'$'
;
MSG11: ASC ' Received EOT'
DB 13,10,'$'
MSG12: DB 13,10
ASC 'File received successfully'
DB 13,10,'$'
MSG13: DB 13,10
ASC 'Transmission aborted'
DB 13,10,'$'
;
; Constantes et variables du programme
; -----
;
T10 EQU 10 ;Temporisation 10 s
T60 EQU 60 ;Temporisation 1 mn
MAXTRY EQU 10 ;Nombre d'essais maxi
BSIZE EQU 128 ;Taille Buffer
;
; Caractères Ascii du protocole
SOH EQU 1
EOT EQU 4
ACK EQU 6
NAK EQU 21
CAN EQU 24
SUB EQU 26
;
; Table de variables, pointée par IY
COMPT: DS 1 ;Compteur
NESSAI: DS 1 ;Nb d'essais
FINI: DS 1 ;Flag fin
NBLOC: DS 1 ;Bloc courant
PREBLOC: DS 1 ;Bloc précédent
DEBUT: DS 1 ;Flag départ
CODE: ES 1 ;Code à envoyer
;
REC_BUF: DS BSIZE+4 ;Buffer de réception
PAGE
START EQU $ ;Début programme
;
LD A, (BUFFER)
OR A ;Nom de fichier ?
LD DE,MSG0
JP Z,PRMSG1 ;Non, retour CCP
LD DE,MSG1
CALL PRMSG1
;
LD C,15 ;Ouverture fichier
LD DE,FCB
CALL BDOS
INC A ;Trouvé ?
JR NZ,PREPAR
;
LD C,22 ;Création fichier
LD DE,FCB
CALL BDOS
INC A ;Correct ?
LD DE,MSG2
JP Z,PRMSG1 ;Non, je sort

```

```

;
; Préparation
; -----
PREPAR: CALL CONF_ACIA
LD IY,COMPT
LD A,1
LD (IY+3),A ;1er bloc
XOR A
LD (IY+5),A ;rdébut
LD (IY+4),A ;précédent
LD A,6
LD (IY+1),A ;Nb essais
LD A,NAK
LD (IY+6),A ;Code d'appel
;
LD DE,REC_BUF+3
LD C,1AH ;Adresse DMA vers Buffer
CALL BDOS
LD DE,MSG3
CALL PRMSG1
SET 0,(IY+5)
;
; Envoi code d'appel
; -----
ENVOI: LD E,'.'
CALL PRCAR
ENVOI1: LD A,T10
LD (IY+0),A ;Attente 10 s
LD DE,REC_BUF ;Buffer d'arrivée
LD B,BSIZE+4
LD C,(IY+6) ;Caractère
CALL ECRITURE
;
; Attente réponse
; -----
WAIT: CALL LECTURE ;Qlque chose ?
JR NC,RANGE
DEC (IY+0) ;J'attends encore ?
JR NZ,WAIT
DEC (IY+1) ;Un autre essai ?
JR NZ,ENVOI
LD DE,MSG9 ;Non, trop tard
JP ANNULE
;
; Boucle interne au Buffer
; -----
WAIT1: CALL LECTURE
JR NC,RANGE
;
; Buffer incomplet
LD A,BSIZE+3
CP B ;1 car reçu ?
JR NZ,PURGE
DEC DE
LD A,(DE)
CP EOT ;Fin de transmission ?
JP Z,TERMINE ;Oui
; Pb réception, on purge et on refuse
PURGE: LD B,5
CALL SECONDE
LD DE,MSG14
JP REFUSE
;
; Remplissage du Buffer de réception
RANGE: LD (DE),A
INC DE
DJNZ WAIT1
;
; Traitement du Bloc reçu
; -----
LD DE,MSG5
CALL PRMSG
LD IX,REC_BUF
LD DE,MSG6
LD A,SOH
CP (IX+0) ;Header correct ?
JR NZ,REFUSE
LD A,(IX+2)
CPL
CP (IX+1)
JR NZ,REFUSE

```



```

/
CP (IY+4) ;Bloc précédent ?
JR Z,ACCEPTE
LD DE,MSG8
CP (IY+3) ;Bloc attendu ?
JR NZ,ANNULE

/
XOR A ;Init checksum
LD IX,REC_BUF+3
LD B,BSIZE
CALC_CHK: ADD A,(IX+0)
INC IX
DJNZ CALC_CHK

/
LD DE,MSG7
CP (IX+0) ;Checksum Ok ?
JR NZ,REFUSE

/
; Ecriture sur Disque des données
/ -----
LD C,21
LD DE,FCB
CALL BDOS
OR A ;Problème ?
LD DE,MSG10
JR NZ,ANNULE ;Oui, on ferme

/
LD A,(IY+3)
LD (IY+4),A
INC (IY+3) ;Bloc suivant

/
; Accepte le Bloc reçu
/ -----
ACCEPTE: LD IX,NUMBLOC
CALL CONVERS ;No de bloc en Ascii
LD DE,NUMBLOC
LD A,ACK

/
SUIVANT: LD (IY+6),A
LD A,1
LD (IY+1),A ;1 essai
CALL PRMSG
JP ENVOI1 ;Validation

/
; Refuse le Bloc reçu
/ -----
REFUSE: LD A,NAK
JR SUIVANT

/
; Annule la transmission
/ -----
ANNULE: CALL PRMSG
LD C,CAN
CALL ECRITURE
LD B,3
CALL SECONDE
CALL ECRITURE
LD DE,MSG13
JR CLOSE

/
; Fin de transmission
/ -----
TERMINE: LD DE,MSG11
CALL PRMSG
LD C,ACK
CALL ECRITURE
LD DE,MSG12

/
CLOSE: CALL PRMSG
LD C,16 ;Ferme le fichier
LD DE,FCB
CALL BDOS
LD DE,BUFFER
LD C,1AH
JP BDOS

/
PAGE
; Routines de service
/ -----
/
; Adresses du port RS232

```

## Glossaire

Les messages en anglais simplifient les transmissions outre-manche et outre-océan mais si la langue des Beatles vous échappe...

Downloading	Télé-réception
Can't open file	Ouverture fichier impossible
Prompting sender	Attente de l'émetteur
Received bloc	Bloc reçu
Header error	Erreur d'entête
Checksum error	Erreur de somme de contrôle
Got something but ???	Bien reçu, mais qu'est-ce ?
Received EOT	Code de fin de fichier reçu
Uploading	Téléchargement
Waiting for receiver	Attente du récepteur
Sending bloc	Envoi bloc
Acknowledged	Compris
Cancelled	Annulé
File sent successfully	Fichier bien transmis
Transmission aborted	Transmission annulée
Bad command	Erreur de commande

```

/ -----
NSLOT EQU 0E0A0H ;Carte en slot 2
DIPSW1 EQU NSLOT+1 ;Switches 1
DIPSW2 EQU NSLOT+2 ;Switches 2
RDREG EQU NSLOT+8 ;Registre lecture
TDREG EQU NSLOT+8 ;Registre écriture
STATUS EQU NSLOT+9 ;Registre d'état
COMMAND EQU NSLOT+0AH ;Registre de commande
CONTROL EQU NSLOT+0BH ;Registre de controle

/
; Clavier
/ -----
KBD EQU 0E000H
KBDSTR EQU 0E010H

/
; Configuration de l'ACIA
/ -----
; 8 DATA, 1 STOP, sans Parité, Vitesse courante
; Utilise A
/
CONF_ACIA: LD A,(CONTROL)
AND 0FH ;Récup. vitesse
OR 10H ;Data et Stop
LD (CONTROL),A
LD A,9 ;RTS, DTR, sans parité
LD (COMMAND),A
LD A,(RDREG) ;Init RDR
RET

/
; Routine de lecture du port RS232.
/ -----
; Retour avec caractère dans A et Carry à 0,
; ou retour au bout d'une seconde avec Carry à 1.
; Si frappe Clavier, retour avec car. clavier.
; Utilise A, HL, IX
/
LECTURE: LD HL,STATUS
BIT 3,(HL) ;RDR plein ?
JR NZ,LECT2
LD IX,TEMPO
XOR A
LD (IX+0),A
LD A,98
LD (IX+1),A
LECT1: BIT 3,(HL)
JR NZ,LECT2
LD A,(KBD)
OR A
JP M,LECT3

```

# Source 'XMODE.SOURCE'

```

DEC (IX+0)
JR NZ,LECT1
DEC (IX+1)
JR NZ,LECT1
SCF
RET
LECT2: DEC HL ;Adresse RDR
LD A,(HL)
OR A
RET
LECT3: LD (KBDSTR),A
AND 7FH
RET
;
; Routine d'écriture du port RS232
; -----
; Écrit le contenu de C.
; Utilise C et HL
;
ECRITURE: LD HL,STATUS
ECRI1: BIT 4,(HL) ;TDR vide ?
JR Z,ECRI1
DEC HL ;Adresse RDR
LD (HL),C
RET
;
; Boucle d'attente, nb de secondes dans B
; -----
; Utilise A et IX
;
SECONDE: LD IX,TEMPO
SECON1: XOR A
LD (IX+0),A
LD A,228 ;init. temporisation
LD (IX+1),A
SECON2: DEC (IX+0)
JR NZ,SECON2
DEC (IX+1)
JR NZ,SECON2
DJNZ SECON1
RET
;
TEMPO: DS 2 ;compteur
;
; Conversion en Ascii du contenu de A
; -----
; Range à (IX), sur 3 octets.
; Utilise A, IX, DE, HL
;
CONVERS: LD L,A
LD H,0
LD DE,100
CALL CONVER1
LD DE,10
CALL CONVER1
LD A,30H
OR L
LD (IX+0),A
LD A,(IX+0FEH)
CP 30H ;0 ?
RET NZ
LD A,20H ;on met un Blanc
LD (IX+0FEH),A
LD A,(IX+0FFH)
CP 30H
RET NZ
LD A,20H
LD (IX+0FFH),A
RET
CONVER1: XOR A
CONVER2: SBC HL,DE
JR C,CONVER3
INC A
JR CONVER2
CONVER3: ADD HL,DE
OR 30H
LD (IX+0),A
INC IX
RET
;
END

```

```

TITLE XMODE
;-----
;
; Emission de fichiers au protocole XMODEM
;
; JF-R
; 1986
;-----
; Adresses CP/M
; -----
BOOT EQU 0
BDOS EQU 5
FCB EQU 5CH
BUFFER EQU 80H
;
.PHASE 103H
JP START
;
; Routines CP/M
; -----
; Affiche le message pointé par DE
; -----
PRMSG: LD C,9
JP BDOS
;
; Affiche le caractère dans E
; -----
PRCAR: LD C,2
JP BDOS
;
; Messages du programme
; -----
MSG0: DB 13,10
ASC 'Bad Command...$'
MSG1: DB 27,'*'
ASC '-----'
DB 13,10
ASC ' XMODEM Uploading Utility'
DB 13,10
ASC ' JF-R 1986'
DB 13,10
ASC '-----'
DB 13,10,13,10,'$'
MSG2: ASC 'Can''t open File'
DB 13,10,'$'
MSG3: ASC 'Waiting for receiver.'
DB 13,10,13,10,'$'
DS 1
MSG4: ASC 'xx Sending Bloc '
NUMBLOC: ASC 'xxx $'
DS 1
MSG5: ASC 'xx Sending EOT '
ASC ' $'
;
MSG6: ASC 'Acknowledged'
DB 13,10,10,'$'
MSG7: ASC 'Non Acknowledged'
DB 13,10,'$'
MSG8: ASC 'Cancelled'
DB 13,10,'$'
MSG9: ASC 'Timeout'
DB 13,10,'$'
;
MSG10: DB 13,10
ASC 'File sent successfully'
DB 13,10,'$'
MSG11: DB 13,10
ASC 'Transmission aborted'
DB 13,10,'$'
;
; Constantes et variables du programme
; -----
;
T10 EQU 10 ;Temporisation 10 s
T60 EQU 60 ;Temporisation 1 mn

```

```

MAXTRY      EQU 10 ;Nombre d'essais maxi
BSIZE      EQU 128 ;Taille Buffer
;
; Caractères Ascii du protocole
SOH        EQU 1
EOT        EQU 4
ACK        EQU 6
NAK        EQU 21
CAN        EQU 24
SUB        EQU 26
;
; Table de variables, pointées par IY
COMPT:     DS 1 ;Compteur
NESSAI:    DS 1 ;Nb d'essais
FINI:      DS 1 ;Flag fin
NBLOC:     DS 1 ;Bloc courant
;
PAGE
START      EQU $ ;Début programme
;
LD A, (BUFFER)
OR A ;Nom de fichier ?
LD DE,MSG0
JP Z,PRMSG ;Non, retour CCP
LD DE,MSG1 ;En-tete
CALL PRMSG
;
LD C,15 ;Ouverture fichier
LD DE,FCB
CALL BDOS
INC A ;Fichier trouvé ?
LD DE,MSG2
JP Z,ABORT ;Non, erreur.
;
; Préparation
; -----
CALL CONF_ACIA
LD IY,COMPT
XOR A
LD (IY+2),A ;Pas fini
INC A
LD (IY+3),A ;Bloc 1
LD A,T60
LD (IY+0),A ;1 minute
LD DE,MSG3
CALL PRMSG
;
; Attente récepteur
; -----
WAIT_REC:  CALL LECTURE ;Caractère ?
JR NC,WAIT2
WAIT1:    DEC (IY+0) ;J'attends encore ?
JR NZ,WAIT_REC
LD DE,MSG9 ;Temps maxi
JP ABORT
WAIT2:    LD DE,MSG8
CP CAN ;Cancel ?
JP Z,ABORT
CP NAK
JR NZ,WAIT1
;
; Lire un Buffer
; -----
RDBUF     EQU $
LD C,20
LD DE,FCB
CALL BDOS
OR A ;Fin de fichier ?
JR Z,ENVOI ;Non, envoi du Bloc
;
SET 0,(IY+2) ;Fin
LD IX,BUFFER+BSIZE-1
LD A,SUB ;Fin du bloc
CP (IX+0) ; = Ctrl-Z ?
JR Z,ENVOI
;
; Renvoyer un bloc plein de Ctrl-Z
LD IX,BUFFER
LD B,BSIZE
RDBUF1:   LD (IX+0),A
INC IX
DJNZ RDBUF1
;
RES 0,(IY+2) ;Pas fini
;
; Envoi du Bloc
; -----
ENVOI:    LD A,MAXTRY ;10 essais max
LD (IY+1),A
;
ENVOI1:   LD B,(IY+1)
LD A,MAXTRY+1
SUB B ;No de l'essai en cours
BIT 0,(IY+2) ;Fin ?
JR Z,ENVOI2 ;Non, sortie du Bloc
;
LD IX,MSG5-1
CALL CONVERS ;No essai en Ascii
LD DE,MSG5
CALL PRMSG
LD C,EOT
CALL ECRITURE
JR VALID
;
ENVOI2:   LD IX,MSG4-1
CALL CONVERS ;No essai
LD A,(IY+3) ;No Bloc
LD IX,NUMBLOC
CALL CONVERS
LD DE,MSG4
CALL PRMSG
;
LD C,SOH ;Header
CALL ECRITURE
LD A,(IY+3)
LD C,A ;No Bloc
CALL ECRITURE
CPL
LD C,A ;255-No Bloc
CALL ECRITURE
XOR A ;Init Checksum
LD IX,BUFFER
LD B,BSIZE
ENV_BCL:  LD C,(IX+0) ;Caractère
ADD A,C
INC IX
CALL ECRITURE
DJNZ ENV_BCL
LD C,A ;Checksum
CALL ECRITURE
;
; Attente validation
; -----
VALID:    LD A,T10 ;T max
LD (IY+0),A
VALID1:   CALL LECTURE ;Réponse ?
JR NC,VALID4 ;Oui
VALID2:   DEC (IY+0) ;J'attends encore ?
JR NZ,VALID1 ;Oui
LD DE,MSG9
CALL PRMSG
;
; Si pas de réponse, on renvoi le Bloc actuel
; -----
VALID3:   DEC (IY+1) ;Encore un essai ?
JR NZ,ENVOI1 ;Oui, renvoi du Bloc
JP ABORT1
;
; Test du code reçu
; -----
VALID4:   CP CAN
JR NZ,VALID5
LD DE,MSG8
JR ABORT
VALID5:   CP NAK
JR NZ,VALID6
LD DE,MSG7
CALL PRMSG
JR VALID3
VALID6:   CP ACK
JR NZ,VALID2
;
; Bloc accepté
; -----

```

```

LD DE,MSG6
CALL PRMSG
BIT 0,(IY+2) ;Fin ?
JR NZ,TERMINE
INC (IY+3) ;Bloc suivant
JP RDBUF
;
; Fin de transmission
; -----
TERMINE: LD DE,MSG10
JP PRMSG ;Message et retour CCP
;
ABORT: CALL PRMSG
ABORT1: LD DE,MSG11
JP PRMSG
PAGE
; Routines de service
; -----
; Adresses du port RS232
; -----
NSLOT EQU 0E0A0H ;Carte en slot 2
DIPSW1 EQU NSLOT+1 ;Switches 1
DIPSW2 EQU NSLOT+2 ;Switches 2
RDREG EQU NSLOT+8 ;Registre lecture
TDREG EQU NSLOT+8 ;Registre écriture
STATUS EQU NSLOT+9 ;Registre d'état
COMMAND EQU NSLOT+0AH ;Registre de commande
CONTROL EQU NSLOT+0BH ;Registre de controle
;
; Clavier
; -----
KBD EQU 0E000H
KBDSTR EQU 0E010H
;
; Configuration de l'ACIA
; -----
; 8 DATA, 1 STOP, sans Parité, Vitesse courante
; Utilise A
;
CONF_ACIA: LD A,(CONTROL)
AND 0FH ;Récup. vitesse
OR 10H ;Data et Stop
LD (CONTROL),A
LD A,9 ;RTS, DTR, sans parité
LD (COMMAND),A
LD A,(RDREG) ;Init RDR
RET
;
; Routine de lecture du port RS232.
; -----
; Retour avec caractère dans A et Carry à 0,
; ou retour au bout d'une seconde avec Carry à 1.
; Si frappe Clavier, retour avec car. clavier.
; Utilise A, HL, IX
;
LECTURE: LD HL,STATUS
BIT 3,(HL) ;RDR plein ?
JR NZ,LECT2
LD IX,TEMPO
XOR A
LD (IX+0),A
LD A,98
LD (IX+1),A
LECT1: BIT 3,(HL)
JR NZ,LECT2
LD A,(KBD)
OR A
JP M,LECT3
DEC (IX+0)
JR NZ,LECT1
DEC (IX+1)
JR NZ,LECT1
SCF
RET
LECT2: DEC HL ;Adresse RDR
LD A,(HL)
OR A
RET
LECT3: LD (KBDSTR),A
AND 7FH

```

```

RET
;
; Routine d'écriture du port RS232
; -----
; Ecrit le contenu de C.
; Utilise C et HL
;
Ecriture: LD HL,STATUS
Ecri1: BIT 4,(HL) ;TDR vide ?
JR Z,Ecri1
DEC HL ;Adresse RDR
LD (HL),C
RET
;
; Boucle d'attente, nb de secondes dans B
; -----
; Utilise A et IX
;
SECONDE: LD IX,TEMPO
SECON1: XOR A
LD (IX+0),A
LD A,228 ;init. temporisation
LD (IX+1),A
SECON2: DEC (IX+0)
JR NZ,SECON2
DEC (IX+1)
JR NZ,SECON2
DJNZ SECON1
RET
;
TEMPO: DS 2 ;compteur
;
; Conversion en Ascii du contenu de A
; -----
; Range à (IX), sur 3 octets.
; Utilise A, IX, DE, HL
;
CONVERS: LD L,A
LD H,0
LD DE,100
CALL CONVER1
LD DE,10
CALL CONVER1
LD A,30H
OR L
LD (IX+0),A
LD A,(IX+0FEH)
CP 30H ;0 ?
RET NZ
LD A,20H ;on met un Blanc
LD (IX+0FEH),A
LD A,(IX+0FFH)
CP 30H
RET NZ
LD A,20H
LD (IX+0FFH),A
RET
CONVER1: XOR A
CONVER2: SBC HL,DE
JR C,CONVER3
INC A
JR CONVER2
CONVER3: ADD HL,DE
OR 30H
LD (IX+0),A
INC IX
RET
;
END

```

### Dump hexadécimal de XMODE.COM

```

0100 00 00 00 C3 6D 02 0E 09 C3 05 00 0E 02 C3 05 00
0110 0D 0A 42 61 64 20 43 6F 6D 6D 61 6E 64 2E 2E 2E
0120 24 1B 2A 20 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D
0130 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 0D 0A
0140 20 58 4D 4F 44 45 4D 20 20 20 55 70 6C 6F 61 64
0150 69 6E 67 20 55 74 69 6C 69 74 79 0D 0A 20 4A 46
0160 2D 52 20 20 20 20 20 20 20 20 20 20 20 20 20 20
0170 20 20 20 20 31 39 38 36 0D 0A 20 2D 2D 2D 2D 2D

```

```

0180 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D
0190 2D 2D 2D 2D 2D 0D 0A 0D 0A 24 43 61 6E 27 74 20
01A0 6F 70 65 6E 20 46 69 6C 65 0D 0A 24 57 61 69 74
01B0 69 6E 67 20 66 6F 72 20 72 65 63 65 69 76 65 72
01C0 2E 0D 0A 0D 0A 24 E5 78 78 20 20 53 55 6E 64 69
01D0 6E 67 20 42 6C 6F 63 20 78 78 78 20 20 20 20 20
01E0 24 E5 78 78 20 20 53 65 6E 64 69 6E 67 20 45 4F
01F0 54 20 20 20 20 20 20 20 20 20 20 24 41 63 6B 6E
0200 6F 77 6C 65 64 67 65 64 0D 0A 24 4E 6F 6E 20
0210 41 63 6B 6E 6F 77 6C 65 64 67 65 64 0D 0A 24 43
0220 61 6E 63 65 6C 6C 65 64 0D 0A 24 54 59 6D 65 6F
0230 75 74 0D 0A 24 0D 0A 46 69 6C 65 20 73 65 6E 74
0240 20 73 75 63 63 65 73 73 66 75 6C 6C 79 0D 0A 24
0250 0D 0A 54 72 61 6E 73 6D 69 73 73 69 6F 6E 20 61
0260 62 6F 72 74 65 64 0D 0A 24 E5 E5 E5 55 3A 80 00
0270 B7 11 10 01 CA 06 01 11 21 01 CD 06 01 0E 0F 11
0280 5C 00 CD 05 00 3C 11 9A 01 CA 9E 03 CD A7 03 FD
0290 21 69 02 AF FD 77 02 3C FD 77 03 3E 3C FD 77 00
02A0 11 AC 01 CD 06 01 CD BA 03 30 0B FD 35 00 20 F6
02B0 11 2B 02 C3 9E 03 11 1F 02 FE 18 CA 9E 03 FE 15
02C0 20 E9 0E 14 11 5C 00 CD 05 00 B7 28 20 FD CB 02
02D0 C6 DD 21 FF 00 3E 1A DD BE 00 28 11 DD 21 80 00
02E0 06 80 DD 77 00 DD 23 10 F9 FD CB 02 86 3E 0A FD
02F0 77 01 FD 46 01 3E 0B 90 FD CB 02 46 28 14 DD 21
0300 E1 01 CD 15 04 11 E2 01 CD 06 01 0E 04 CD EF 03
0310 18 3E DD 21 C6 01 CD 15 04 FD 7E 03 DD 21 D8 01
0320 CD 15 04 11 C7 01 CD 06 01 0E 01 CD EF 03 FD 7E
0330 03 4F CD EF 03 2F 4F CD EF 03 AF DD 21 80 00 06
0340 80 DD 4E 00 81 DD 23 CD EF 03 10 F5 4F CD EF 03
0350 3E 0A FD 77 00 CD BA 03 30 13 FD 35 00 20 F6 11
0360 2B 02 CD 06 01 FD 35 01 20 88 C3 A1 03 FE 18 20
0370 05 11 1F 02 18 28 FE 15 20 08 11 0C 02 CD 06 01
0380 18 E3 FE 06 20 D4 11 FC 01 CD 06 01 FD CB 02 46
0390 20 06 FD 34 03 C3 C2 02 11 35 02 C3 06 01 CD 06
03A0 01 11 50 02 C3 06 01 3A AB E0 E6 0F F6 10 32 AB
03B0 E0 3E 09 32 AA E0 3A A8 E0 C9 21 A9 E0 CB 5E 20
03C0 24 DD 21 13 04 AF DD 77 00 3E 62 DD 77 01 CB 5E
03D0 20 13 3A 00 E0 B7 FA E9 03 DD 35 00 20 F0 DD 35
03E0 01 20 EB 37 C9 2B 7E B7 C9 32 10 E0 56 7F C9 21
03F0 A9 E0 CB 66 28 FC 2B 71 C9 DD 21 13 04 AF DD 77
0400 00 3E E4 DD 77 01 DD 35 00 20 FB DD 35 01 20 F6
0410 10 EB C9 3E FF 6F 26 00 11 64 00 CD 41 04 11 0A
0420 00 CD 41 04 3E 30 B5 DD 77 00 DD 7E FE FE 30 C0
0430 3E 20 DD 77 FE DD 7E FF FE 30 C0 3E 20 DD 77 FF
0440 C9 AF ED 52 38 03 3C 18 F9 19 F6 30 DD 77 00 DD
0450 23 C9 00 00 00 00 00 00 00 00 00 00 00 00 00

```

```

0300 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0310 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0320 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350 00 00 00 00 00 00 00 00 3A 80 00 B7 11 25 01 CA
0360 1B 01 11 36 01 CD 1B 01 0E 0F 11 5C 00 CD 05 00
0370 3C 20 0F 0E 16 11 5C 00 CD 05 00 3C 11 B7 01 CA
0380 1B 01 CD A6 04 FD 21 CD 02 3E 01 FD 77 03 AF FD
0390 77 05 FD 77 04 3E 06 FD 77 01 3E 15 FD 77 06 11
03A0 D7 02 0E 1A CD 05 00 11 C9 01 CD 1B 01 FD CB 05
03B0 C6 1E 2E CD 20 01 3E 0A FD 77 00 11 D4 02 06 84
03C0 FD 4E 06 CD EE 04 CD B9 04 30 2C FD 35 00 20 F6
03D0 FD 35 01 20 DC 11 41 02 C3 70 04 CD B9 04 30 17
03E0 3E 83 B8 20 07 1B 1A FE 04 CA 85 04 06 05 CD F8
03F0 04 11 66 02 C3 6C 04 12 13 10 E0 11 E0 01 CD 06
0400 01 DD 21 D4 02 11 FE 01 3E 01 DD BE 00 20 5D DD
0410 7E 02 2F DD BE 01 20 54 FD BE 04 28 35 11 2C 02
0420 FD BE 03 20 4B AF DD 21 D7 02 06 80 DD 86 00 DD
0430 23 10 F9 11 14 02 DD BE 00 20 31 0E 15 11 5C 00
0440 CD 05 00 B7 11 52 02 20 27 FD 7E 03 FD 77 04 FD
0450 34 03 DD 21 F0 01 CD 14 05 11 F0 01 3E 06 FD 77
0460 06 3E 01 FD 77 01 CD 06 01 C3 B6 03 3E 15 18 EE
0470 CD 06 01 0E 18 CD EE 04 06 03 CD F8 04 CD EE 04
0480 11 B4 02 18 0E 11 85 02 CD 06 01 0E 06 CD EE 04
0490 11 95 02 CD 06 01 0E 10 11 5C 00 CD 05 00 11 80
04A0 00 0E 1A C3 05 00 3A AB E0 E6 0F F6 10 32 AB E0
04B0 3E 09 32 AA E0 3A A8 E0 C9 21 A9 E0 CB 5E 20 24
04C0 DD 21 12 05 AF DD 77 00 3E 62 DD 77 01 CB 5E 20
04D0 13 3A 00 E0 B7 FA E8 04 DD 35 00 20 F0 DD 35 01
04E0 20 EB 37 C9 2B 7E B7 C9 32 10 E0 E6 7F C9 21 A9
04F0 E0 CB 66 28 FC 2B 71 C9 DD 21 12 05 AF DD 77 00
0500 3E E4 DD 77 01 DD 35 00 20 FB DD 35 01 20 F6 10
0510 EB C9 CD 96 6F 26 00 11 64 00 CD 40 05 11 0A 00
0520 CD 40 05 3E 30 B5 DD 77 00 DD 7E FE FE 30 C0 3E
0530 20 DD 77 FE DD 7E FF FE 30 C0 3E 20 DD 77 FF C9
0540 AF ED 52 38 03 3C 18 F9 19 F6 30 DD 77 00 DD 23
0550 C9 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

**Dump hexadécimal de XMODR.COM**

```

0100 00 00 00 C3 58 03 FD CB 05 46 0E 09 CA 05 00 D5
0110 11 DB 01 FD CB 05 86 CD 1B 01 D1 0E 09 C3 05 00
0120 0E 02 C3 05 00 0D 0A 42 61 64 20 43 6F 6D 6D 61
0130 6E 64 2E 2E 2E 24 1B 2A 20 2D 2D 2D 2D 2D 2D
0140 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D
0150 2D 2D 2D 2D 2D 0D 0A 20 58 4D 4F 44 45 4D 20 20
0160 20 44 6F 77 6E 6C 6F 61 64 69 6E 67 20 55 74 69
0170 6C 69 74 79 0D 0A 20 4A 46 2D 52 20 20 20 20 20
0180 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 31
0190 39 38 36 0D 0A 20 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D
01A0 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D
01B0 2D 2D 0D 0A 0D 0A 24 43 61 6E 27 74 20 6F 70 65
01C0 6E 20 46 69 6C 65 0D 0A 24 50 72 6F 6D 70 74 69
01D0 6E 67 20 53 65 6E 64 65 72 20 24 0D 0A 0D 0A 24
01E0 20 52 65 63 65 69 76 65 64 20 42 6C 6F 63 20 24
01F0 78 78 78 20 20 20 20 4F 6B 0D 0A 0D 0A 24 20 20
0200 20 20 20 20 20 48 65 61 64 65 72 20 45 72 72 6F
0210 72 0D 0A 24 20 20 20 20 20 20 20 43 68 65 63 6B
0220 73 75 6D 20 45 72 72 6F 72 0D 0A 24 20 20 20 20
0230 20 20 20 46 61 74 61 6C 20 45 72 72 6F 72 0D 0A
0240 24 20 20 20 20 20 20 20 54 69 6D 65 6F 75 74 0D
0250 0A 24 20 20 20 20 20 20 44 69 73 6B 20 45 72
0260 72 6F 72 0D 0A 24 20 20 20 47 6F 74 20 73 6F
0270 6D 65 74 68 69 6E 67 2C 20 62 75 74 20 3F 20 3F
0280 20 3F 0D 0A 24 20 52 65 63 65 69 76 65 64 20 45
0290 4F 54 0D 0A 24 0D 0A 46 69 6C 65 20 72 65 63 65
02A0 69 76 65 64 20 73 75 63 63 65 73 73 66 75 6C 6C
02B0 79 0D 0A 24 0D 0A 54 72 61 6E 73 6D 69 73 73 69
02C0 6F 6E 20 61 62 6F 72 74 65 64 0D 0A 24 00 00 00
02D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

**12 fonctions de base,  
200 disques,  
500 programmes...**

**Capture le catalogue,  
Imprime,  
Archive,  
Retrouve les programmes de vos  
disquettes des trois DOS**

180,00 F franco France métropolitaine

**Badaroux C.A.**

144, rue Legendre - 75017 Paris

Tél. : (1) 47 57 91 11

Apple //c et ImageWriter

# Cette fois Gutenberg n



*Gutenberg 1395-1468  
a inventé l'imprimerie.*

Gutenberg et ses associés, des gens très perfectionnistes au demeurant, n'hésitant pas à regarder à la loupe le moindre détail, n'avaient pourtant fait que la moitié du travail.

Le pouvoir de l'édition c'est bien, pouvoir éditer soi-même, c'est mieux.

C'est d'ailleurs le seul moyen de pouvoir parvenir à la vraie liberté d'expression. Cela Gutenberg n'y avait pas songé.

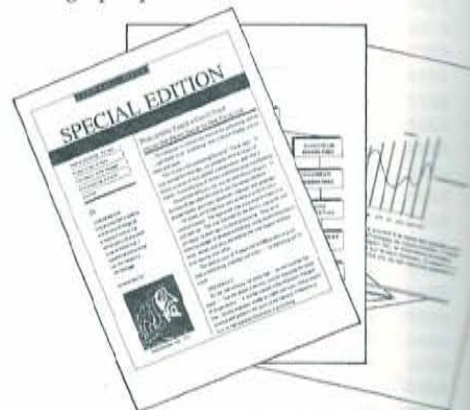
En toute bonne foi, il croyait fermement dans les vertus de la vis à bois et d'une imprimerie pour tout le monde.

Chez Apple, nous croyons aux ressources de l'individu et à l'édition personnelle.

Par contre pour créer son atelier, Macintosh a fait comme Gutenberg, il s'est associé avec la "LaserWriter", l'imprimante à laser d'Apple. A la seule différence que si 31 assistants se bousculent autour d'une presse, 31 Macintosh reliés par AppleTalk se partagent aisément une LaserWriter.

Mais, malgré son million d'octets de mémoire morte (extension LaserWriter Plus) et sa résolution de 130 points/cm, la LaserWriter ne serait qu'une version

légèrement améliorée de la presse en bois sans la puissance et les capacités graphiques de Macintosh Plus.



# aura pas le dernier mot.

C'est-à-dire, un méga-octet de mémoire extensible à 4 méga-octets, un nouveau lecteur de disque interne double face 800K et en option un disque dur de 20 méga-octets, ce qui permet de stocker des milliers de pages de documents.

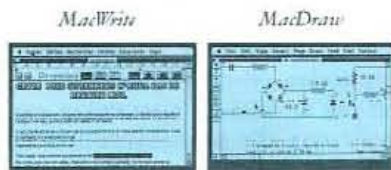
Et comme toujours, Macintosh Plus met à votre disposition tous ses fameux outils de bureau, pour couper, coller, remodeler le texte, choisir les caractères, mélanger texte et dessin, etc.

Le texte si brillant soit-il ne suffit pas. Avec MacPaint et MacDraw pour illustrer, tracer des schémas et des dessins techniques, définir des cadres, avec un logiciel comme Page Maker pour organiser et mettre en page, vos rapports d'entreprise, vos formulaires, vos manuels, votre journal interne, pour ne citer qu'eux, laisseront de vous une excellente impression. Et vous pourrez toujours tout modifier, y compris à la dernière minute!

Et si vous voulez vraiment faire les choses en grand, Macintosh peut aussi se connecter directement à une photocomposeuse d'imprimerie.



MacDraw



MacWrite

MacDraw

Pour Apple, il n'y a pas de petits et de grands écrivains, de littérature de bureau et de littérature tout court: tout le monde est logé à la même enseigne, celle de la qualité.

Avec la LaserWriter, une impression impeccable est à la portée de tous. Elle vous propose en effet onze familles de typographies (extension LaserWriter Plus), imprime sur papier, sur calque ou sur transparents et fournit des documents de qualité bromure.



En fait, Apple ne vous offre rien de plus que votre imprimeur, sauf que vous n'avez pratiquement plus besoin de lui.

Désormais, c'est vous, l'éditeur, qui éprouverez la sensation de Gutenberg il y a 436 ans lorsqu'il contempla son premier document.

C'est ainsi qu'Apple vous offre le meilleur de vous-même.

Au commencement de l'édition, il y a l'écriture: avec des logiciels de traitement de texte comme MacWrite, Word ou Writer Plus, le vrai problème des auteurs c'est l'inspiration.

Page Maker

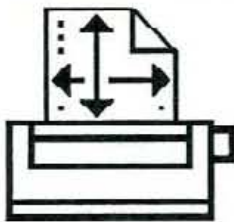


LaserWriter



Apple

Macintosh est une marque déposée d'Apple Computer, Inc. Apple, le logo Apple, MacWrite et MacDraw sont des marques déposées d'Apple Computer, Inc. Page Maker est une marque déposée d'Apple Computer, Inc.



# MISE EN PAGE

## GEORGES ZWINGELSTEIN

Vous avez sans doute remarqué que, lorsqu'une application ouvre un nouveau document, son format d'impression initial est rarement celui souhaité (il s'agit généralement du format 'Lettre américaine'). Ayant souvent oublié de choisir immédiatement le format A4, nous avons parfois dû refaire totalement la mise en page d'un texte, le format 'Lettre américaine' étant plus court que le format 'Lettre A4'.

Nous avons donc décidé de nous attaquer au problème, avec l'aide de MacsBug et REdit. La solution en fin de compte est assez simple :

- dans le document ImageWriter (ou LaserWriter) se trouvent trois ressources de type 'PREC'. Deux d'entre-elles ont la même longueur que le 'Print Record' utilisé par la plupart des routines du 'Printing Manager' (la troisième contient les dimensions et noms des différents formats de papier) ;
- la ressource numéro 1 contient le 'Print Record' utilisé pour afficher le dialog du 'Job', c'est-à-dire la qualité d'impression, etc. Celui-ci ne pose pas

**ImageWriter** v2.3 OK

Papier :  Lettre US  Format A4 Annuler  
 Légal US  Format B5  
 Papier informatique

Orientations :  Portrait  Paysage

Effets spéciaux :  Portrait ajusté  
 Réduction 50 %  
 Pas de saut de page

**LaserWriter** OK

Papier :  Lettre américaine  Format A4 Réduire  
 Légal américain  Format B5 ou 85 %  
 Format D5 Agrandir

Orientations :  Portrait  Paysage

Substituer les caractères ?  FINITION ? Annuler

de problème puisqu'il est enregistré à chaque utilisation ;

- la ressource numéro 0 contient le 'Print Record' qui nous intéresse : celui du dialogue du *style* (Format d'impression, orientation, etc.) qui, lui, n'est pas modifié mais est utilisé par l'application afin de connaître la taille d'une page dans le nouveau document.

La mini-application proposée ici ne fait que charger cette ressource, s'en servir pour afficher le dialogue du *style* et enfin enregistrer le résultat, modifié par les sélections dans le dialogue, en remplaçant l'ancienne ressource.



### Source 'Mep.Asm'

```

;*****
; Fichier 'Mise en Page.Asm'
;-----
; Modifie la ressource contenant la
; mise en page par défaut
; (format du papier, orientation
; etc.)
;-----
; G. Zwingelstein
; 19/9/86
;-----

Include DrEqu.Txt

.TRAP _InitGraf          $A86E
.TRAP _InitFonts        $A8FF
.TRAP _FlushEvents      $A032
.TRAP _InitWindows      $A012
.TRAP _InitMenus        $A930
.TRAP _InitDialogs      $A97B
.TRAP _TEInit           $A9CC
.TRAP _InitCursor       $A850
.TRAP _GetResource       $A9A0
.TRAP _HNoPurge         $A04A
.TRAP _ChangedResource  $A9AA
.TRAP _WriteResource    $A9B0
.TRAP _HPurge           $A049
.TRAP _GetNewDialog     $A07C
.TRAP _DisposeDialog    $A093
.TRAP _GetDItem         $A98D
.TRAP _SetDItem         $A98E
.TRAP _PenSize          $A898
.TRAP _InsetRect        $A8A9
.TRAP _FrameRoundRect  $A8B0

```

```

.TRAP _ShowWindow       $A915
.TRAP _ModalDialog      $A991
.TRAP _ResError         $A9AF
.TRAP _HLock            $A029
.TRAP _HUnLock          $A02A
.TRAP _ParamText        $A98B
.TRAP _StopAlert        $A986
.TRAP _Pack7            $A9EE

```

```

; Constantes
okButton EQU 1
numToString EQU 0

```

```

; Macro
;-----
;MACRO _PackCall
MOVE $1,-(SP)
$2
;ENDM
;MACRO _NumToString
_PackCall $NUMTOSTRING,_Pack7
;ENDM

```

```

; Programme -----
Start
BSR _InitManagers
CLR.B -(SP)
MOVE $128,-(SP)
JSR _Dialog
MOVE.B (SP)+,D0
BEQ $1
BSR _GetDefault
$1 RTS

```

```

; Initialise les 'Managers' -----
InitManagers
PDA -4(A5)
_InitGraf

```

```

_InitFonts
MOVE.L #$0000FFFF,D0
_FlushEvents
_InitWindows
_InitMenus
CLR.L -(SP)
_InitDialogs
_TEInit
_InitCursor
RTS

```

```

; Dialogue -----
; Affiche le dialogue de Numéro
; DialogID et retourne vrai si Bouton
; 'OK' Choisi, faux sinon.
; Le Bouton 'OK' doit être le premier
; dans la définition du dialogue.
; Il doit y avoir un UserItem en 3ème
; position.

```

```

; Variables dans la pile :
; DialogID EQU 8
; ItemHit EQU 10
; Variables locales :
; DialogPtr EQU -4
; Type EQU -6
; Item EQU -10
; Box EQU -18

```

```

Dialog
LINK A6,#-18
CLR.L -(SP)
MOVE 8(A6),-(SP)
CLR.L -(SP)
MOVE.L #-1,-(SP)
_GetNewDialog
MOVE.L (SP)+,-4(A6)

```



```
; Initialise ProcPtr du UserItem vers
; Procédure qui entoure le bouton par
; défaut.
```

```
MOVE.L -4(A6),-(SP)
MOVE #3,-(SP)
PEA -6(A6)
PEA -10(A6)
PEA -18(A6)
GetDItem
MOVE.L -4(A6),-(SP)
MOVE #3,-(SP)
MOVE -6(A6),-(SP)
PEA OutLineButton
PEA -18(A6)
SetDItem
```

```
ModalLoop ;attend une action
CLR.L -(SP)
PEA 10(A6)
ModalDialog
CMP #okButton,10(A6)
SEQ 10(A6)
```

```
;Fin du dialog
MOVE.L -4(A6),-(SP)
DisposeDialog
UNLK A6
MOVE.L (SP)+,A0
ADDQ.L #2,SP
JMP (A0)
```

```
-----
; Entoure le bouton par défaut (le
; numéro 1)
; cf Inside T-407 $2 et T-405 $16-18
; Variables dans la pile :
; itemNo EQU 8
; WindowPL EQU 10
; Variables Locales :
; Type EQU -2
; ItemHandle EQU -4+Type
; Box EQU -8+ItemHandle
```

```
OutLineButton
LINK A6,#-14
MOVR L 10(A6),-(SP)
MOVE #1,-(SP)
PEA -2(A6)
PEA -6(A6)
PEA -14(A6)
GetDItem
MOVE #5,-(SP)
MOVE #3,-(SP)
PenSize
PEA -14(A6)
MOVE #4,-(SP)
MOVE #4,-(SP)
InsetRect
PEA -14(A6)
MOVE #16,-(SP)
MOVE #16,(SP)
FrameRoundRect
UNLK A6
MOVE.L (SP)+,A0
ADDQ.L #6,SP
JMP (A0)
```

```
-----
; GetDefault
; Affiche le dialog de la mise en page
; et enregistre les modif.
; Variable locale
hPrintRec EQU -4
```

```
GetDefault
LINK A6,hPrintRec
JSR PrOpen
```

```
; Récupère ressource de type 'PREC' du
; fichier ImageWriter.
```

```
CLR.L -(SP)
MOVE.L #'PREC',-(SP)
MOVE #0,-(SP)
GetResource
MOVE.L (SP)+,hPrintRec(A6)
BNE @1
```

```
; Erreur chargement ressource.
MOVE #1,(SP)
CLR -(SP)
JSR StopAlert
```

```
BRA @10
; Protège la ressource 'PREC'
@1 MOVE.L hPrintRec(A6),A0
_HNOFpurge
; Valide la ressource 'PREC'
CLR.B -(SP)
MOVE.L hPrintRec(A6),-(SP)
JSR PrValidate
MOVE.D (SP)+,D0
; Affiche le dialog de la mise en page
CLR.B -(SP)
MOVE.L hPrintRec(A6),-(SP)
JSR PrStdDialog
MOVE.D (SP)+,D0
BEQ @9
```

```
; Marque la ressource comme modifiée
MOVE.L hPrintRec(A6),-(SP)
_ChangedResource
CLR -(SP)
_ResourceError
MOVE (SP)+,D0
BEQ @7
```

```
; Erreur fonction ChangedResource
; (disquette protégée en écriture ou
; fichier protégé)
```

```
CMP #-44,D0
BEQ @4
CMP #-45,D0
BEQ @3
CMP #-46,D0
BEQ @2
```

```
; Erreur Inconnue
MOVE #10,-(SP)
MOVE D0,-(SP)
JSR StopAlert
BRA @9
```

```
; Le volume est verrouillé
```

```
@2 MOVE #2,-(SP)
CLR -(SP)
JSR StopAlert
BRA @9
```

```
; Le fichier est verrouillé
```

```
@3 MOVE #3,-(SP)
CLR -(SP)
JSR StopAlert
BRA @9
```

```
; Disquette protégée en écriture
```

```
@4 MOVE #4,-(SP)
CLR -(SP)
JSR StopAlert
BRA @9
```

```
; Enregistre la ressource
```

```
@7 MOVE.L hPrintRec(A6),-(SP)
_WriteResource
CLR -(SP)
_ResourceError
MOVE (SP)+,D0
DDQ @9
MOVE #5,-(SP)
MOVE D0,-(SP)
JSR StopAlert
```

```
; Débloque la ressource
```

```
@9 MOVE.L hPrintRec(A6),A0
_HNofpurge
@10 JSR PrClose
UNLK A6
MIS
```

```
; StopAlert
; Affiche la stopAlert d'ID 128 avec
; message dont l'ID est dans la pile
; et No de l'erreur (si non nul)
; Variables dans la pile
```

```
; ErrNum EQU 8
; StrID EQU 10
; Variables locales
; String PQH -16
StopAlert
```

```
LINK A6,#-16
MOVE.L A2,-(SP)
MOVE B(A6),D0
```

```
BEQ @1
EXT.L D0
LEA -16(A6),A0
_NumTostring
BRA @2
@1 ; Chaine de longueur nulle
MOVE.D #0,-16(A6)
@2 CLR.L -(SP)
MOVE.L #'STR',-(SP)
MOVE 10(A6),-(SP)
GetResource
MOVE.L (SP)+,A2
MOVE.L A2,A0
_HLock
```

```
; Insère le texte dans l'alerte
```

```
MOVE.L (A2),-(SP)
PEA -16(A6)
CLR.L -(SP)
CLR.L -(SP)
```

```
_ParamText
```

```
; Affiche la fenêtre d'alerte
; appliquant l'annulation
```

```
CLR -(SP)
MOVE #128,-(SP)
CLR.L -(SP)
_StopAlert
MOVR (SP)+,D0
MOVE.L A2,A0
_HUnLock
```

```
; Retour
```

```
MOVE.L (SP)+,A2
INTK A6
MOVE.L (SP)+,A0
ADDQ.L #4,SP
JMP (A0)
```

```
END
```

## Fichier 'Mep.R'

Note : le caractère 'f' indique la continuité de la ligne.

```
*
* Ressources pour le programme
* Mise en Page
*
```

```
[Mep
ADPLMYAD
```

```
Type MYAP - STR
; 0
Mise en Page - G. ZWINGELSTEIN
```

```
Type BNDL
,128(32)
MYAP 0
ICN#
0 128
PREF
0 120
```

```
Type rNER
,128(32)
APPL 0
```

```
Type ICN# = GNRL
```

```
,128(32)
```

```
00000000 00000000 07FFC000 04086000
051C5000 043E4800 05084400 04084200
05087F00 04080100 04080900 05000D00
07EBFF00 05880D00 04880900 043E0100
051C0500 04080100 07FFFFFF 9000004F
9FFFFFFC9 90000049 90000049 9000004F
9FFFFFFC8 80000008 80000008 80000008
80000008 FFFFFFFF8 80000008 FFFFFFFF8
*
00000000 00000000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
07FFC000 07FFC000 07FFC000 07FFC000
```

\* fenêtre d'alerte si erreur système  
 Type ALRT  
 ,128  
 50 50 250 450  
 128  
 5555

\* DITL de la fenêtre d'alerte  
 Type DITL  
 ,128  
 2

Button  
 170 150 190 250  
 OK

StaticText  
 60 20 150 300  
 ^0^1^2^3

\* message d'erreur de la fenêtre/  
 d'alerte

Type STR  
 ,1  
 Le document ImageWriter n'est pas  
 compatible avec cette  
 application.

,2  
 Désolé: la disquette est verrouillée.  
 Les nouvelles sélections ++  
 ne peuvent être enregistrées.

,3  
 Désolé: le document ImageWriter est/  
 verrouillé. Les nouvelles ++  
 sélections ne peuvent être/  
 enregistrées.

,4  
 Les nouvelles sélections n'ont pas pu/  
 être enregistrées. ++  
 Veuillez déprotéger la disquette de/  
 démarrage et recommencer.

,5  
 Désolé. Une erreur imprévue est/  
 survenue lors de/  
 l'enregistrement ++

des nouvelles sélections. Erreur No :

,10  
 Une erreur système imprévue est/  
 survenue. Erreur No :

\* Dialog des instructions  
 Type DLOG  
 ,128  
 dialog box  
 50 10 532 502  
 Visible NoGoAway  
 1 ;dBoxProc (alert box or modal/  
 dialog box)

0  
 129

\* DITL du dialog des instructions  
 Type DITL  
 ,129  
 12

Button  
 270 80 290 180  
 OK

Button  
 270 280 290 380  
 Annuler

UserItem Disabled  
 0 0 0 0

StaticText Disabled  
 10 5 10 482  
 Cette application permet de modifier/  
 le format d'impression/  
 pré-établi ++  
 utilisé dans la plupart des/  
 applications (voir l'article/  
 'mise en page' ++  
 ou 'format d'impression' du menu/  
 Fichier) par les nouveaux/  
 documents.

StaticText Disabled  
 10 5 120 482  
 Si vous cliquez sur la case 'OK',

cette application affichera ++  
 le dialogue au format d'impression/  
 pré-établi (contenu dans le ++  
 document ImageWriter de la disquette/  
 de démarrage) :

StaticText Disabled  
 120 40 190 482  
 Il suffira alors de choisir votre/  
 format habituel ++  
 et cliquer sur la case 'OK' pour/  
 l'enregistrer comme nouveau/  
 format ++  
 d'impression pré-établi ou sur la/  
 case 'Annuler' pour conserver/  
 l'ancien.

StaticText Disabled  
 190 5 230 482  
 Si vous préférez quitter/  
 immédiatement cette/  
 application, cliquez ++  
 sur 'Annuler'.

StaticText Disabled  
 240 5 260 482  
 © G. Zwingelstein et Pom's - 1986

## Fichier 'Mep.Job'

ASM	Mep.Asm	Exec	Edit
LINK	Mep.Link	Exec	Edit
RMaker	Mep.R	Finder	Edit

## Fichier 'Mep.Link'

1  
 Mep  
 PrLink  
 \$

# Mac Astuces

## Plus de 24 points avec MacWrite

Mais oui c'est possible... Il suffit pour cela de disposer d'un éditeur de fichiers - FEdit par exemple - et de quelques minutes pour les manipulations.

Voici la méthode :

- tapez un mot - par exemple 'MOT' - et donnez-lui la taille 24 points ;
- enregistrez le petit fichier ainsi constitué ;
- avec votre éditeur de fichiers, visualisez le contenu du document : on voit apparaître le 'MOT' ;
- le sixième octet après 'MOT' indique la taille du texte (ici 518 pour 24 en décimal) ;
- remplacez la valeur par 532 (48 en décimal) ;
- lorsque l'on appellera à nouveau le fichier

depuis MacWrite, 'MOT' sera désormais affiché en 48 points, il est donc possible de le sélectionner et de taper ce que l'on veut à la place.

Cette méthode autorise aussi les tailles de caractères "hors normes" comme, par exemple, 8 ou 47 points.

## MacPaint et FullPaint

Si vous maintenez la touche Option enfoncée lors du tracé d'une surface remplie (rectangle, oval), le bord de la surface ne sera pas tracé.

Si vous voulez totalement remplacer MacPaint par FullPaint, au point qu'un double clic sur une image MacPaint ouvre FullPaint, il faut :

changer PANT en MNPT dans la ressource BNDL FullPaint, utilisez Lire les Informations de ResEdit pour effectuer la même modification et mettre en fonction le bit de liaison. Après avoir quitté ResEdit, effectuez un changement de dossier pour faire reconnaître au Bureau la modification du type. Et voilà...

*N.D.L.R. : en procédant de cette façon, un double clic sur un document FullPaint créé avec une version non modifiée n'ouvrira pas ledit document. Une autre solution, plus simple et moins définitive, consiste à charger le document MacPaint depuis FullPaint, et demander Enregistrer sous... le même nom.*

## La mémoire du +...

...peut être visualisée si vous appuyez sur la touche "INTEKRUPT" placée sur le côté gauche du Macintosh Plus :

- le 'prompt' > apparaît ;
- tapez DM 1000 ;
- le contenu de la mémoire, de l'adresse \$1000 à \$105F s'affiche.

Il est aussi possible de modifier le contenu de la mémoire (attention, danger...) en tapant 'SM', l'adresse (en hexa) et la ou les valeurs à placer à partir de cette adresse.

Pour reprendre le cours normal de vos activités, tapez 'G'.

## Tempo

Depuis très longtemps, nous rêvions d'un accessoire de bureau qui permettrait d'enregistrer tout un traitement clavier/souris et de le rappeler avec une simple combinaison de touches. Il y a bien eu MacTracks, mais cet accessoire ne cessait de faire la bombe. Aussi étions-nous ravis de voir débarquer Tempo, l'accessoire de bureau pour la création et l'utilisation de macro-instructions.

Tempo permet d'enregistrer, en temps réel ou en mode programmation, toute une séquence d'utilisation du clavier et de la souris. Cette séquence peut être ensuite modifiée à loisir, puis exécutée sur l'appel d'une touche associée à la touche *Commande*.

### Quelques critiques

En fait, une fois activé, Tempo se loge à droite de la barre des menus, avec le symbole "Commande". D'où une première incompatibilité avec les logiciels qui, tel Excel, remplissent cette ligne. En fait, l'incompatibilité vient ici d'Excel, puisque tout le monde sait maintenant que les applications doivent laisser une place dans la barre des menus, pour un éventuel accessoire justement. Certains logiciels fort connus, tels ReadySetGo, Switcher ou ThinkTank, sont allergiques à Tempo : félicitons ici l'initiative de l'importateur qui fournit une première liste de ces allergies. Avec PageMaker, Tempo exige l'original, ne permettant même pas qu'on l'introduise seulement lors de la vérification de copyright. Enfin, Tempo est vite débordé quand on utilise les possibilités les plus intéressantes telles que la programmation de pauses ou de boucles : il affiche alors des messages totalement incohérents et se comporte bizarrement.

### Quelques avantages

Tempo est en fait un outil idéal pour accomplir sans difficultés et sans risque d'erreur des actions répétitives simples. Dans ce cas d'ailleurs, même les fonctions avancées (pauses, boucles et tests) fonctionnent de façon satisfaisante. Tempo est vraiment utile, pour prendre trois exemples :

- comme glossaire pour MacWrite ;
- pour effectuer des Recherches/Remplacer ou des changements de police et de style fréquents ;
- pour faciliter avec PageMaker une opération ennuyeuse telle que le choix de caractères.

### Une bizarrerie

Cette dernière utilisation fait toucher du doigt une particularité de Tempo. Supposons que, dans une fenêtre de dialogue, on fasse défiler une liste (par exemple, de polices de caractères), pour trouver celle que l'on veut, et qu'on la sélectionne. Tempo n'enregistre alors pas la police choisie, mais le mouvement de la souris. Le résultat est que, lors de l'utilisation, la police sélectionnée par la macro dépendra de la position antérieure, c'est-à-dire de la sélection précédente. Pour être francs, il faut dire que nous ne voyons pas très bien comment le programmeur de la chose aurait pu faire autrement.

Une astuce résout ce problème : quand on doit faire défiler une liste dans une fenêtre, il faut d'abord revenir au premier élément de la liste et le sélectionner, puis pratiquer la sélection voulue.

### Conclusion

C'est finalement un produit qui comporte des potentialités intéressantes mais qui, en dehors des exemples simples d'application présentés ci-dessus, risque de décevoir. Peut-être faut-il attendre une prochaine version ?

Créé par Affinity Software (Boulder, Colorado) et vendu par Alpha Systèmes avec une documentation en français.



## dMac III

Ce produit, malgré son nom et la compatibilité dBase III qu'il semble impliquer et qu'il revendique d'ailleurs dans sa publicité, ne provient pas d'Ashton Tate. Ce n'est pas non plus un clone logiciel, puisqu'il ne tourne pas sur le même matériel que le logiciel qu'il imite. Son objectif est clair : donner aux utilisateurs du Macintosh la possibilité à la fois de récupérer des fichiers dBase développés sur IBM PC/XT, et de retrouver la syntaxe et le vocabulaire de dBase auquel ils sont déjà familiarisés.

Ce programme donnera à ceux qui ne connaissent pas encore le manque de convivialité caractéristique de l'IBM PC et de dBase (par rapport à ses concurrents sur Mac) une assez bonne compréhension - par comparaison - de la grande convivialité du Mac.

La documentation est entièrement francisée, mais le travail a été fait un peu vite, ce qui donne parfois jusqu'à 10 fautes par page, entre le français, l'orthographe et la ponctuation. Pourquoi donc les éditeurs et importateurs français ont-ils tant tendance à sous-estimer l'importance d'une documentation bien finie ?

La première conséquence que l'on peut en tirer d'un essai de dMac III est que ce produit, vu son ambition de ressembler à dBase, ne peut exploiter pleinement le graphisme/souris du Mac. Deuxièmement, il est de notre devoir de souligner que la compatibilité avec dBase III n'est en aucun cas totale : les caractères graphiques de l'IBM PC, ainsi que les caractères accentués, ne se récupèrent pas tels quels. Les fichiers mémo ne sont pas récupérables non plus. Ajoutons que PC Diffusion International a l'honnêteté d'avertir l'utilisateur de ces limitations, et promet de résoudre ces problèmes. Enfin, la définition de la structure d'un fichier est celle de dBase II, et non pas celle de dBase III.

En conclusion d'une analyse rapide, disons que l'utilisateur de dBase ne sera pas dépaycé - contrairement au Macophile impénitent - et, qu'au prix de certaines modifications, il pourra récupérer ses fichiers dBase IBM ; notons au passage que rien n'est dit sur la récupération de fichiers Apple II, ce qui pourtant devrait aussi être possible sans difficulté supplémentaire. Il serait intéressant de voir comment sera le dBase Mac d'Ashton Tate, qui devrait être disponible



en France en début 87. Contrairement à dMac III, dBase Mac, compatible avec dBase III, doit sortir en fin d'année; c'est une base de donnée relationnelle complète, permettant d'ouvrir 36 fichiers à la fois, un produit concurrent d'Omni III, de 4<sup>ème</sup> Dimension et du plus récent REFLEX de Borland, lui aussi bientôt disponible en français.

PC Diffusion International, 9 bis, rue Casimir Pinel, 92200 Neuilly. Prix : 3.950 F HT.



## FullPaint

C'est bien simple : si vous aimez MacPaint, vous adorerez FullPaint. Voici enfin programme totalement compatible avec MacPaint (c'était la moindre des choses !), et qui permet de faire tout ce que fait MacPaint, plus :

- travailler sur des images en plein écran ;
- avoir plusieurs documents ouverts à la fois ;
- effectuer des rotations et transformations diverses ;
- déplacer ou cacher les outils et les motifs ;
- utiliser des règles et des pointeurs de position ;
- coller depuis l'Album des documents plus grands que la fenêtre ;
- imprimer avec l'ImageWriter dans les formats 50%, 100%, 200% et 400% ;
- imprimer avec la LaserWriter jusqu'à 400%.

Seul inconvénient majeur : il ne se charge pas automatiquement quand on clique dans un document MacPaint. Et encore, la solution à ce problème vous est proposée dans la rubrique MacAstuces...

Enfin, à l'aide du 'driver' ColorPrint disponible sur la disquette FullPaint, vous pouvez imprimer des documents FullPaint en couleur, avec de nombreuses imprimantes, dont, bien entendu les ImageWriter I (on peut y mettre des rubans de couleur) et II, Epson JX-80, Canon à jet d'encre ou Scribe, pour ne citer que les plus connues.

### Conclusion

Un produit que nous recommandons vigoureusement et sans réserve, ce qui est hélas un plaisir trop rare.

Diffusé par Ann Harbor Softworks, 308<sup>1/2</sup> S. State Street, Ann Harbor, MI 48104, USA. Version US.



Dans ce numéro, Pom's propose deux programmes effectuant à peu près les mêmes tâches. L'un fonctionne sur les Apple II, l'autre - celui qui correspond à cet article de présentation - sur les Macintosh ; les deux utilisant le Minitel. Afin de ne pas mettre deux fois la même chose dans la revue, il nous a semblé préférable de placer les généralités sur le Minitel ainsi que les problèmes qui lui sont liés dans un article commun que vous pourrez trouver, si cela vous intéresse, page 53.

Un schéma vous indique le câblage nécessaire à la réalisation, très simple (nous joignons un petit dessin qui donne la correspondance entre les transistors sur le schéma et leur apparence physique, ce qui rend le montage accessible à tous), de l'"interface" reliant le Minitel au Macintosh ou au Macintosh plus. Ce dernier utilise une fiche 'Mini-DIN' qui n'est pas encore en vente partout ; si vous avez des problèmes pour vous en procurer une, il vous sera toujours possible d'employer la solution Macintosh 128 ou 512Ko avec, en plus, un adaptateur Mac/MacPlus (DB9 femelle/Mini-DIN mâle). L'autre solution consiste en l'achat d'une "interface" du commerce.

Le programme emploie plusieurs menus ; le menu 'Mode' est le principal. Voici le premier de ces modes de fonctionnement :

<b>Mode</b>
<input checked="" type="checkbox"/> Stockage
Consultation
Copie d'écran
Préparation/envoi

Ce mode sera utilisé pour 'enregistrer' rapidement les informations qui arrivent sur l'écran du Minitel. Pour cela, il faut utiliser le clavier et la souris du Macintosh à la place du clavier du Minitel (on ne perd pas au change !). Les commandes Minitel (Sommaire, Guide, etc.) sont obtenues par l'action sur des boutons affichés à l'écran ; certaines d'entre-elles sont accessibles par des "équivalents clavier" :

Envoi	'Retour chariot' ou 'Entrée'
Suite	'Tabulation'
Correction	'BackSpace'
Annulation	'Annulation'

(uniquement MacPlus)

Le but du jeu est donc d'enregistrer les écrans ; il faut pour cela ouvrir un fichier sur une disquette (ou un disque dur) à l'aide du menu fichier et de son article :

<b>Fichier</b>
Nouveau...
Ouvrir...
Enregistrement
Fermer
Quitter

Cet article affiche la célèbre fenêtre de choix du nom de fichier. Cette dernière se comportant d'une manière tout à fait

classique, ajouter quelque chose sur le sujet ne serait que gâchis de papier.

Lorsqu'un fichier est ouvert, il devient impossible de changer de mode et il en sera ainsi jusqu'à ce que le fichier soit fermé. Ah! mais...

<b>Fichier</b>
Nouveau...
Ouvrir...
<b>Enregistrement</b>
Fermer
Quitter

L'article magistralement visualisé ci-dessus permet de mettre en ou hors fonction l'enregistrement, ceci afin d'éviter le stockage d'écrans indésirables comme, par exemple, les premiers écrans de l'annuaire électronique. Il s'agit en fait de l'équivalent de la touche "Pause" d'un magnétophone.

Lorsque l'enregistrement est en cours, l'article 'Enregistrement' apparaît "coché" et, simultanément (parfois en même temps), un 'voyant' s'affiche à côté du chronomètre :



<b>Fichier</b>
Nouveau...
Ouvrir...
<input checked="" type="checkbox"/> Enregistrement
Fermer
Quitter

Les articles du menu 'Mode' redeviennent valides : il est possible de changer de mode, ou encore de débiter un nouvel enregistrement.

<b>Chronomètre</b>
<input checked="" type="checkbox"/> Départ automatique
Déconnexion automatique...
Marche
Mise à zéro

Le programme propose l'utilisation optionnelle - d'un chronomètre dont le but est double : donner une idée du temps passé sur un serveur et permettre une déconnexion automatique à l'expiration d'un temps donné. Puisqu'il y a déconnexion automatique, il fallait aussi une option "Départ automatique".

Quand cette option est choisie (article "coché"), le départ automatique aura lieu dès que le Minitel émettra quelque chose. Notons que, s'il est possible de rendre à tous moments cette option valide ou invalide, elle ne fonctionne que lorsqu'un fichier est ouvert et l'enregistrement autorisé. Si le chronomètre était en fonctionnement ou simplement n'était pas à zéro, le premier caractère reçu provoquera sa réinitialisation.

Pratiquement, une séance d'enregistrement avec départ automatique du chronomètre pourrait débiter ainsi (on considère que le Minitel est sous tension) :

- ouverture d'un nouveau fichier ;
- autorisation d'enregistrement ;
- demande de l'option "départ automatique" ;
- composition du 11 (annuaire électronique) ;
- connexion au réseau par sélection de l'article "connexion" du menu "Liaison".

La dernière opération provoque le déclenchement du chronomètre.

<b>Chronomètre</b>
<input checked="" type="checkbox"/> Départ automatique
Déconnexion automatique...
Marche
Mise à zéro

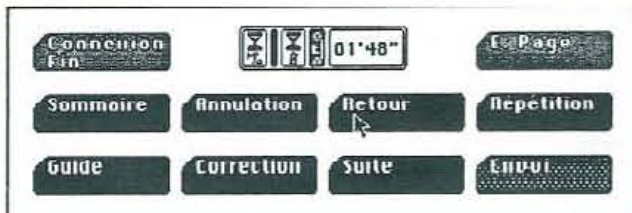
La "déconnexion automatique" pourra être utile à ceux qui ne veulent pas que leurs factures téléphoniques atteignent des sommets vertigineux. La sélection de l'article provoque l'affichage de la fenêtre de sélection suivante :

<b>Déconnexion automatique</b>
ou bout de <input type="text" value="3"/> minutes
<input type="button" value="OK"/>

La fenêtre propose, par défaut, trois minutes ; ceci correspond au temps pendant lequel l'annuaire électronique est gratuit dans les zones où il est accessible par le 11. Il est possible de choisir une durée allant de 1 à 59 minutes. Une fois le choix effectué, l'article apparaît coché avec, à sa suite, un rappel du nombre de minutes.

Remarques :

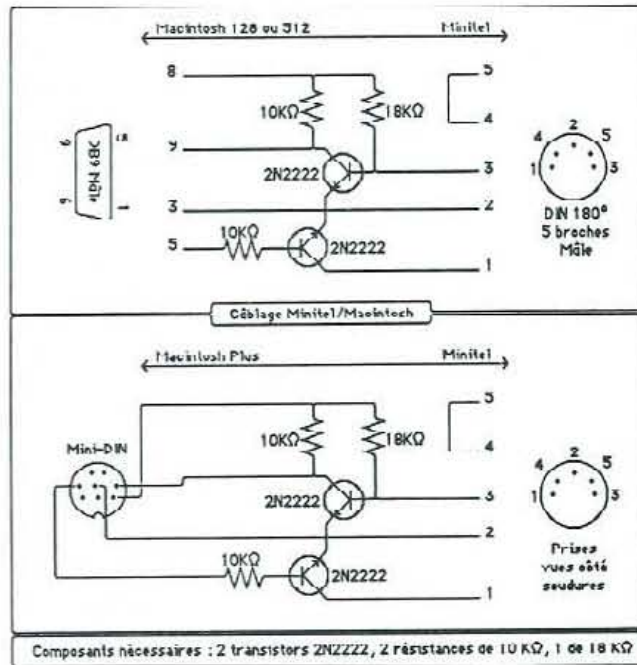
- le processus de déconnexion du réseau est entamé deux secondes avant l'expiration du délai, ceci afin de tenir compte de l'inertie de l'ensemble Mac/programme/Minitel (prononcez *chaîne télématique*, comme dans les salons où l'on cause) ;
- la déconnexion a lieu lorsque le chronomètre 'passe' au temps donné : si le chronomètre n'est pas à zéro lorsque l'option déconnexion automatique est demandée, le temps est faussé (on se croirait dans la quatrième dimension !). De même, si on demande un temps de déconnexion inférieur à la valeur courante du chronomètre, la déconnexion sera immédiate ;
- l'option est outrepassable : si la déconnexion n'est plus désirable, sélectionner à nouveau l'article "déconnexion automatique" ou demander l'arrêt du chronomètre ;
- bien que les options "déconnexion automatique" et "départ automatique" soient indépendantes, elles seront, la plupart du temps, utilisées de conserve.



**Fichier : Texte à transmettre**

Nombre de caractères: 752 **Envoi**

Envoyer un message à un serveur par l'intermédiaire du Minitel revient généralement assez cher, ceci à cause du clavier - pas vraiment ergonomique - et des possibilités d'édition (inexistantes). On passe facilement dix minutes ou plus à entrer un texte comme celui-ci. L'option "Préparation/Envoi" du programme proposé dans la numéro 27 de Pom's autorise une saisie préalable et aisée (Couper, Coller...) de textes qui pourront être par la suite (ou immédiatement) transmis au serveur, et ceci rapidement (un peu plus d'une minute pour ce texte). Le programme permet bien sûr la sauvegarde, le chargement et la modification des textes créés à partir de cette option, mais aussi le chargement de textes mis en forme avec n'importe quel éditeur de texte.



**Chronomètre**

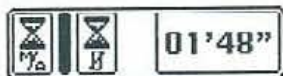
Départ automatique

Déconnexion automatique /3'

**Marche**

Mise à zéro

Permet la mise en marche du chronomètre et remplit le même rôle que le bouton 'M/A' placé dans la fenêtre (ceci fonctionne quelle que soit la valeur courante du chronomètre). Une fois sélectionné, l'article "Marche" devient logiquement "Arrêt", et un petit 'voyant' s'affiche entre les boutons 'M/A' et 'Ø'.



**Chronomètre**

Départ automatique

Déconnexion automatique /3'

**Arrêt**

Mise à zéro

Force à zéro le chronomètre sans l'arrêter. Il fonctionne comme le bouton 'Ø' placé à côté dans la fenêtre.

**Liaison**

**Connexion**

Déconnexion

1200 bauds

4800 bauds

Le premier article du menu "Liaison" est utilisé pour connecter le Minitel au réseau, après avoir fait le 11, le 36 15, etc.

**Liaison**

**Connexion**

Déconnexion

1200 bauds

4800 bauds

C'est le contraire, c'est-à-dire la déconnexion complète du réseau. Attention, le bouton "Connexion/ Fin" situé dans la fenêtre n'a pas la même

fonction que les articles du menu "Liaison" : il sera utilisé pour passer, par exemple, du niveau serveur au niveau Télétel.

Pour conclure le 'mode d'emploi' du mode "Stockage", il convient de signaler qu'un 'bip' juste après une commande signifie qu'elle n'a pu être prise en compte ; la demande doit être répétée.

**Mode**

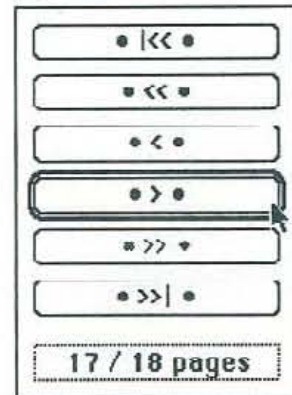
**Stockage**

Consultation

Copie d'écran

Préparation/envoi

Le mode "consultation" est employé pour renvoyer sur le Minitel les informations accumulées à partir du mode "Stockage". La fenêtre contient les boutons de commande, ainsi qu'une indication sur le nombre de pages dans le document actuellement ouvert et le numéro de la page courante.



**Fichier**

Nouveau...

Ouvrir...

Enregistrement

Fermer

Quitter

Permet d'ouvrir un fichier, uniquement s'il a été créé à partir du mode "stockage"

Une fois le fichier ouvert, les boutons de commandes sont rendus valides en fonction du nombre de pages du document. Leur signification :

- |<< • première page du document
- << • dix pages en arrière
- < • une page en arrière
- > • page suivante
- >> • dix pages en avant
- >>| • dernière page du document

Il est aussi possible de passer à la page suivante avec les touches "retour-chariot" et "entrée".

Note : afin d'éviter la confusion des écrans, ou 'vide' celui du Minitel avant chaque changement de page, sauf lorsqu'on passe à la page suivante.

**Liaison**

Connexion

Déconnexion

1200 bauds

4800 bauds

La vitesse de transmission par défaut vers le Minitel est de 1200 bauds ; si votre 'mini-terminal' le permet, il est possible de passer à 4800 bauds, ce qui donne un plus grand confort à la consultation. Bien que cela ne présente pas beaucoup d'intérêt, on peut changer de vitesse à tout moment.

Attention : de manière générale lorsque vous utilisez ce programme, et surtout si vous utilisez le changement de vitesse de transmission, il faut que le Minitel soit relié au Macintosh, et sous tension. Ceci garantit l'utilisation de vitesses identiques par les deux appareils.

## Mode

Stockage

Consultation

✓ Copie d'écran

Préparation/envoi

INFORMATIQUE (MATERIEL ET FOURNITURES DIVERS)		LES UTILIS ESSEURS	
1	3 Com (Sté) av Québec	(1)	09 07 10 93
Apple Computer France			
2	-av Océanie	(1)	69 28 22 00
3	-same adresse	(1)	69 28 12 11
Apple Computer France			
4	5 Axel (SA) av Québec	(1)	69 28 27 27
plus d'informations tapez le #1 choisi... puis > ENVOI page suivante > SUITE			
<input type="checkbox"/> Réception de copies active...			

Le mode "Copie d'écran" est utilisé conjointement aux fonctions du même nom, disponibles sur la plupart des modèles de Minitel :

Fnc-I puis F pour une copie d'écran avec caractères accentués ;

Fnc-I puis A pour une copie d'écran sans caractère accentué.

Pour que la réception de copie(s) fonctionne, il faut qu'un fichier - sur lequel seront dirigées les informations - soit ouvert, et que le bouton "Réception de copies actives..." soit 'on'. Ce dernier est nécessaire pour ne pas accumuler tout ce qui est envoyé par le Minitel entre deux copies d'écran.

Si plusieurs copies sont enregistrées sur le même fichier, elles sont séparées par deux 'sauts de ligne' et une ligne de '-'.

Pratiquement, pour faire une copie d'écran (on suppose qu'un fichier est ouvert), il faut :

- mettre le bouton "Réception de copies actives..." 'on' ;
- faire Fnc-I puis F, ou Fnc-I puis A sur le Minitel.

Si l'ordre n'est pas respecté, on peut avoir une copie tronquée et, en même temps, l'impression de ne plus "avoir la main" puisque le programme attend la fin de la page, qui ne viendra jamais. Dans ce cas, il convient d'envoyer à nouveau une copie qui, elle aussi, sera tronquée, mais déclenchera une protection du programme qui remettra tout en ordre.

Les fichiers constitués depuis ce mode sont exploitables depuis tous les programmes qui acceptent le format 'TEXT' (MacWrite, Word, Edit...). Afin de respecter la géométrie de l'écran, il convient, lors d'une impression, d'utiliser une police de caractères à chasse fixe (Monaco ou Courier).

## Mode

Stockage

Consultation

Copie d'écran

✓ Préparation/envoi

Ce mode sera utile à ceux qui ont fréquemment besoin d'émettre des messages assez longs vers des serveurs et, de manière plus générale, aux futurs utilisateurs du 'courrier électronique' qui sera prochainement mis au service du public par les PTT. Il permet l'émission de messages préparés 'hors réseau' - donc sans contrainte de temps - et, éventuellement, leur édition.

## Fichier

Nouveau...

Ouvrir...

Enregistrer...

Fermer

Quitter

Pour commencer l'édition d'un nouveau message.

## Fichier

NOUVEAU...

Ouvrir...

Enregistrer...

Fermer

Quitter

Ouverture d'un fichier contenant un texte édité avec ce mode ou un programme autorisant un sauvegardé en format 'TEXT'. Si le fichier en question contient plus de 960 caractères (un écran Vidéotex complet), le programme émet un 'Bip' et seuls des 960 premiers caractères sont chargés.

## Fichier

Nouveau...

Ouvrir...

Enregistrer...

Fermer

Quitter

Pour sauvegarder un texte. Si le texte ne porte pas de nom, le programme affiche la fenêtre de saisie habituelle. Comme pour le chargement si le texte à sauvegarder contient plus de 960 caractères, un 'Bip' est émis et seuls des 960 premiers caractères seront enregistrés.

Le bouton "Nombre de caractères" donne cette indication dans la 'case' placée au milieu de la fenêtre. Il peut être utile dans le cas de serveur acceptant seulement un nombre limité de caractères.

Le bouton "Envoi" émet les caractères contenus dans la fenêtre vers le serveur, ceci à raison de dix caractères par secondes.

## Remarques :

- les caractères sont filtrés et recodés si besoin est au moment de l'émission. Certains caractères 'exotiques' qui ne seraient pas reconnus par le serveur sont simplement supprimés. Ainsi, le caractère '¥' n'est pas transmis. Le

tableau de la page 54 indique les caractères valides ;

- la fenêtre de saisie du mode "Préparation/Envoi" n'accepte pas les 'retours-chariot'. Cependant, un 'retour-chariot' se trouvant dans un texte saisi avec un programme comme MacWrite sera envoyé au Minitel comme la commande "Suite" ;
- bien que les 'retours-chariot' ne soient pas acceptés directement par la fenêtre, les fonctions d'édition Couper, Copier et Coller étant valides, il est possible de coller un de ces caractères récupéré avec un accessoire ;
- la commande "Suite" - qui remplace un 'retour-chariot' - est suivie d'un délai d'environ une seconde et demi, ceci afin de laisser au serveur le temps de 'digérer' la ligne de texte.

## Fichier

Nouveau...

Ouvrir...

Enregistrer...

Fermer

Quitter

Quel que soit le mode de fonctionnement en cours, cet article ferme l'éventuel fichier ouvert et provoque un retour au Finder.

## Quelques généralités

Le programme est en deux parties, la première est baptisée 'Minitel/1' mais ce nom peut être changé. En revanche, la seconde partie doit impérativement s'appeler 'Minitel/2' (sinon, il faut modifier le 'CHAIN' situé à la dernière ligne de la première partie). Pour lancer l'exécution du programme : un double 'clic' sur la première partie. Si vous avez la disquette d'accompagnement de ce numéro, elle contient une version RunTime du Basic Microsoft.

## Sur un 128Ko

De grands efforts d'optimisation ont été faits afin que ce programme fonctionne aussi sur un Macintosh 128Ko. Cependant, le système se trouvant un peu à l'étroit, il est préférable d'utiliser un ancien fichier 'système' (numéro de version inférieur ou égal à 2.0). Les versions postérieures constituent quelquefois, avec le MSBasic 2, un mélange 'détonnant' (seulement sur un 128Ko).

## En HFS

À notre connaissance, il n'existe pas encore de version du Basic Microsoft compatible HFS. Il faudra donc, jusqu'à ce que la grande maison nous fournisse un Basic en état de marche, laisser la totalité des fichiers (interpréteur et programmes compris) au niveau 0.



# Programme 'Minitel'

## Première partie ('Minitel/1')

• Fait un peu de place si on est sur un Mac128Ko.

```
IF FRE(0) < 50000: THEN CLEAR, 20000, 2200
```

• Initialise les tableaux de variables et quelques variables utilisées plus loin.

```
DEFINT A-Z: DR=ABS(FRE(0) > 50000): T  
20=2000+30000*DR: T22=63+447*DR: D  
V=0: VS="12": CNS="Déconnexion aut  
omatique": CC$="Clic" pour conti  
nuer..."
```

```
DIM A1 (T22), I (T22), V (2), A (98), B (23  
2), C (7), D (269), E (19), A$ (9), F (3),  
G (3), H (3), J (26), K (26), L (26), M (22  
) , N (3), O (3), P (3), Q (3), B$ (5), R (3)  
, S (3), T (89), U (23), W (18), X (65), C$  
(4, 1), Y (3), Z1 (14), CR (2, 3), Z2 (4, 4  
) , PL (1)
```

• Code objet de la routine "Délai".

```
DATA #H4E56, 0, #H362E, 8, #H48C3, #H59  
4F, #HA975, #H281F, #HD883, #H594F, #  
HA975, #HB99F, #H62F8, #H4E5E, #H4E7  
5
```

• Code objet de la routine de chargement des curseurs (Pom's 18).

```
DATA #H4E56, 0, #H42A7, #H3F2E, 8, #HA9  
B9, #H2057, #H2E90, #HA851, #H4E5E, #  
H4E75
```

• Titres des boutons en mode "Consultation".

```
DATA * | << *, * << *, * < *, * > *, *  
>> *, * >> | *
```

• Positions et titres des boutons en mode "Stockage".

```
DATA 16, 16, Connexion, 56, 16, Sommair  
e, 96, 16, Guide, 56, 112, Annulation  
, 96, 112, Correction, 56, 208, Retour  
, 96, 208, Suite, 16, 304, E. Page, 56,  
304, Répétition, 96, 304, Envai
```

• Pour les polygones des boutons en mode "Stockage".

```
S2: DATA 54, 0, 0, 24, 88, 0, 9, 0, 86, 2, 88  
, 22, 88, 24, 86, 24, 2, 22, 0, 10, 0, 7, 1,  
1, 7, 0, 9
```

• Définition des boutons 'Marché/Arrêt' et 'Mise à zéro' en mode "Stockage".

```
DATA 16, 25, #H7FFE, #H8001, #H9FF9, #H  
9FF9, #H8811, #H8761, #H83C1, #H8181  
, #H8181, #H8241, #H84A1, #H8BD1, #H9  
FF9, #H9FF9, #H8001, #H0101, #H02G1,  
, #H8241, #H82C1, #H8341, #H8241, #H86  
41, #H8181, #H8001, #H7FFE, #H9141, #  
H9B41, #H9581, #H9181, #H8131, #H814  
9, #H8279, #H8249, #H8001, #H7FFE
```

• Définition du voyant enregistrement en mode "Stockage".

```
DATA 11, 25, #H7FC0, #HFFE0, #HFF1E0, #H  
EAE0, #HE4E0, #HEAE0, #HF160, #HFF60  
, #HFF60, #HC160, #HCF60, #HCF60, #HC  
360, #HCF60, #HCF60, #HC160, #HFF60,  
, #HFF60, #HF160, #HEAE0, #HE4E0, #HEA  
E0, #HF1E0, #HFF60, #H7FC0
```

• Pour le polygone du chrono.

```
DATA 46, 12, 150, 40, 258, 12, 152, 12, 25  
6, 14, 258, 38, 258, 40, 256, 40, 152, 38  
, 150, 14, 150, 12, 152
```

• Rectangles divers.

```
DATA #HDDFF, 14, 14, 14, 42, 165, 168, 80  
, #H17FF, 208, 195, 170, 10, 8, 9, 5, #HD  
FFF, 39, 39, 39, 68, 183, 182, 104, #H77  
FF, 257, 206, 175, 176, 128, 127, 131
```

• Code objet de la routine "Conversion".

```
DATA #H4E56, 0, #H302E, 8, #H41FA, #HB8  
, #H721F, #H43FA, #H32, #HB011, #H662  
0, #H2091, #H4269, 2, #H670C, #H4A29,  
3, #H670C, #H10BC, 3, #H6014, #H10BC,  
1, #H600E, #H10BC, 2, #H6008, #H5889,  
#H51C9, #HFFD0, #H51D0, #H4E5E, #H4E  
75, #H8819, #H4161, #H8919, #H4361, #  
H8D19, #H4R63, #H8F19, #H4265
```

```
DATA #H8F19, #H4165, #H9019, #H4365, #  
H9119, #H4865, #H9419, #H4369, #H951  
9, #H4866, #H9919, #H436F, #H9A19, #H  
486F, #H9D19, #H4175, #H9E19, #H4375  
, #H9F19, #H4875, #HA119, #H3000, #HA  
319, #H2300, #HA419, #H2700, #HA719,  
, #H7B00, #HB119, #H3100, #HCE19, #H6A  
00
```

```
DATA #HCF19, #H7A00, #HD619, #H3800, #  
H700, 0, #H913, #H4100, #H513, #H4200  
, #H813, #H4300, #H213, #H4400, #H313  
, #H4500, #H113, #H4600, #H413, #H470  
0, #H613, #H4800, #H13, #H4900
```

• Code objet de la routine "Clic".

```
DATA #H4E56, 0, #H246E, 8, #H3A2E, 12, #  
HCAPC, #H36, #H4EBA, #H144, #H4EBA, #  
H12E, #H70FF, #H51C8, #HFFFE, #H4EBA  
, #H124, #H4EBA, #H12C, #H4EFA, #HC6,  
, #H4E56, 0, #H246E, 8, #H487A, #H198, #  
HA972, #H7609, #H7800, #H7A00, #H426  
7, #H47FA, #H18A, #H2F13, #H4872, #H5  
002, #HA0AD, #H4A5F, #HCC00
```

```
DATA #HA2, #H645, #H36, #H5244, #H51CB  
, #HFFE4, #H43FA, #H166, #H47FA, #H16  
A, #H51EB, 5, #H22BC, 15, #H99, #H237C  
, #H26, #HA7, 4, #H4EBA, #H132, #H661A  
, #H43FA, #H146, #H337C, #HR2, 2, #H33  
7C, #HC0, 6, #H4EBA, #H11C, #H6758, #H  
50EB, 5, #H4EBA, #H124, #H50EB, 4, #H4  
267
```

```
DATA #HA973, #H4A5F, #H6700, #H26, #H4  
87A, #H122, #HA972, #H4EBA, #HFA, #H6  
708, #H4A2D, 4, #H66E4, #HG0DA, #H4A2  
B, 4, #H67DC, #H4EBA, #HF8, #H51EB, 4,  
, #H60D2, #H4A2B, 4, #H6602, #H6016, #H  
4A2B, 5, #H670A, #H36BC, 11, #H4EBA, #  
HDC, #H600A, #H36BC, 10, #H60F4, #H36  
BC, #HFFFE, #H4E5E
```

```
DATA #H4E75, #H4EBA, #H62, #H614C, #H5  
0EB, 4, #H4267, #HA973, #H4A5F, #H672  
E, #H487A, #HC6, #HA972, #H4267, #H43  
FA, #HBE, #H2F11, #H4872, #H5002, #HA  
8AD, #H4A5F, #H6708, #H4A2B, 4, #H66D  
A, #H60D2, #H4A2D, 4, #H67D2, #H6118,  
, #H51EB, 4, #H60CA, #H4A2B, 4, #H6604,  
, #H6116, #H60AF, #H6106, #H3684
```

```
DATA #H610E, #H60AA, #H2F0C, #HA8C9, #  
H487A, #H7A, #HA8BA, #H4E75, #H204C,  
, #HA023, #H4E75, #H7036, #HA122, #H43  
FA, #H68, #H22B2, #H5002, #H2351, 4, #  
H5369, 0, #H5369, 2, #H5869, 4, #H5869  
, 6, #H43F2, #H5000, #H2848, #H2050, #  
H701A, #H30D9, #H51C8, #HFFFC, #H205  
4, #H5268, 10, #H5268, 14
```

```
DATA #H5268, #H22, #H5268, #H24, #H526  
8, #H28, #H5268, #H2C, #H5268, #H2E, #  
H5268, #H32, #H4F75, #H4267, #H47FA,  
, #H1E, #H2F13, #H487A, 16, #HA8AD, #H4  
A5F, #H4E75, #H487A, 6, #HA8A4, #H4E7  
5
```

• Code objet de la routine "I/O".

```
DATA #H4E56, 0, #H226E, 0, #H5409, #H70  
00, #H1019, #HE188
```

```
DATA #H1019, #HR188, #H1011, #H41FA, #  
H1006A, #H2248, #H7218, #H4259
```

```
DATA #H51C9, #HFFFC, #H2140, 18, #HA00  
0, #H43FA, #H54, #H32A8
```

```
DATA 16, #H4E5E, #H4E75, #H4E56, 0, #H4  
1FA, #H46, #H42A8
```

```
DATA 12, #H226E, 8, #H1019, #HE148, #H1  
019, #H314U, #H26
```

```
DATA #H7000, #H1019, #HE188, #H1019, #  
HE188, #H1011, #H2140, 32
```

```
DATA #H317C, 1, #H2C, #H216E, 12, #H2E,  
, #HA002, #H60BA
```

```
DATA #H4E56, 0, #H41FA, 12, #H42A8, 12,  
, #HA001, #H60AA
```

• Code objet de la routine de fermeture des fenêtres (Pom's 19).

```
DATA #H42A7, #HA924, #H2E1F, #H6706, #  
H2F07, #HA916, #H60F2, #H4E75
```

• Code objet de la routine "MAJ".

```
DATA #H4E56, 0, #H76FF, #H266E, 8, #H38  
1D, #H643, 11, #H3F3C, 2, #H3F03, #HA8  
93, #H5344, #H6B10, #H7000, #H101B, #  
HC00, 13, #H67E6, #H3F00, #HARR3, #H6  
0EC, #H4E5E, #H4E75
```

• Code objet de la routine de chargement des icônes (Pom's 18).

```
DATA #H4E56, 0, #H42A7, #H3F2E, #HC, #H  
A9BD, #H2057, #H2250, #H246E, #H8, #H  
24FC, 32, 32, #H701F, #H24D9, #H51C8,  
, #HFFFC, #H4E5E, #H4E75
```

• Pour message si plus de 2000 caractères par pages (128Ko).

```
DATA .s.t.ent., .s." a", .ont", .s.
```

• Titres des menus.

```
DATA Mode, Stockage, Consultation, Co  
pit d'écian, Préparation/envoi, Fi  
chier, Nouveau..., Ouvrir..., Enregistr  
ement, Fermer, Quitter, Chronomètre  
, Départ automatique, Déconnexion  
automatique..., Marche, Mise à zéro  
DATA Liaison, Connexion, Déconnexion  
, LZU bauds, 4800 bauds, Pom's, (1)  
39.51.24.43
```

• Positions des caractères du chrono.

```
DATA 212, 220, 232, 240
```

• États des articles du menu "fichier" pour les différents modes.

```
DATA 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1  
, 1, 1, 1, 0, 1, 0, 0, 1, 1
```

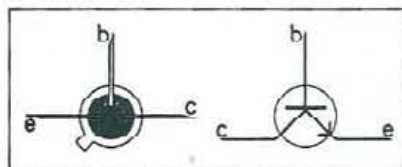
• Initialisation des tableaux de variables et mise en place des menus.

```
FOR I=0 TO 14: READ Z1(I): NEXT: FOR  
I=0 TO 10: READ Z(I): NEXT: A:=VARP  
TR(Z(0)): A: 4  
FOR I=0 TO 5: READ B5(I): NEXT: FOR I  
=0 TO 3: F(I)=#HAA55: H(I)=#HFFFE:  
NEXT: FOR I=0 TO 10 STEP 2: READ B  
(I), E(I+1), A$(I\2): NEXT: N=0: FOR  
I=0 TO 18 STEP 2: RESTORE S2: READ
```

```

D(N):N=N+1:FOR J=0 TO 12:READ D
(N).D(N+1)
D(N)-D(N)+E(I):D(N+1)-D(N+1)+E(I+1)
N=N+2:NEXT:NEXT:FOR I=0 TO 16:
READ A:J(I)=A:K(I)=A:NEXT:FOR I=
17 TO 26:READ J(I):NEXT:FOR I=17
TO 26:READ K(I):NEXT:FOR I=0 TO
26:READ L(I):NEXT:FOR I=0 TO 22
:READ M(I):NEXT
FOR I=0 TO 3:READ G(I),N(I),O(I),P
(I),Q(I),R(I),S(I),Y(I):NEXT:FOR
I=0 TO 96:READ A(I):NEXT:FOR I=
0 TO 225:READ B(I):NEXT:FOR I=0
TO 63:READ T(I):NEXT:FOR I=0 TO
7:READ C(I):NEXT:FOR I=0 TO 23:R
EAD U(I):NEXT:FOR I=0 TO 18:READ
W(I):NEXT
FOR I=0 TO 4:READ C$(I,0),C$(I,1):
NEXT:A1=0:A1=VARPTR(C(0))+A1+A16
-"3".A29=CHR$(27)+CHR$(57)+CHR$(
103):A69=CHR$(27)+CHR$(57)+CHR$(
104):A2=0:H9=CHR$(13)
FOR I=0 TO 4:READ C$:MENU 1,I,(I<>
1)+2,C$:NEXT:FOR I=0 TO 5:READ C
$:MENU 3,I,(I>1 AND I<5)+1,C$:NE
XT:FOR I=0 TO 4:READ C$:MENU 4,I
.1,C$:NEXT:FOR I=0 TO 4:READ C$:
MENU 5,I,(I>2)+1,C$:NEXT:FOR I=0
TO 1:READ C$:MENU 6,I,1,C$:NEXT
FOR I=0 TO 3:READ CR(2,I):NEXT:FOR
I=0 TO 4:FOR J=0 TO 4:READ Z(I
,J):NEXT:NEXT:CHAIN "Minitel/2",
,A11.

```



## Programme 'Minitel' Seconde partie (doit être baptisée 'Minitel/2')

• Adresses en cas d'interruptions par les événements.

```

DEFINT A-Z:GOSUB T141:GOSUB T140:G
OSUB S4:ON BREAK GOSUB S1:BREAK
ON:ON ERROR GOTO S0:ON DIALOG GO
SUB S5:ON MENU GOSUB S17:ON MOUS
E GOSUB S20:ON TIMER(1) GOSUB S2
Z:GOSUB S19:INITCURSOR

```

• Boucle principale.

```

10 ON A3 GOTO 11,10,13,10
GOTO 10
11 IF A4 THEN IF A5>=A6 THEN GOSUB
S18:GOSUB S55:GOSUB MA:GOSUB S5
3:GOSUB S19
C$=INKEY$:IF C$<>" THEN 20
12 IF A3=1 THEN A8=LOC(2):IF DE TH
EN IF A8 THEN PRINT#3,INPUT$(A8,
2):A9=A9+A8:IF C6 THEN IF E7 TH
EN GOSUB S18:E7=0:DT=0:GOSUB S67
:GOSUB S68:GOSUB MA:GOSUB S19
IF B1 THEN B1=0:GOSUB S48
IF DE=0 THEN IF A8 THEN CD$=INPUT$(
A8,2)
GOTO 10
13 IF B2 THEN IF D3 THEN A0=LOC(2)
:IF A8 THEN GOSUB S80

```

```

GOTO 10
14 GOSUB S63:GOTO 10
15 GOSUB 130:GOTO 10
20 N=ASC(C$):IF N=13 OR N=3 THEN N
=9:GOTO 30
IF N=8 THEN N=4:GOTO 30
IF N=9 THEN N=6:GOTO 30
IF N=27 THEN N=3:GOTO 30
IF N>31 AND N<128 THEN PRINT#1,CHK
$(N):ELSE GOSUB S94
GOTO 10
30 A1=VARPTR(B(0)):A1 N,VARPTR(D(0
)):GOSUB S94:GOTO 10

```

• On arrive ici en cas d'action sur un menu.

```

S17:GOSUB S18:M0=MENU(0):M1=MENU(1
):IF M0<6 THEN ON A3 GOSUB T7,T8
,S30,T9
MENU:GOSUB S19:RETURN

```

• Menu et mode "Stockage".

```

T7:ON M0 GOSUB S45,S1,S50,S54,S59:
RETURN

```

• Menu et mode "Consultation".

```

T8:ON M0 GOSUB S45,S1,T10,S54,S59:
RETURN

```

• Menu et mode "Préparation/Envoi".

```

T9:ON M0 GOSUB S45,S1,S35,S54,S59:
RETURN

```

• Article "Stockage" du menu "Mode".

```

S45:IF M1=A3 THEN RETURN
ON A3 GOSUB U1,U2,U1,U4:MENU 4,0,(
M1>1)+1:IF M1=2 THEN FOR I=1 TO
4:MENU 5,I,(I<3)+1:NEXT:MENU 5,3
,Z ELSE FOR I=1 TO 4:MENU 5,I,(I
>2)+1:NEXT
IF M1=4 THEN CB$="Enregistrer..." EL
SE CB$="Enregistrement"
MENU 3,3,1,CRS:A1=PRE(" "):ERASE A
1,I,V:ON M1 GOSUB S41,S42,S43,S4
4:RETURN
U2:IF DV THEN DV=0:V9="12":PRINT#1
,CHR$(27) "kd";
U4:CLOSE 1:RETURN
U1:CLOSE 1:CLOSE 2:RETURN

```

• Recodage des caractères si nécessaire.

```

S94:IF N<10 THEN IF LOC(2) THEN DE
EP:RETURN
IF N=10 OR N=11 THEN GOSUB S69:RET
URN
S88:A1=VARPTR(A(0)):A1 N:A1=A1
+194:A1=PREK(A1):A1=A1+1:IF A
1=0 THEN RETURN
FOR I=0 TO A1-1:PRINT#1,CHR$(PEEK(
A1+I)):NEXT:IF B4 THEN RETURN
IF N<10 THEN GOSUB S96
RETURN

```

• Affichage du contenu de la fenêtre en mode "Stockage". Attention : il y a deux 'espaces collantes' dans le PRINT "" "".

```

S65:IF DT THEN TIMER STOP
GOSUB S61:N=0:FOR I=0 TO 9:IF I=0
OR I=7 THEN FILLPOLY VARPTR(D(N)
),VARPTR(F(0)):GOTO S97
IF I=9 THEN FILLPOLY VARPTR(D(N)),
VARPTR(G(0)) ELSE FILLPOLY VARPTR
R(D(N)),VARPTR(H(0))
S97:FRAMEPOLY VARPTR(D(N)):IF I=0
OR I=7 OR I=9 THEN TEXTMODE 1:TE
XTFACE 0 ELSE TEXTMODE 3:TEXTFAC
E 0

```

• On arrive ici en cas d'action sur la souris, et dans la fenêtre.

```

S20:GOSUB S18:ON A3 GOSUB S21,S1,S
1,S1:GOSUB S19:RETURN
S21:TNITCURSOR:A1=VARPTR(R(21)):A1
VARPTR(D(0)):GOSUB S3:IF B(230)
>-1 THEN N=B(230):GOSUB S94
RETURN

```

• En mode "stockage", mise à jour de la table d'index après chaque action sur une commande Minitel.

```

S96:IF A9<100 THEN S98
T0:IF A9<T20 THEN S99
B5=B5+1:A!(B6+1)=A!(B6)+T20:I(B6)=
T20:BG-BG+1:IF B6-T22+1 THEN B1=
-1:ERROR 240:RETURN
A9=A9-T20:GOTO T0
S99:A!(DG+1)-A!(DG):A9:I(B6)-A9:B6
=B6+1:IF B6-T22+1 THEN B1=-1:ERR
OR 240:RETURN
A9=0:RETURN
S98:A!(B6)=A!(B6)+A9:IF B6 THEN I(
B6-1)=I(B6-1)+A9:A9=0
RETURN

```

• Ouverture d'un nouveau fichier en mode "stockage".

```

T1:DA=0:FS=FILES$(0):GOSUB S3
100 IF F9="" THEN DA=-1:RETURN
101 OPEN"o",3,F$:NAME F$ AS F$, "mt
1F":R6=0:A9=0:A1(0)=0:RETURN

```

• Ouverture d'un fichier en mode "Consultation".

```

S62:DA=0:FS=FILES$(1,"mt1F"):GOSUB
S3:IF FS="" THEN DA=-1:RETURN
OPEN"i",3,F$:RETURN

```

• On arrive ici en cas d'événements de type 'Dialog'.

```

S5:GOSUB S18:ON A3 GOSUB T2,T3,T4,
T5:GOSUB S19:RETURN

```

• 'Dialog' en mode "Stockage".

```

T2:B8=DIALOG(0):IF B8=5 THEN GOSUB
S97
RETURN

```

• 'Dialog' en mode "Consultation".

```

T3:B8=DIALOG(0):IF B8=5 THEN GOSUB
S93:RETURN
IF B8=1 THEN GOSUB S77:RETURN
IF B8=6 THEN IF B7<B9 THEN BUTTON
4,2:C1=4:GOSUB S70
RETURN

```

• 'Dialog' en mode "Copie d'écran".

```

T4:C2=DIALOG(0):IF C2=5 THEN GOSUB
S83
IF C2<>1 THEN RETURN
IF DIALOG(1)=1 THEN IF B3 THEN BUT
TON 1,1:B3=0 ELSE BUTTON 1,2:B3=
-1:A1=0:CB$=INPUT$(LOC(2),2)

```

```

MOVETO D(N+2)+9,D(N+1)+11:PRINT AS
(I):IF I=0 THEN TEXTFACE 8:MOVET
O 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 22
7,31:PRINT"" "":PUT(152,14),K,P
SET:PUT(177,14),J,DSET:IF A3=1 A
ND DE THEN PUT(195,14),L,PSET
IF DT THEN PAINTROUNDRECT VARPTR(P
(0)),4,4:TIMER ON
RETURN

```

• On arrive ici en cas d'action sur la souris, et dans la fenêtre.

```

S20:GOSUB S18:ON A3 GOSUB S21,S1,S
1,S1:GOSUB S19:RETURN
S21:TNITCURSOR:A1=VARPTR(R(21)):A1
VARPTR(D(0)):GOSUB S3:IF B(230)
>-1 THEN N=B(230):GOSUB S94
RETURN

```

• En mode "stockage", mise à jour de la table d'index après chaque action sur une commande Minitel.

```

S96:IF A9<100 THEN S98
T0:IF A9<T20 THEN S99
B5=B5+1:A!(B6+1)=A!(B6)+T20:I(B6)=
T20:BG-BG+1:IF B6-T22+1 THEN B1=
-1:ERROR 240:RETURN
A9=A9-T20:GOTO T0
S99:A!(DG+1)-A!(DG):A9:I(B6)-A9:B6
=B6+1:IF B6-T22+1 THEN B1=-1:ERR
OR 240:RETURN
A9=0:RETURN
S98:A!(B6)=A!(B6)+A9:IF B6 THEN I(
B6-1)=I(B6-1)+A9:A9=0
RETURN

```

• Ouverture d'un nouveau fichier en mode "stockage".

```

T1:DA=0:FS=FILES$(0):GOSUB S3
100 IF F9="" THEN DA=-1:RETURN
101 OPEN"o",3,F$:NAME F$ AS F$, "mt
1F":R6=0:A9=0:A1(0)=0:RETURN

```

• Ouverture d'un fichier en mode "Consultation".

```

S62:DA=0:FS=FILES$(1,"mt1F"):GOSUB
S3:IF FS="" THEN DA=-1:RETURN
OPEN"i",3,F$:RETURN

```

• On arrive ici en cas d'événements de type 'Dialog'.

```

S5:GOSUB S18:ON A3 GOSUB T2,T3,T4,
T5:GOSUB S19:RETURN

```

• 'Dialog' en mode "Stockage".

```

T2:B8=DIALOG(0):IF B8=5 THEN GOSUB
S97
RETURN

```

• 'Dialog' en mode "Consultation".

```

T3:B8=DIALOG(0):IF B8=5 THEN GOSUB
S93:RETURN
IF B8=1 THEN GOSUB S77:RETURN
IF B8=6 THEN IF B7<B9 THEN BUTTON
4,2:C1=4:GOSUB S70
RETURN

```

• 'Dialog' en mode "Copie d'écran".

```

T4:C2=DIALOG(0):IF C2=5 THEN GOSUB
S83
IF C2<>1 THEN RETURN
IF DIALOG(1)=1 THEN IF B3 THEN BUT
TON 1,1:B3=0 ELSE BUTTON 1,2:B3=
-1:A1=0:CB$=INPUT$(LOC(2),2)

```



```

RETURN
* "Dialog" en mode "Préparation/Envoi".
T5:D0-DIALOG(0):IF D0=5 THEN GOSUB
S84:RETURN
TF RA=1 THEN GOSUB S86
RETURN
* Mise à jour, mode "Stockage".
S92:GOSUB S65:C3=-1:IF A5 AND DT T
HEN GOSUB S22 ELSE GOSUB S68
C3=0:RETURN
* Menu "Fichier" en mode "Consultation".
T10:ON M1 GOSUB S46,T11,S27,S32,S2
8:RETURN
* Mode "Consultation", menu "Fichier" et
article "Ouvrir".
T11:F$=FILES$(1,"mt1F"):GOSUB S3
120 IF F$="" THEN RETURN
OPEN"R",3,F$+".T":FIELD 3,4 AS A3$
,2 AS A4$:GET 3,1
121 A1=CVS(A3$):IF A1<>123.4567 TH
EN ERROR 254
B9=CVI(A4$):IF B9 THEN B7=1 ELSE B
EEP:F$="",CLOSE 3:RETURN
FOR I=2 TO B9+2:GET 3,1:I(I-2)=CVI
(A4$):A1(I-2)=CVS(A3$):NEXT:CLOSE
3:GOSUB S24:K=4:GOSUB T98:GOSU
B S18
A5S=CHR$(LEN(F$))+F$:A1=VARPTR(T(0
)):A1:VARPTR(A5$):IF T(64) THEN
ERROR 255
PRINT#1,CHR$(12):GOSUB S33:RETURN
* Mode "Consultation", menu "Fichier" et
article "Fermer".
S32:FOR I=1 TO 6:BUTTON 1,0:NEXT:B
7=0:B9=0:GOSUB S34:A1=VARPTR(T(5
6)):A1:GOSUB S31:K=2:GOSUB T98:F
$="":RETURN
* Menu et mode "Copie d'écran".
S30:ON M0 GOSUB S45,S1,S29,S54,S59
:RETURN
* Menu "Fichier" en mode "Copie d'écran".
S29:ON M1 GOSUB S25,S1,S27,130,S28
:RETURN
* Mode "Copie d'écran", menu "Fichier" et
article "Ouvrir".
S25:GOSUB S79:IF DA THEN K=1:GOSUB
T98:MENU 1.3.2 ELSE V(0)=0:BU
TT ON 1,1:LINE(1,1) (242,269),30,DF
:GOSUB S24:EB=24:K=4:GOSUB T98:C
B$=INPUT$(LOC(2),2)
RETURN
* Mode "Copie d'écran", menu "Fichier" et
article "Fermer".
130 CLOSE 3:B2=0:BUTTON 1,0:B3=0:G
OSUB S31:K=1:GOSUB T98:F$="":RET
URN
* Menu "Fichier" en mode
"Préparation/Envoi".
S35:ON M1 GOSUB S36,S37,S38,S1,S28
:RETURN
* Mode "Préparation/Envoi", menu
"Fichier" et article "Nouveau".
S36:F$="":C4=0:GOSUB SPE:MOVETO 6,
16:PRINT" Fichier : Sans titre":C
B$="":GOSUB T30:RETURN
* Mode "Préparation/Envoi", menu
"Fichier" et article "Ouvrir".

```

```

S37:INITCURSOR:CRS=F$:FS=FILES$(1,
"TEXT"):GOSUB S3. IF F$="" THEN F
$=CB$:RETURN
OPEN"R",3,F$:L=LOF(3):IF L>960 THE
N BEEP:L=960
CB$=INPUT$(L,3):CLOSE 3:C4=0:GOSUB
SPE:GOSUB T30:RETURN
* Mode "Préparation/Envoi", menu
"Fichier" et article "Enregistrer".
S38:CB$=EDIT$(1):L=LEN(CB$):IF L=0
THEN BEEP:RETURN
IF L>960 THEN L=960:BEEP:C4=L:G$="
960":CRS=LEFT$(CRS,L):GOSUB T30
IF F$<>" THEN S39
INITCURSOR:F$=FILES$(0)
160 IF F$="" THEN RETURN
GOSUB S3
161 OPEN"O",3,F$:NAME F$ AS F$, "TE
XT":GOTO 162
S39:OPEN"O",3,F$
162 PRINT#3,CB$:
163 CLOSE 3
164 RETURN
* Menu "Fichier" en mode "Stockage".
S50:ON M1 GOSUB S46,S47,S27,S48,S2
8:RETURN
* Menu "Chronomètre" en mode
"Stockage".
S54:ON M1 GOSUB S51,S52,MA,S53:RET
URN
* Menu "Liaison" en mode "Stockage".
S59:ON M1 GOSUB S56,S55,S57,S58:RE
TURN
* Menu "Mode", article "Stockage".
S41:GOSUB T141:GOSUB T140:DIM A!(T
22),I(T22),V(2):GOSUB S60:C5=-1:
GOSUB S4:C5=0:K=1:GOSUB T98:RETU
RN
* Menu "Mode", article "Consultation".
S42:GOSUB T140:DIM A!(T22),I(T22),
V(2):PRINT#1,A2$,:GOSUB S60:WIND
OW 1,"", (20,50)-(156,240),4:GOSU
R S61:FOR I=0 TO 5:BUTTON I+1,0,
B$(I), (8,8+25*I)-(128,26+25*I).N
EXT:B7=0:B9=0:K=2:GOSUB T98:GOSU
B T11:RETURN
* Menu "Mode", article "Copie d'écran".
S43:GOSUB T141:GOSUB T140:DIM A!(2
),I(2),V(492):GOSUB S60:WINDOW 1
,"", (20,32)-(264,324),4:WIDTH 41
:TEXTFONT 4:TEXTMODE 1:TEXTSIZE
9:K=1:GOSUB T98:BUTTON 1,0,"Réce
ption de copie active...", (2,274)
-(242,289),2:GOSUB S25:V(U)=U:KE
TURN
* Menu "Mode", article
"Préparation/Envoi".
S44:GOSUB T140:DIM A!(2),I(2),V(2)
:GOSUB S60:C4=0:WINDOW 1,"", (70,
50)-(411,324),4:GOSUB S61:WIDTH
80:K=3:GOSUB T98:BUTTON 1,1,"Nom
bre de caractères", (2,23)-(163,4
2):BUTTON 2,1,"Envoyer", (228,23)-(
389,42):CB$="":GOSUB T30:RETURN
* Mode "Stockage", menu "Fichier" et
article "Nouveau".
S46:GOSUB T1:IF DA=0 THEN B5=0:GOS
UB S24:K=0:GOSUB T98:E7=-1
RETURN

```

```

* Mode "Stockage", menu "Fichier" et
article "Ouvrir".
S47:GOSUB S62:IF DA=0 THEN K=4:GOS
UB T98
RETURN
* Mode "Stockage", menu "Fichier" et
article "Enregistrement".
S27:IF DE THEN DE=0:ERASERECT VARP
TR(0(0)) ELSE DE=-1:PUT(195,14),
L,PSET
MENU 3,3,ABS(DE)+1:RETURN
* Mode "Stockage", menu "Fichier" et
article "Fermer".
S40:IF D5 THEN ERROR 241 ELSE GOSU
B S96
110 CLOSE 3:F2S=F$+".T"
111 OPEN"R",3,F2$:NAME F2$ AS F2$,
"mt1T":FIELD 3,4 AS A3$,2 AS A4$
:T.SRT A4$=MKI$(R6):LSET A3$=MKS$
(123.4567)
112 PUT 3,1:FOR I=0 TO B6:LSET A3$
-MKS$(A1(I)):LSET A4$=MKI$(T(I))
:PUT 3,I+2:NEXT:CLOSE 3
S63:GOSUB S31:ERASERECT VARPTR(0(0
)):DE=0:F$="":K=1:GOSUB T98:RETU
RN
* Menu "Fichier", article "Quitter".
S28:PRINT#1,A2$,:CLOSE:A1=VARPTR(T
(56)),A1:A1=VARPTR(C(0)):A1:SYST
EM
* Mode "Stockage", menu "Chronomètre"
et article "Départ automatique".
S51:IF C6 THEN MENU 4.1.1:C6=0:E7=
1 ELSE MENU 4,1,2:C6=-1
RETURN
* Mode "Stockage", menu "Chronomètre"
et article "Déconnexion automatique".
S52:IF A4 THEN MENU 4,2,1,CD$+"...":
A4=0:RETURN
IF DT THEN TIMER OFF
A4=-1:WINDOW 2,"", (32,62)-(222,134
),-2:GOSUB S64:EDIT FIELD 1,A1$,
(83,21)-(120,35),1,3:BUTTON 1,1,
"OK", (14,46)-(172,64):ON DIALOG
GOSUB S6:DIALOG ON:INITCURSOR
S9:COTO S9
S8:GOSUB S3:GOSUB S65:MENU 4,2,2,C
DS+ "/"+A1S+"" :ON DIALOG GOSUB
S9:IF DT THEN TIMER ON
RETURN
* Mode "Stockage", menu "Chronomètre"
et article "Mise à zéro".
S53:GOSUB S67:GOSUB S68:RETURN
* Menu "Liaison" et article "Connexion".
S56:PRINT#1,A6$,:RETURN
* Mode "Consultation", menu "Liaison" et
article "1200 bauds".
S57:IF DV THEN DV=0:V$="-12":PRINT#
1,CHR$(27) " :kd":GOTO T99
* Mode "Consultation", menu "Liaison" et
article "4800 bauds".
S58:IF DV=0 THEN DV=-1:V$="48":PRI
NT#1,CHR$(27) " :kv":
T99:CLOSE 1:GOSUB T140:MENU 5,3,DV
+2:MENU 5,4,ABS(DV)+1:RETURN
* Affichage et mise à jour du chronomètre.
S22:A5=TIMER-B1-C7:IF A5=3600 THEN
B1=TIMER:C7=0:A5=0

```

```

S68:C8=A5\60:D4=A5 MOD 60:CR(0,0)=
C8\10:CR(0,1)=C8 MOD 10:CR(0,2)=
D4\10:CR(0,3)=D4 MOD 10:TEXTFACE
0

```

```

FOR K=0 TO 3:IF CR(0,K) <> CR(1,K) O
R C3 THEN CR(1,K)=CR(0,K):LINE(
CR(2,K),22)-(CR(2,K)+6,30),30,BF
:MOVETO CR(2,K)-9,31:PRINT CR(1,
K);
NEXT:RETURN

```

*Mise à zéro du chronomètre.*

```

S67:IF DT THEN D=TIMER
C!=0:C7=0:A5=0:FOR K=0 TO 3:CR(1,K)
)=-1:NEXT:RETURN

```

*Marche/Arrêt chronomètre.*

```

MA:IF DT THEN TIMER OFF:MENU 4,3,1
,"Marche":DT=0:C!=TIMER:ERASEREC
T VARPTR(P(0)):RETURN
MENU 4,3,1,"Arrêt":DT=-1:IF A5 THE
N C7=C7+(TIMER-C!) ELSE C7=0:B!=
TIMER
FAINTKOUNDRECT VARPTR(P(0)),4,4:TI
MER ON:RETURN

```

*Action sur les boutons "Marche/Arrêt" ou "Mise à zéro" du chronomètre.*

```

S69:IF N=10 THEN GOSUB MA:RETURN
GOSUB S67:GOSUB S68:RETURN

```

*Fenêtre pour choix du temps si déconnexion automatique.*

```

S64:GOSUB S61:TEXTFACE 0:MOVETO 8,
16:PRINT CDS:MOVETO 9,33:PRINT"à
u bout de":MOVETO 126,33:PRINT"m
inutes":PENSIZE 3,3:FRAMEROUNDRE
CT VARPTR(Q(0)),16,16,PENNORMAL:
RETURN

```

*Si événements dans la fenêtre pour choix du temps.*

```

S6: C2=DIALOG(0):IF C2<>1 AND C2<>6
THEN RETURN
EDC=EDIT$(1):IF LEN(EDC)>2 THEN S7
D9=VAL(EDC):IF D9<1 OR D9>59 THEN
S7
A1$=STR$(D9):A1$=RIGHT$(A1$,LEN(A1
$)-1):A6=D9*60-1:WINDOW CLOSE 2:
RETURN S8
S7:BEEP:EDIT FIELD 1,A1$, (83,21)-(
120,35).1.3:RETURN

```

*Déconnexion.*

```

S55:PRINT#1,A2$;:RETURN

```

*Mets 'off' les articles du menu mode, sauf le mode courant qui est coché.*

```

S24:FOR I=1 TO 4:MENU 1,I,2*ABS(I-
A3):NEXT:RETURN

```

*Mets 'on' les articles du menu mode, sauf le mode courant qui est coché.*

```

S31:FOR I=1 TO 4:MENU 1,I,(A3<>I)+
2:NEXT:RETURN

```

*Contenu de la fenêtre en mode "Consultation".*

```

S93:PENSIZE 2,2:FRAMEROUNDRECT VAK
PTR(V(0)),16,16:PENNORMAL:PENPAT
VARPTR(F(0)):FRAMERECT VARPTR(R
(0)):PENNORMAL
S34:ERASERECT VARPTR(S(0)):CBS=STR
$(B1)+" /"+STR$(B9)+" page":IF B
9>1 THEN CBS=CBS+"s"
MOVETO 0:(118-WIDTH(CBS))/2,178:DR
INT CBS;:RETURN

```

*Mise à jour du compteur de pages après action sur un bouton en mode "consultation".*

```

S77:C1=DIALOG(1)
S70:E1=-1:ON C1 GOSUB S71,S72,S73,
S74,S75,S76:GOSUB S78:GOSUB S34:
GOSUB S33:RETURN
S71:B7=1:RETURN
S72:B7-B7+10:RETURN
S73:B7=B7-1:RETURN
S74:B7=B7+1:E1=0:RETURN
S75:D7-B7+10:RETURN
S76:B7=B9:RETURN

```

*État des boutons après consultation d'une page.*

```

S78:BUTTON 1,(B7=1)+1:BUTTON 2,(B7
<11)+1:BUTTON 3,(B7=1)+1:BUTTON
4,(B7-B9)+1:BUTTON 5,((B7+10)>B9
)+1:BUTTON 6,(B7=B9)+1:RETURN

```

*Lecture et envoi d'une page.*

```

S33:MENU OFF:CB$="":A!=FRE(" "):CB
$=SPACE$(I(B7-1)):A1=VARPTR(T(27
)):A! A!(B7-1),VARPTR(CB$):MENU
ON:IF T(64) THEN ERROR 254
IF B1 THEN PRINT#1,CHR$(12);
PRINT#1,CB$;:RETURN

```

*Autorise les événements.*

```

S19:DIALOG ON:MENU ON:MOUSE ON:INI
TCURSOK:RETURN

```

*Interdit les événements.*

```

S18:DIALOG STOP:MENU STOP:MOUSE OF
F:GOSUB S3:RETURN

```

*Fenêtre du mode "Stockage"*

```

S4:A3-1:WINDOW 1,"", (20,50)-(428,1
86),4:GOSUB S61:WIDTH 80:IF C5=0
THEN GOSUB S65
GOSUB S67:GOSUB S68:RETURN

```

*Ouverture d'un nouveau fichier en mode "Copie d'écran".*

```

S79:DA=0:F$=FILES$(0):GOSUB S3
140 IF F$="" THEN DA=-1:BUTTON 1,0
:RETURN
141 OPEN"o",3,F$:NAME F$ AS F$, "TE
XT":BUTTON 1,1:B2=-1:RETURN

```

*Traitement en mode "Copie d'écran".*

```

S80:DIALOG STOP:IF E5=24 THEN LINE
(1,1) (243,269),30,BF:E5=0:MOVET
O 2,10:BUTTON 1,0
E5=INPUT$(1,2):IF E5 THEN 150
IF E3=CHR$(19) THEN CBS=INPUT$(11,
2):DIALOG ON:RETURN
150 IF E5=B$ THEN E5=E5+1:IF E5=24
THEN I90 ELSE I89
IF ASC(E5)>31 THEN S82
DIALOG ON:RETURN
S82:IF E5="{ THEN E5="è"
IF E5="}" THEN E5="à"
IF E5="è" THEN E5="à"
IF E5="\ THEN E5="ç"
IF E5="'" THEN E5="\"
IF E5="[" THEN E5="]"

```

```

T89:PRINT E5:GETPEN VARPTR(PL(0))
:IF PL(0)-263 AND PL(1)>237 THEN
T88
PRINT #3,E5:A1=A1+1:V(0)=A1-1:A!=
VARPTR(V(1)):POKE A1+A1-1,ASC(E5
)
IF A1<984 THEN DIALOG ON:RETURN
188:E5=24

```

```

T90:PRINT #3,H$ H$ "-----
-----" H$ H
$:CBS=INPUT$(LOC(2),2):BUTTON 1
.1:B3=0:DIALOG ON:RETURN ELSE GO
TO S82

```

*Mise à jour de la fenêtre en mode "Copie d'écran".*

```

S83:PENPAT VARPTR(F(0)):MOVETO 2,2
70:LINETO 241,270:PENNORMAL:IF V
(0) THEN V(0)=V(0)+1:A!=VARPTR(U
(0)):A! VARPTR(V(0)):V(0)=V(0)-1
RETURN

```

*Mise à jour de la fenêtre en mode "Préparation/Envoi".*

```

S84:LINE(2,2)-(388,20),,B:LINE(2,3
)-(388,19),,B:MOVETO 6,16:PRINT"
Fichier : ":IF F$="" THEN PRINT
"Sans titre":GOTO T12
I=INSTR(F$,":"):I5=RIGHT$(F$,LEN(F
$)-I)
IF WIDTH(I5)<320 THEN PRINT I5;:GO
TO T12
T13:I5=LEFT$(I5,LEN(I5)-1):IF WIDT
H(I5+"...")>320 THEN I13 ELSE PRIN
T I5+"...";
T12:LINE(165,23)-(225,41),,B:LINE(
165,24)-(225,40),,B:GOSUB S85:LI
NE(2,44)-(388,271),,B:RETURN

```

*Action sur un bouton en mode "Préparation/Envoi".*

```

S86:CB$=EDIT$(1):C4=LEN(CB$):G$=ST
R$(C4):G$=RIGHT$(G$,LEN(G$)-1):D
6=DIALOG(1):IF E6=1 THEN GOSUB S
85:RETURN
IF E6<>2 THEN RETURN
FOR J=1 TO C4:N=ASC(MID$(CB$,J,1))
:IF N>31 THEN IF N<128 THEN DRIN
T#1,CHR$(N);:GOTO S87
IF N=13 THEN N=6
B4=-1:GOSUB S88:B4=0:IF N=6 THEN A
!=VARPTR(Z1(0)):A! 144
S87:A!=VARPTR(Z1(0)):A! 6:NEXT:RET
URN

```

*Affichage du nombre de caractère en mode "Préparation/Envoi".*

```

S85:LINE(166,25)-(224,39),30,BF:IF
C4 THEN MOVETO 166+(58-WIDTH(G$
))\2,37:PRINT G$
RETURN

```

*Initialisations pour chaque changement de mode.*

```

S60:F$="":CLS:IF DT THEN DT=0:TIME
R OFF
MENU 1,A3,1:A3=M1:MENU 1,A3,2:RETU
RN

```

*Sous-programme de traitement des erreurs.*

```

S0:IF ERR=53 THEN RESUME NEXT
IF ERR=254 THEN IF T(64)=-49 THEN
RESUME NEXT
WINDOW 2,"", (32,62)-(400,110),-2:W
IDTH 80:GOSUB S61:TEXTFACE 0:IF
ERR=240 OR ERR=241 THEN S89
IC=-6047:GOSUB S16:PUT(8,8),X(0):I
C=0:GOSUB S16:DIFF(48,8),X(0):MOV
ETO 216,40:PRINT CC$;:MOVETO 96,
17:IF ERR=254 THEN PRINT"Fichier
d'index introuvable.":GOTO S11
IF ERR=57 THEN PRINT"Erreur d'entr

```

```

ees/sorties.":GOTO S11
IF ERR=61 THEN PRINT"Disquette sat
urée.";COTO S11
IF ERR=70 THEN PRINT"Disquette pro
tégée.":GOTO S11
PRINT"Erreur système.";
S11:ON A3 GOTO S12,S13,S14,S15
S12:PRINT"Enregistrement":MOVETO
96,33:PRINT"Impossible.";COSUB
S10:IF ERL=12 THEN CLOSE 3:KILL
F$:RESUME 14
IF ERL=101 THEN F$:RESUME 100
IF ERL=110 OR ERL=111 THEN KILL F$
RESUME S63
IF ERL=112 THEN KILL F$:CLOSE 3:KI
LL F$:RESUME S63
RESUME NEXT
S13:GOSUB S10:IF ERL=121 THEN CLOS
E 3:KILL F$+".T":F$="":RESUME 12
0
RESUME NEXT
S14:GOSUB S10:IF ERL=130 THEN RESU
ME NEXT
IF ERL=141 THEN F$="":RESUME 140
IF ERL=150 THEN RESUME 15
RESUME NEXT
S15:GOSUB S10:IF ERL=161 THEN F$="
":RESUME 160
IF ERL=162 THEN CLOSE 3:RESUME 164
IF ERL=163 THEN RESUME 164
RESUME NEXT
S89:IC=1:GOSUB S16:PUT(8,8),X(0):M
OVETO 216,45:PRINT CCS:IF ERR=2
41 THEN S90
MOVETO 48,17:PRINT"Maximum 64 page
s, le fichier va":MOVETO 48,33:PR
INT"être clos.":GOSUB S10:RESUM
E NEXT
S90:MOVETO 44,10:CB$=STR$(B5)+" pa
ge"+C$(0,(B5=1)+1)+" contena1"+C
$(1,(B5=1)+1)+" plus de 2000 car
actères.":PRINT CB$:MOVETO 48,26
:CB$="elle"+C$(2,(B5=1)+1)+C$(3,
(B5=1)+1)+" n0 être scindée"+C$(
4,(B5=1)+1):PRINT CB$:COSUB S10:
RESUME 110
S16:A:=VARPTR(W(0)):A! IC.VARPTR(X
(0)):RETURN
S10:WHILE MOUSE(0):WEND:WHILE MOUS
E(0)=0:WEND:WINDOW CLOSE 2:IF ER
R=241 THEN RETURN
GOSUB S91:RETURN
S91:ON A3 COSUB S92,S93,S93,S94:RE
TURN
S1:RETURN

```

\* Curseur en forme de montre.

S3:A:=VARPTR(Z(U)):A! 4:RETURN

\* Chicago 12 points.

S61:TEXTFONT 0:TEXTMODE 1:TEXTSIZE

12:RETURN

\* Ouverture de l'interface COM1 en entrée.

T141:OPEN "COM1:"+V\$+"00,E,7,1" FO

R INPUT AS 2 LEN=1000:RETURN

\* Ouverture de l'interface COM1 en sortie.

T140:OPEN "COM1:"+V\$+"00,E,7,1" FO

R OUTPUT AS 1:RETURN

\* Zone d'édition pour le mode

"Préparation/Envoi".

T30:EDIT FIELD 1,CRS,(4,46)-(387,2

69):RETURN

```

SPE:LINE(3,4)-(387,18),30,BF:LINE(
166,25)-(224,39),30,BF:RETURN
T98:FOR J=0 TO 4:MENU 3,J+1,22(K,J
):NEXT:RETURN

```

## Source 'Clic.Asm'

```

;*****
; Routine utilisée en cas de 'clic'
; dans la fenêtre en mode "stockage"
; La routine détecte une action sur
; des boutons et opère en conséquence.

```

```

Trap _GetMouse $A972
Trap _PtInRect $A8AD
Trap _NewHandle $A1Z2
Trap _InvertPoly $A8C9
Trap _StillDown $A973
Trap _InvertOval $A8BA
Trap _DisposeHandle $A023
Trap _InvertRect $A9AA

```

```

LINK A6,#0
MOVEA.L 8(A6),A2
MOVE 12(A6),D5
MULU $54,D5
RSR S4
BSR S2
MOVEQ #-1,D0
DBRA D0,01
BSR S2
BSR S3
BRA Fin
LINK A6,#0
MOVEA.L 8(A6),A2
PEA E5

```

```

_GetMouse
MOVEQ $0,D3
MOVEQ $0,D4
MOVEQ $0,D5
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA 2(A2,D5)
PtInRect
TST (A7)+
DND 02
ADDI $54,D5
ADDQ $1,D4
DBRA D3,02
LEA E4,A1
LEA E5,A3
SF 5(A3)
MOVE.L $5F0099,(A1)
MOVE.L $2600A7,4(A1)
BSR 05
BNE.S 03
LEA E4,A1
MOVE $178,2(A1)
MOVE $192,6(A1)
RSR S5
BEQ.S 010
ST 5(A3)
BSR 06
ST 4(A3)
CLR -(A7)

```

```

StillDown
TST (A7)+
REQ 06
PEA E5
_GetMouse
BSR S5
BEQ.S 05
TST.B 4(A3)
BNE.S 04
BRA.S 03
TST.B 4(A3)
BEQ.S 04
BSR 06
SF 4(A3)
BRA.S 04
TST.B 4(A3)
BNE.S 07
BRA.S 010
TST.B 5(A3)

```

```

_GetMouse
MOVEQ $0,D3
MOVEQ $0,D4
MOVEQ $0,D5
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA 2(A2,D5)
PtInRect
TST (A7)+
DND 02
ADDI $54,D5
ADDQ $1,D4
DBRA D3,02
LEA E4,A1
LEA E5,A3
SF 5(A3)
MOVE.L $5F0099,(A1)
MOVE.L $2600A7,4(A1)
BSR 05
BNE.S 03
LEA E4,A1
MOVE $178,2(A1)
MOVE $192,6(A1)
RSR S5
BEQ.S 010
ST 5(A3)
BSR 06
ST 4(A3)
CLR -(A7)

```

```

StillDown
TST (A7)+
REQ 06
PEA E5
_GetMouse
BSR S5
BEQ.S 05
TST.B 4(A3)
BNE.S 04
BRA.S 03
TST.B 4(A3)
BEQ.S 04
BSR 06
SF 4(A3)
BRA.S 04
TST.B 4(A3)
BNE.S 07
BRA.S 010
TST.B 5(A3)

```

```

_GetMouse
MOVEQ $0,D3
MOVEQ $0,D4
MOVEQ $0,D5
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA 2(A2,D5)
PtInRect
TST (A7)+
DND 02
ADDI $54,D5
ADDQ $1,D4
DBRA D3,02
LEA E4,A1
LEA E5,A3
SF 5(A3)
MOVE.L $5F0099,(A1)
MOVE.L $2600A7,4(A1)
BSR 05
BNE.S 03
LEA E4,A1
MOVE $178,2(A1)
MOVE $192,6(A1)
RSR S5
BEQ.S 010
ST 5(A3)
BSR 06
ST 4(A3)
CLR -(A7)

```

```

StillDown
TST (A7)+
REQ 06
PEA E5
_GetMouse
BSR S5
BEQ.S 05
TST.B 4(A3)
BNE.S 04
BRA.S 03
TST.B 4(A3)
BEQ.S 04
BSR 06
SF 4(A3)
BRA.S 04
TST.B 4(A3)
BNE.S 07
BRA.S 010
TST.B 5(A3)

```

```

_GetMouse
MOVEQ $0,D3
MOVEQ $0,D4
MOVEQ $0,D5
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA 2(A2,D5)
PtInRect
TST (A7)+
DND 02
ADDI $54,D5
ADDQ $1,D4
DBRA D3,02
LEA E4,A1
LEA E5,A3
SF 5(A3)
MOVE.L $5F0099,(A1)
MOVE.L $2600A7,4(A1)
BSR 05
BNE.S 03
LEA E4,A1
MOVE $178,2(A1)
MOVE $192,6(A1)
RSR S5
BEQ.S 010
ST 5(A3)
BSR 06
ST 4(A3)
CLR -(A7)

```

```

StillDown
TST (A7)+
REQ 06
PEA E5
_GetMouse
BSR S5
BEQ.S 05
TST.B 4(A3)
BNE.S 04
BRA.S 03
TST.B 4(A3)
BEQ.S 04
BSR 06
SF 4(A3)
BRA.S 04
TST.B 4(A3)
BNE.S 07
BRA.S 010
TST.B 5(A3)

```

```

DEQ.S 02
MOVE $11,(A3)
BSR S6
BRA.S Fin
MOVE $10,(A3)
BRA.S 08

```

```

MOVE $0FFFF,(A3)
UNLK A6
RTS
BSR S4
BSR.S S2
ST 4(A3)
CLR -(A7)

```

```

StillDown
TST (A7)+
BEQ.S 05
PEA E5
_GetMouse
CLR -(A7)
LEA E5,A1
MOVE.L (A1),-(SP)
PEA 2(A2,D5,W)
PtInRect
TST (A7)+
BEQ.S 04
TST.B 4(A3)
BNE.S 03
BRA.S 02
TST.B 4(A3)
BEQ.S 03
DBR.S 02
SF 4(A3)
BRA.S 03
TST.B 4(A3)
BNE.S 06
BSR.S S3
BRA.S Nnn
BSR.S S2
MOVE D4,(A3)
BSR.S 03
BRA.S Fin
MOVE.L A4,-(SP)

```

```

InvertPoly
PEA E4
InvertOval
RTS
MOVEA.L A4,A0
DisposeHandle
KLS
MOVEQ $54,D0
NewHandle
LEA E4,A1
MOVE.L 2(A2,D5,W),(A1)
MOVE.L (A1),4(A1)
SUBQ $1,0(A1)
GUDQ $1,2(A1)
ADDQ $4,4(A1)
ADDQ $4,6(A1)
LEA 0(A2,D5,W),A1
MOVEA.L A0,A4
MOVEA.L (A0),A0
MOVEQ $26,D0
MOVE (A1)+,(A0)+
DBRA D0,01
MOVEA.L (A4),A0
ADDQ $1,10(A0)
ADDQ $1,14(A0)
ADDQ $1,34(A0)
ADDQ $1,36(A0)
ADDQ $1,40(A0)
ADDQ $1,44(A0)
ADDQ $1,46(A0)
ADDQ $1,50(A0)
KLS
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA E4
PtInRect
TST (A7)+
RTS
RTS
PEA E4
InvertRect
RTS
DC 0,0,0,0
DC 0,0,0

```

```

_GetMouse
CLR -(A7)
LEA E5,A1
MOVE.L (A1),-(SP)
PEA 2(A2,D5,W)
PtInRect
TST (A7)+
BEQ.S 04
TST.B 4(A3)
BNE.S 03
BRA.S 02
TST.B 4(A3)
BEQ.S 03
DBR.S 02
SF 4(A3)
BRA.S 03
TST.B 4(A3)
BNE.S 06
BSR.S S3
BRA.S Nnn
BSR.S S2
MOVE D4,(A3)
BSR.S 03
BRA.S Fin
MOVE.L A4,-(SP)

```

```

InvertPoly
PEA E4
InvertOval
RTS
MOVEA.L A4,A0
DisposeHandle
KLS
MOVEQ $54,D0
NewHandle
LEA E4,A1
MOVE.L 2(A2,D5,W),(A1)
MOVE.L (A1),4(A1)
SUBQ $1,0(A1)
GUDQ $1,2(A1)
ADDQ $4,4(A1)
ADDQ $4,6(A1)
LEA 0(A2,D5,W),A1
MOVEA.L A0,A4
MOVEA.L (A0),A0
MOVEQ $26,D0
MOVE (A1)+,(A0)+
DBRA D0,01
MOVEA.L (A4),A0
ADDQ $1,10(A0)
ADDQ $1,14(A0)
ADDQ $1,34(A0)
ADDQ $1,36(A0)
ADDQ $1,40(A0)
ADDQ $1,44(A0)
ADDQ $1,46(A0)
ADDQ $1,50(A0)
KLS
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA E4
PtInRect
TST (A7)+
RTS
RTS
PEA E4
InvertRect
RTS
DC 0,0,0,0
DC 0,0,0

```

```

_GetMouse
CLR -(A7)
LEA E5,A1
MOVE.L (A1),-(SP)
PEA 2(A2,D5,W)
PtInRect
TST (A7)+
BEQ.S 04
TST.B 4(A3)
BNE.S 03
BRA.S 02
TST.B 4(A3)
BEQ.S 03
DBR.S 02
SF 4(A3)
BRA.S 03
TST.B 4(A3)
BNE.S 06
BSR.S S3
BRA.S Nnn
BSR.S S2
MOVE D4,(A3)
BSR.S 03
BRA.S Fin
MOVE.L A4,-(SP)

```

```

InvertPoly
PEA E4
InvertOval
RTS
MOVEA.L A4,A0
DisposeHandle
KLS
MOVEQ $54,D0
NewHandle
LEA E4,A1
MOVE.L 2(A2,D5,W),(A1)
MOVE.L (A1),4(A1)
SUBQ $1,0(A1)
GUDQ $1,2(A1)
ADDQ $4,4(A1)
ADDQ $4,6(A1)
LEA 0(A2,D5,W),A1
MOVEA.L A0,A4
MOVEA.L (A0),A0
MOVEQ $26,D0
MOVE (A1)+,(A0)+
DBRA D0,01
MOVEA.L (A4),A0
ADDQ $1,10(A0)
ADDQ $1,14(A0)
ADDQ $1,34(A0)
ADDQ $1,36(A0)
ADDQ $1,40(A0)
ADDQ $1,44(A0)
ADDQ $1,46(A0)
ADDQ $1,50(A0)
KLS
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA E4
PtInRect
TST (A7)+
RTS
RTS
PEA E4
InvertRect
RTS
DC 0,0,0,0
DC 0,0,0

```

```

_GetMouse
CLR -(A7)
LEA E5,A1
MOVE.L (A1),-(SP)
PEA 2(A2,D5,W)
PtInRect
TST (A7)+
BEQ.S 04
TST.B 4(A3)
BNE.S 03
BRA.S 02
TST.B 4(A3)
BEQ.S 03
DBR.S 02
SF 4(A3)
BRA.S 03
TST.B 4(A3)
BNE.S 06
BSR.S S3
BRA.S Nnn
BSR.S S2
MOVE D4,(A3)
BSR.S 03
BRA.S Fin
MOVE.L A4,-(SP)

```

```

InvertPoly
PEA E4
InvertOval
RTS
MOVEA.L A4,A0
DisposeHandle
KLS
MOVEQ $54,D0
NewHandle
LEA E4,A1
MOVE.L 2(A2,D5,W),(A1)
MOVE.L (A1),4(A1)
SUBQ $1,0(A1)
GUDQ $1,2(A1)
ADDQ $4,4(A1)
ADDQ $4,6(A1)
LEA 0(A2,D5,W),A1
MOVEA.L A0,A4
MOVEA.L (A0),A0
MOVEQ $26,D0
MOVE (A1)+,(A0)+
DBRA D0,01
MOVEA.L (A4),A0
ADDQ $1,10(A0)
ADDQ $1,14(A0)
ADDQ $1,34(A0)
ADDQ $1,36(A0)
ADDQ $1,40(A0)
ADDQ $1,44(A0)
ADDQ $1,46(A0)
ADDQ $1,50(A0)
KLS
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA E4
PtInRect
TST (A7)+
RTS
RTS
PEA E4
InvertRect
RTS
DC 0,0,0,0
DC 0,0,0

```

```

_GetMouse
CLR -(A7)
LEA E5,A1
MOVE.L (A1),-(SP)
PEA 2(A2,D5,W)
PtInRect
TST (A7)+
BEQ.S 04
TST.B 4(A3)
BNE.S 03
BRA.S 02
TST.B 4(A3)
BEQ.S 03
DBR.S 02
SF 4(A3)
BRA.S 03
TST.B 4(A3)
BNE.S 06
BSR.S S3
BRA.S Nnn
BSR.S S2
MOVE D4,(A3)
BSR.S 03
BRA.S Fin
MOVE.L A4,-(SP)

```

```

InvertPoly
PEA E4
InvertOval
RTS
MOVEA.L A4,A0
DisposeHandle
KLS
MOVEQ $54,D0
NewHandle
LEA E4,A1
MOVE.L 2(A2,D5,W),(A1)
MOVE.L (A1),4(A1)
SUBQ $1,0(A1)
GUDQ $1,2(A1)
ADDQ $4,4(A1)
ADDQ $4,6(A1)
LEA 0(A2,D5,W),A1
MOVEA.L A0,A4
MOVEA.L (A0),A0
MOVEQ $26,D0
MOVE (A1)+,(A0)+
DBRA D0,01
MOVEA.L (A4),A0
ADDQ $1,10(A0)
ADDQ $1,14(A0)
ADDQ $1,34(A0)
ADDQ $1,36(A0)
ADDQ $1,40(A0)
ADDQ $1,44(A0)
ADDQ $1,46(A0)
ADDQ $1,50(A0)
KLS
CLR -(A7)
LEA E5,A3
MOVE.L (A3),-(SP)
PEA E4
PtInRect
TST (A7)+
RTS
RTS
PEA E4
InvertRect
RTS
DC 0,0,0,0
DC 0,0,0

```

## Source 'I/O.Asm'

```

;*****
; Routine d'ouverture, lecture et
; fermeture des fichiers (mode
; consultation).

```

```

.Trap _Open SA000
.Trap _Read SA002
.Trap _Close SA001

OuvertureL
LINK A6,#0
MOVEA.L 8(A6),A1
ADDQ.L #2,A1
MOVEQ #0,D0
MOVFR.R (A1)+,D0
LSL.L #8,D0
MOVE.B (A1),D0
LCL.L #8,D0
MOVE.B (A1),D0
LEA TamponIO,A0
MOVEA.L A0,A1
MOVEQ #24,D1
Boucle CLR (A1)+
DBRA D1,Boucle
MOVE.L D0,18(A0)

Sortie _Open
LEA Drapeau,A1
MOVE 16(A0),(A1)
UNLK A6
RTS

LectureL
LINK A6,#0
LEA TamponIO,A0
CLR.L 12(A0)
MOVEA.L 8(A0),A1
MOVE.B (A1)+,D0
LSL #8,D0
MOVE.B (A1)+,D0
MOVE D0,38(A0)
MOVEQ #0,D0
MOVFR.R (A1)+,D0
LSL.L #8,D0
MOVE.B (A1),D0
LCL.L #8,D0
MOVE.B (A1),D0
MOVE.L D0,32(A0)
MOVE #1,44(A0)
MOVE.L 12(A6),46(A0)

_Read
BRA.S Sortie

FermetureL
LINK A6,#0
LEA TamponIO,A0
CLR.L 12(A0)
Close
BRA.S Sortie
Drapeau DC 0

TamponIO
DCD.D 50,0

```

## Source 'Délai'

```

;*****
; Routine d'attente. Boucle pendant
; N interruptions vidéo.
; N est passé depuis le programme
; Basic par la pile (entier).

```

```

.Trap _TickCount SA975

LINK A6,#0
MOVE 8(A6),D3
EXT.L D3
SUBQ #4,SP
_TickCount
MOVE.L (SP)+,D4
ADD.L D3,D4
@1 SHRO #4,SP
_TickCount
CMP.L (SP)+,D4
BHI.S @1
UNLK A6
RTS

```

## Source 'Conversion.Asm'

```

;*****
; Routine de conversion des
; caractères non ASCII et des
; commandes Minitel.
; Le code du caractère ou de
; commande est passé par la pile.
; La suite de caractères
; résultante est placée dans la
; zone "tampon".
; Le premier octet de la zone
; indique le nombre de caractères
; à transmettre.

```

```

LINK A6,#0
MOVE 8(A6),D0
LEA Tampon,A0
MOVEQ #31,D1
LEA Table,A1
Boucle CMP.B (A1),D0
RNF.S Non
MOVFR.L (A1),(A0)
TOT 2(A1)
BEQ.S Un
TST.B 3(A1)
BEQ.S Deux
MOVE.B #3,(A0)
BRA.S Fin
Un MOVE.B #1,(A0)
RRA.C Fin
Deux MOVE.B #2,(A0)
DRA.G Fin
Non ADDQ.L #4,A1
DBRA D1,Boucle
SF (A0)
Fin UNLK A6
RTS

```

```

Table
DC.D 'a',#000,#019,#041,#061
,'A',#089,#019,#043,#061
,'c',#08D,#019,#04B,#063
,'e',#08F,#019,#047,#065
,'h',#08F,#019,#041,#065
,'i',#090,#019,#043,#065
,'o',#091,#019,#048,#065
,'t',#094,#019,#043,#069
,'v',#095,#019,#040,#069
,'w',#099,#019,#043,#06F
,'y',#09A,#019,#048,#06F
,'b',#09D,#019,#041,#075
,'n',#09E,#019,#043,#075
,'u',#09F,#019,#048,#075
,'r',#0A1,#019,#030,#000
,'l',#0A3,#019,#023,#000
,'s',#0A4,#019,#027,#000
,'B',#0A7,#019,#07B,#000
,'t',#0B1,#019,#031,#000
,'E',#0CF,#019,#06A,#000
,'e',#0CF,#019,#07A,#000
,'w',#0D6,#019,#038,#000
;E.Page

```

```

DC.B #07,#0D,#000,#000
;Envoi
DC.B #09,#013,#041,#000
;Retour
DC.B #0D,#013,#042,#000
;Repetition
DC.B #08,#013,#043,#000
;Guide
DC.B #02,#013,#044,#000
;Annulation
DC.B #03,#013,#045,#000
;Sommaire
DC.B #01,#013,#046,#000
;Correction
DC.B #04,#013,#047,#000
;Suite
DC.B #06,#013,#048,#000
;Connexion/Fin
DC.B #00,#013,#049,#000
Tampon DC.L 0

```

## Source 'MAJ.Asm'

```

;*****
; Routine de mise à jour de
; la fenêtre de "copie
; d'écran".
; Les caractères à afficher
; sont dans un tableau de
; variables entières.
; L'adresse du premier
; élément est passé par la
; pile.

```

```

.Trap _MoveLo $A893
.Trap _DrawChar $A883

LINK A6,#0
MOVEQ #-1,D3
MOVEA.L 8(A6),A3
MOVE (A3)+,D4
CR ADD #11,D3
MOVE #2,-(SP)
MOVE D3,-(SP)
_MoveLo
Boucle SUBQ #1,D4
BMI.S Fin
MOVEQ #0,D0
MOVE.B (A3)+,D0
CMP.L D #13,D0
CR BEQ.S CR
MOVE D0,-(SP)
_DrawChar
BRA.S Boucle
Fin UNLK A6
RTS

```

*Vous avez un  
Macintosh, mais  
pas le Basic ?*

Sur la disquette  
d'accompagnement,  
vous trouverez un  
'RUNTIME' du  
Basic Microsoft  
pour utiliser  
normalement les  
programmes de la  
revue...

Les programmes relatifs au Minitel que vous lirez dans ces pages, nécessitent une présentation sommaire des aspects *soft* et *hard* de nos terminaux télématiques (*hard* = matériel : il ne sera pas question de Minitel rose dans les lignes qui suivent !).

Il peut être utile de rappeler dans ce préambule que, dans les zones desservies par l'annuaire électronique, le Minitel est gratuit ce qui donne un modem d'un rapport qualité/prix... incalculable.

## Le 'hard'

Les programmes de ce numéro supposent que votre ordinateur est équipée d'une carte interface RS232, courante sur nos micros mais incompatible directement avec la V28 du Minitel, c'est ce qui impose de fabriquer ou d'acheter le câble de liaison enrichi d'un petit montage inverseur. Ce petit montage est peu coûteux : 30,00 à 50,00 F selon les connecteurs utilisés, si vous le faites vous-même, environ 200,00 F chez l'une des nombreuses sociétés qui le proposent, Olitec\* par exemple.

En ce qui concerne le raccordement des appareils, il est important de veiller à ce que le Minitel soit isolé du secteur avant de brancher et débrancher.

La prise péri-informatique à laquelle vous reliez votre Apple délivre et accepte des signaux à 1200 bauds, valeur par défaut.

La plupart des Minitels livrés aujourd'hui disposent d'une touche FNCT qui permet, entre autres, d'effectuer des copies d'écran vers ladite prise : FNCT-I puis A ou bien FNCT-I puis F déclenche le processus, ANNULATION l'interrompt.

Lors de ces copies d'écran, seuls les caractères ASCII sont transmis, les éventuels graphismes Vidéotex ne le sont pas. Pour changer la vitesse de transmission vers la prise :

- FNCT-P puis 3 donne 300

- bauds ;
- FNCT-P puis 4 donne 4800 bauds ;
- FNCT-P puis 1 donne 1200 bauds.

Les programmes qui suivent gèrent eux-mêmes les changements de vitesse, vous n'aurez donc pas à intervenir.

La prise se fait l'écho en permanence de ce que reçoit le Minitel, sauf lorsque vous l'inhibez en faisant FNCT P puis I, seules les copies d'écran passent alors.

Notons à l'attention de ceux qui se sentent une âme de 'télématiciens', que le protocole utilisé sur cette prise permet de s'adresser non seulement à une imprimante ou à un micro-ordinateur, mais également à un périphérique donné, parmi plusieurs autres branchés simultanément, presque AppleTalk...

Dans ses échanges avec le réseau, votre terminal reçoit les informations à 1200 bauds et les transmet à 75 ; les appareils livrés aujourd'hui sont tous retournables (vitesses inversées).

## Le soft

La gestion de l'écran du Minitel

n'est pas des plus simples : caractères ASCII (ABCD), polices 'parallèles' (βçàðœœ) et autres semi-graphiques, couleur de fond, couleur de caractère, masques. Ajoutons l'obligation d'écrire un espace de validation des changements d'attributs graphiques pour éclairer l'ensemble... Les programmes d'enregistrement Minitel de ce numéro stockent l'intégralité de ces informations (sans devoir les analyser) et lors de la consultation, vous retrouvez toutes vos couleurs.

Notons également que les caractères que vous tapez au clavier de l'Apple sont transmis au Minitel, transmis à l'éventuel serveur en ligne, affichés à l'écran du Minitel puis transmis vers la prise où votre Apple pourra les recueillir pour les afficher. Lorsque votre Apple transmettra un texte à un serveur, le programme tiendra compte du débit et du temps nécessaire au serveur pour interpréter les codes spéciaux de type SUITE.

Pour vos programmes, nous donnons en annexe la table de correspondance des caractères disponibles sur vos Apple avec ceux du mini-terminal ainsi que ses caractères semi-graphiques.

32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95

Jeux de caractères graphiques Vidéotex (ASCII décimal)

En attendant nos prochains programmes de communications via Minitel, ne manquez pas de nous faire part de vos remarques et suggestions.

\* Sarl Olitec - 20, rue de Réménauville - 54000 NANCY  
Tél : 16 83 35 00 65  
Tarif : 225,00 F, pour Apple ][+, //e, //c et Macintosh.

Table de correspondance

	Mac	Apple //	Minitel
' '	32	32	32
' !'	33	33	33
' "'	34	34	34
' #'	35	35	35
' \$'	36	36	36
' %'	37	37	37
' &'	38	38	38
' "'	39	39	39
' ('	40	40	40
' )'	41	41	41
' +'	42	42	42
' +'	43	43	43
' ,'	44	44	44
' .'	45	45	45
' /'	46	46	46
' 0'	47	47	47
' 1'	48	48	48
' 2'	49	49	49
' 3'	50	50	50
' 4'	51	51	51
' 5'	52	52	52
' 6'	53	53	53
' 7'	54	54	54
' 8'	55	55	55
' 9'	56	56	56
' :	57	57	57
' ;'	58	58	58
' <'	59	59	59
' ='	60	60	60
' >'	61	61	61
' ?'	62	62	62
' @'	63	63	63
' A'	64	64	64
' B'	65	65	65
' C'	66	66	66
' D'	67	67	67
' E'	68	68	68
' F'	69	69	69
' G'	70	70	70
' H'	71	71	71
' I'	72	72	72
' J'	73	73	73
' K'	74	74	74
' L'	75	75	75
' M'	76	76	76
' N'	77	77	77
' O'	78	78	78
' P'	79	79	79
' Q'	80	80	80
' R'	81	81	81
' S'	82	82	82
' T'	83	83	83
' U'	84	84	84
' V'	85	85	85
' W'	86	86	86
' X'	87	87	87
' Y'	88	88	88
' Z'	89	89	89
' ['	90	90	90
' \'	91		91
' ]'	92		92
' ^'	93		93
' _'	94	94	94
' `'	95	95	95
' 0'	96	96	96
' a'	97	97	97
' b'	98	98	98

' c'	99	99	99
' d'	100	100	100
' e'	101	101	101
' f'	102	102	102
' g'	103	103	103
' h'	104	104	104
' i'	105	105	105
' j'	106	106	106
' k'	107	107	107
' l'	108	108	108
' m'	109	109	109
' n'	110	110	110
' o'	111	111	111
' p'	112	112	112
' q'	113	113	113
' r'	114	114	114
' s'	115	115	115
' t'	116	116	116
' u'	117	117	117
' v'	118	118	118
' w'	119	119	119
' x'	120	120	120
' y'	121	121	121
' z'	122	122	122
' {'	123		123
'  '	124		124
' }'	125		125
' ~'	126		126
del	127	127	127
' à'	136	64	25
' á'	137		25
' ç'	141	92	25
' é'	142	123	25
' è'	143	125	25
' ê'	144		25
' ë'	145		25
' ì'	148		25
' í'	149		25
' ò'	153		25
' ó'	154		25
' ü'	157	124	25
' ũ'	158		25
' Ů'	159		25
' ı'	161	91	25
' £'	163	35	25
' \$'	164	93	25
' ß'	167		25
' ±'	177		25
' Œ'	206		25
' œ'	207		25
' +'	214		25
' ←'			25
' ↑'			25
' →'			25
' ↓'			25
' 1/4'			25
' 1/2'			25
' 1/3'			25
BRK			0
E.Page			13
Envoi			19
Retour			19
Répétition			19
Guide			19
Annulation			19
Sommaire			19
Correction			19
Suite			19
Connexion/Fin			19

\* Indique que le caractère est différent, mais il existe. Par exemple, le 'V' du Minitel correspond au 'v' du Macintosh.

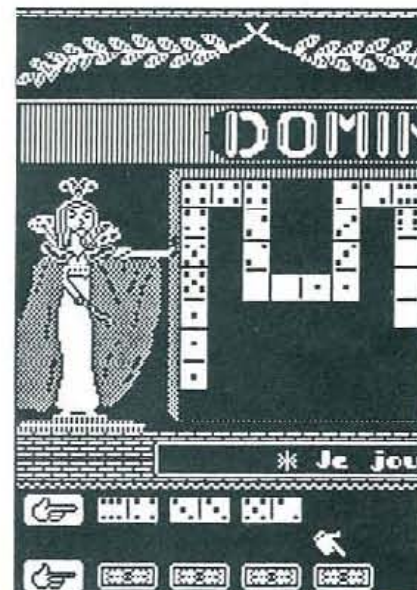


# Pom's vous propose "Dominos"

Apple ][+, //e, //c

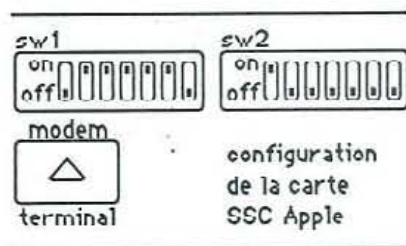
Thierry Haurie

Il est inutile de présenter le jeu de dominos; celui-ci bénéficie d'un graphisme très soigné (en couleur si vous disposez d'une carte "Chat Mauve") et les messages transmis par le programme sont, au choix, en Français, en Italien, en Allemand ou en Anglais.



80.00 F TTC  
franco  
Bon de commande  
page 74

Le présent programme a pour but d'enregistrer des séquences de consultations Minitel, de les restituer, d'enregistrer et d'imprimer des écrans. Il a été conçu sur un Apple IIe, et fonctionne également sur un II+, un IIc et même un IIGS avec, dans ce cas, un gain de temps sensible à l'affichage et à l'impression. Pour les II+, IIe et IIGS, la carte SSC est indispensable et doit être configurée comme ci-dessous. Elle sera installée dans l'un des slots, entre 1 et 5.



Dans un des mêmes slots, une imprimante pourra être connectée. Sur l'Apple IIc, une routine fort pertinemment nommée PARAM configurera le port 2 pour le Minitel et le port 1 servira à l'imprimante.

S'il vous manque l'interface Minitel ou celle pilotant l'imprimante, seules les fonctions liées au périphérique concerné seront indisponibles. Autrement dit il sera toujours possible d'enregistrer et de consulter des informations du Minitel sans imprimante ou d'imprimer des informations préalablement sauvegardées si votre interface série n'est pas connectée.

## La liaison

Les interfaces RS 232 (SSC) et V28 (Minitel) délivrant des signaux série incompatibles, il vous faut prendre votre fer à souder pour réaliser le montage inverseur ci-contre. Les composants trouveront leur place dans l'une des prises ou dans un boîtier à part.

Si vous doutez de vos talents d'électronicien, vous pourrez vous adresser à l'une des nombreuses sociétés qui proposent un tel câble : citons par exemple la Sarl OLITEC\*

## Les deux modes principaux

Le programme proposé ici peut recevoir du Minitel, restituer sur Minitel, sauvegarder et recharger ce que nous appellerons par la suite *séquences*. Il s'agit des pages Minitel complètes, chargées de tous leurs attributs graphiques ; l'ordinateur enregistre dans ce mode tous les signaux disponibles sur la prise péri-informatique. La consultation de ces séquences se fait sur l'écran du Minitel, les pages réapparaissant exactement comme lors de l'enregistrement mais plus rapidement.

Dans le mode que nous appellerons *copies d'écran*, l'ordinateur n'enregistrera que les codes ASCII délivrés par le Minitel lorsqu'il effectue un copie d'écran. Les écrans enregistrés dans ce mode ne sont pas restitués sur le Minitel mais sur votre imprimante ou bien transférés dans un fichier de type 'TEXT' pour exploitation future par un traitement de textes. Vous opterez pour ce mode dans la consultation de l'annuaire électronique, pour, par exemple récupérer des adresses. Le premier mode servira surtout pour consulter sans contrainte de temps des informations qui ne nécessitent pas l'impression. Rappelons que, sur les Minitel qui le permettent, la copie d'écran s'obtient par FNCT-I puis F, ou FNCT-I puis A selon que vous désirez ou non les minuscules accentuées.

## Un mode auxiliaire

De façon accessoire, le programme vous permet de transmettre au Minitel un fichier de type 'TEXT' que vous aurez préparé en traitement de textes. Ceci évite de perdre un temps coûteux lorsqu'il s'agit de taper un texte à destination d'une messagerie ou pour utiliser à moindres frais le futur service des PTT, Postclair/Minitel.

## Le programme Basic...

...ne joue pas activement dans la relation Minitel/Apple mais gère les différents menus, intercepte les éventuelles erreurs DOS, s'occupe des chargements et sauvegardes ainsi que de la consultation des copies d'écrans. Lors de vos adaptations de ce programme, il faut garder en mémoire les deux points suivants :

- la place disponible une fois les variables initialisées est très réduite ;
- il ne fonctionne tel quel qu'avec le DOS, mais comme pour le jeu SNAKE paru récemment, ne n'est pas un handicap puisque le programme est autonome.

## La partie assembleur...

...est appelée par le Basic par l'ordre

CALL AD, SL, COMMANDE

AD étant l'adresse de chargement de la routine, SL étant le port du Minitel et COMMANDE prenant une valeur de 1 à 5.

Commande = 1

La routine est en mode

enregistrement de séquences, les caractères reçus sont stockés de \$2600 à \$95FF, chaque adresse de 'début de page' Minitel est stocké entre \$2500 et \$25FF. Une nouvelle page est repérée par l'utilisation d'une fonction Minitel (Correction, Annulation, Guide...), uniquement si au moins 100 caractères ont été reçus depuis le dernier appel à une fonction.

Ainsi, à la restitution, l'ordinateur ne marquera pas de pause si CORRECTION n'a fait que supprimer un caractère dans une zone de saisie mais il s'arrêtera si CORRECTION vous a scrvi à revenir au menu de l'annuaire électronique.

A 1500 caractères de la saturation, la routine émetts quelques 'bips' et, à \$95FF on se dirige vers le Basic qui propose la sauvegarde.

La routine lit en permanence le clavier et se charge de la conversion des ordres clavier en codes Minitel (Envoi, Suite, Correction...) ainsi que du recodage des caractères non ASCII ('é' 'è' 'ç' 'à' 'ù'...).

#### Commande = 2

C'est le mode restitution des séquences, les caractères sont lus en mémoire, et envoyés au Minitel via la carte interface série, une pause étant marquée entre chaque 'page'.

#### Commande = 3

C'est le mode 'enregistrement de copies d'écran'. La prise n'étant par forcément inhibée sur le Minitel, les éventuels douze caractères destinés aux périphériques du Minitel sont supprimés. On compte par la suite les retours-chariot pour repérer la limite entre les différentes copies d'écran. Chaque fin de copies d'écran est repérée par un code \$FE dans le stockage, et la fin des copies par \$FF.

#### Commande = 4

La routine ne fait qu'émettre un 'Bip' plus agréable que l'habituel CTRL-G.

## Comment faire ?

- 1 Réalisez le câble interface,
- 2 Vous avez un //c ? passez à l'étape 4
- 3 Configurez la carte Série Apple SSC comme indiqué,
- 4 Vous avez la disquette Pom's ? passez à l'étape 8
- 5 Saisissez et sauvegardez le programme Basic (sur disquette DOS),
- 6 Saisissez et sauvegardez le code MINITEL,
- 7 Vous avez un Apple //c ? saisissez et sauvegardez le code PARAM,
- 8 Faites 'RUN MINBAS'.

Le mode d'emploi ci-contre et les menus vous guideront...

#### Commande = 5

Ce mode est utilisé pour l'envoi d'un fichier 'TEXT' au Minitel : La routine prend le caractère que le Basic a stocké en \$6, éventuellement le recode (cas des 'é' 'è' 'ç' 'à' 'ù'...) et l'envoie au Minitel.

En fin de routine sont installés les habituels octets de correction du bug de l'ON ERR.

#### Sur le //c

La carte série est réglé à 1200 bauds, parité paire, 7 bits de données, 1 bit de stop à l'aide de CONFIG. Les trois octets nécessaires sont installés dans la mémoire principale dans un 'trou' de la mémoire-écran, puis transférés en mémoire auxiliaire par MOVEAUX.

## Mode d'emploi

L'ensemble des choix dans les menus se fait par l'utilisation de l'une des quatre flèches, ESC signifie abandon et provoque le retour au menu précédent. On accepte l'option visée par la flèche à l'intérieur d'un menu avec RETURN.

La première étape, indispensable, consiste à définir la configuration pour indiquer les numéros de ports de la SSC reliée Minitel et de l'imprimante. Par défaut, il est prévu 2 et 1. Vous changerez ce choix par défaut en modifiant la ligne Basic n° 20 (SL est le port Minitel et SI le port Imprimante).

## Minitel Vers Apple

### Enregistrement de séquences

```

-----
DOM'S          MINITEL/APPLE          DOM'S
-----
                                MENU GENERAL
                                *--> MINITEL VERS APPLE
                                CONSULTATION
                                CHARGER
                                SAUVEGARDEE
                                OPTIONS
                                ENVOI D'UN FICHIER
                                FIN
-----

```

Il faut se servir du clavier de l'Apple // en lieu et place de celui du Minitel. Ceci permettra à l'ordinateur de repérer les débuts de 'page'. Un rappel des touches de fonction est alors affiché :

```

CTRL-X = Annulation
DEL ou CTRL-C = Correction,
TAB ou CTRL-I = Suite,
RETURN = Envoi...

```

```

-----
DOM'S          MINITEL/APPLE          DOM'S
-----
                                MINITEL VERS APPLE
                                *--> SEQUENCES
                                COPIER D'ECRAN
                                RETOUR
-----

```

ESC interrompt de façon provisoire l'enregistrement, une touche quelconque relance alors l'enregistrement, sauf ESC qui l'interrompt définitivement, le Basic vous proposant de sauvegarder. Si la sauvegarde n'est pas exécutée (vous avez



répondu par un simple RETURN au nom de fichier), l'option SAUVEGARDE du menu principal sera en vidéo inverse, à titre de rappel.

## Enregistrement de copies d'écran

Avant chacune d'elles, vous devrez faire RETURN, l'Apple comptant alors les retours-chariot qui passent... Si la copie d'écran est interrompue par ANNULLATION sur le Minitel ou bien si vous avez fait RETURN après le début de la copie, faites ESC pour indiquer la fin de page; l'Apple en attend alors une nouvelle. ESC permet d'abandonner ce mode et, bien entendu, la sauvegarde est proposée.

## Consultation

### Séquences

La consultation se fera sur le Minitel. Si ce dernier est récent, vous pourrez optez dans le dernier menu pour 1800 bauds, l'affichage devient alors très agréable. Les flèches vous permettent d'avancer et de reculer dans la consultation, et ESC de conclure.

### Copies d'écran

La consultation consistera en l'impression ou la création d'un fichier 'TEXT'. Sur ImageWriter/DMP, l'impression se fait en Élite élargi (code de contrôle à la fin de la ligne 795, à adapter à votre imprimante). Sur les autres imprimantes, aucun code de contrôle n'est envoyé.

Vous pourrez également créer un fichier 'TEXT' à partir de vos copies d'écran. Dans les deux cas, les copies sont séparées par une ligne de 40 '-'.  
 Transmettre à l'ordinateur des séquences alors qu'il attend des copies d'écrans provoquera des effets étonnants lors d'une éventuelle consultation à l'imprimante...

Transmettre à l'ordinateur des séquences alors qu'il attend des copies d'écrans provoquera des effets étonnants lors d'une éventuelle consultation à l'imprimante...

## Chargement

```

-----
POM'S          MINITEL/APPEL  POM'S
-----
                                CHARGER
                                *--> LECTEUR 1
                                LECTEUR 2
                                RETOUR
  
```

Le nom de fichier demandé peut toujours être remplacé par un '?' pour obtenir le catalogue des disquettes. Un simple Retour-Chariot signifie que vous renoncez à l'opération mais, bien sûr, vous pourrez renoncer à renoncer... Le choix du lecteur vous est proposé après saisie du nom et, lors de l'opération, les différentes erreurs disques sont interceptées. Selon leur type (séquences ou copies d'écran), les noms de fichiers sont préfixés par 'S.' ou 'C.'. Ces préfixes ne servent qu'à vous repérer dans le catalogue de la disquette; il est inutile de les taper car ils seraient simplement ignorés.

## Sauvegarde

C'est le menu par lequel vous passez obligatoirement à l'issue d'un enregistrement. Mêmes conventions que pour le chargement.

## Options

```

-----
POM'S          MINITEL/APPEL  POM'S
-----
                                OPTIONS
                                *--> PORT MINITEL
                                PORT IMPRIMANTE
                                RETOUR
  
```

Ce menu vous permet de changer les numéros de port par défaut pour les Minitel et Imprimante. Si aucune imprimante n'est connectée, il convient de mettre NEANT et non de laisser une

autre valeur car les résultats d'une tentative d'impression sont... prévisibles.

## Envoi d'un fichier

```

-----
POM'S          MINITEL/APPEL  POM'S
-----
                                OPTIONS - PORT MINITEL
                                PORT 1
                                *--> PORT 2
                                PORT 3
                                PORT 4
                                PORT 5
                                NEANT
                                RETOUR
  
```

Ce fichier doit être de type TEXT, certainement créé par un traitement de textes. Les Retours Chariot [CHR\$(13)] sont transformés (ligne 4440) en les codes 19 72 qui équivalent à SUITE pour le Minitel, les serveurs attendant généralement ce code pour passer à la ligne. La boucle de délai qui suit permet audit serveur d'exploiter ce code SUITE.



## Source 'T.Param'

(uniquement //c)

```

1 *-----
2 * Paramètres pour
3 * port 2 du //c,
4 * $38, $6B, $81 dans
5 * les octets $47C à
6 * $47E en mémoire
7 * auxiliaire.
0 *-----
9          ORG  $300
10         OBJ  $300
11
12         LDA  #$38
13         STA  $47C
14         LDA  #$6B
15         STA  $47D
16         LDA  #$81
17         STA  $47E
18         LDA  #$7C
19         STA  $42
20         STA  $3C
21         LDA  #4
22         STA  $43
23         STA  $3D
24         LDA  #$7E
25         STA  $3E
26         LDA  #4
27         STA  $3F
28         SEC
29         JMP  $C311
  
```

# Source 'T.MINITEL'

## Assembleur Big Mac

Note : ce source est sauvegardé, sur la disquette d'accompagnement Pom's 27, en format 'TEXT' pour permettre la récupération par votre assembleur.

```

1 * 411
2 *-----
3 * Minitel/Apple
4 *
5 * Syntaxe :
6 * CALL 8266, SLOT, COM
7 *-----
8
9          LST  OFF
10         ORG  $204A
11
12 CARAC   =   $6
13 STOCK  =   $7
14 PAGE   =  $18
15 DRAP   =  $1A
16 COMPT  =  $1C
17 DRAP1  =  $1E
18 CH     =  $24
19 CV     =  $25
20 PTFE   =  $FE
21 STATUS =  $EB
22 DATA  =  $FD
23 SLOT   =  $FC
24 KBD    =  $C000
25 STROBE =  $C010
26 HP     =  $C030
27 STROUT =  $DB3A
28 GETBYT =  $E6F5
29 BASCALC =  $FBC1
30 CLREOL =  $FC9C
31 KEYIN  =  $F00C
32 COUT   =  $FDED
33
34 *-----
35 * Récupère n° slot et
36 * n° commande
37 *-----
38
39         JSR  GETBYT
40         TXA
41         PHA
42         CLC
43         ADC  #$C0
44         STA  SLOT+1
45         PLA
46         ASL
47         ASL
48         ASL
49         ASL
50         ADC  #$89
51         STA  STATUS
52         STA  DATA
53         DEC  DATA
54         LDA  #$C0
55         STA  STATUS+1
56         STA  DATA+1
57         LDA  #0
58         STA  SLOT
59         JSR  GETBYT

```

```

60         TXA
61         CMP  #1
62         BEQ  ENREG
63         CMP  #2
64         BNE  DEB1
65         JMP  LECTURE
66 DEB1    CMP  #3
67         BNE  DEB2
68         JMP  COPIECR
69 DEB2    CMP  #4
70         BNE  DEB3
71         JMP  BEEP
72 DEB3    CMP  #5
73         BNE  ERR
74         JMP  ENCAR
75 ERR     RTS
76
77 *-----
78 * Mode enregistrement
79 *-----
80
81 MESEN   JSR  VTAB
82         LDY  #>ME
83         LDA  #<ME
84         JSR  STROUT
85         RTS
86 ENREC
87
88         JSR  MESEN
89         JSR  INITAD
89 BCL
90         RTT  KRD
91         BMI  CLAV
92         LDX  #0
93         LDA  (STATUS, X)
94         AND  #08
95         BEQ  BCL
96 SSC
97         LDA  (DATA, X)
98         LDY  #0
99         STA  (STOCK), Y
100        ORA  #10000000
101        STA  $7F7
102        INC  STOCK
103        BNE  SS2
104        INC  STOCK+1
105        LDA  STOCK+1
106        CMP  #$96
107        BNE  SS2
108        DEC  STOCK
109        DEC  STOCK+1
110        JSR  VTAB
111        LDY  #>MM
112        LDA  #<MM
113        JSR  STROUT
114 SS1     JSR  KEYIN
115        JMP  FINER
116 SS2
117        INC  COMPT
118        BNE  SS3
119        INC  COMPT+1
120 SS3
121        JMP  BCL
122 CLAV
123        BIT  STROBE
124        LDA  KBD
125        ORA  #$80
126        STA  CARAC
127        CMP  #$9B
128        BNE  S1

```

```

129        JSR  VTAB
130        LDY  #>MO
131        LDA  #<MO
132        JSR  STROUT
133        JSR  KEYIN
134        CMP  #$9B
135        BEQ  FINER
136        JSR  MESEN
137        JMP  BCL
138 FINER   LDY  #0
139        LDA  #$FF
140        STA  (STOCK), Y
141        TNY
142        INY
143        INY
144        STA  (PAGE), Y
145        JMP  VTAB
146 S1
147
148 BCL1    LDY  #0
149        LDA  TABLE, X
150        BEQ  FIN
151        CMP  CARAC
152        BEQ  OK
153        INX
154        INX
155        INX
156        BNE  BCL1
157 OK
158        TXA
159        CMP  #45
160        BCS  SUITE
161 OK1     LDY  #0
162        LDA  COMPT+1
163        BNE  PLDE100
164        LDA  COMPT
165        CMP  #100
166        BCS  PLDE100
167 MODE100
168        ADC  (PAGE), Y
169        STA  (PAGE), Y
170        BCC  MO1
171        INY
172        SEC
173        BCS  MO2
174 MO1     INY
175 MO2     LDA  COMPT+1
176        ADC  (PAGE), Y
177        STA  (PAGE), Y
178        JSR  CONTP
179        JSR  INITCO
180        JMP  SUITE
181 PLDE100
182        CLC
183        LDA  COMPT
184        ADC  (PAGE), Y
185        PHA
186        BCC  PL1
187        INY
188        SFC
189        BCS  PL2
190 PL1     INY
191 PL2     LDA  COMPT+1
192        ADC  (PAGE), Y
193        INC  PAGE
194        INC  PAGE
195        LDY  #1
196        STA  (PAGE), Y
197        JSR  CONTP

```

198	DEY	267 TOUCH	336 SSC0
199	PLA	268 LDA KBU	337 LDA CARAC
200	STA (PAGE), Y	269 RPT. TOUCH	338 CMP #19
201	JSK INITCU	270 BIT STROBE	339 BNE SSC00
202 SHITE		271 CMP #S9B	340 LDY #12
203	LDY #3	272 BEQ FINFICH	341 SSC01 JSR GETDATA
204 BCL2	INX	273 CMP #S8A	342 BCS CO2
205	LDA TABLE, X	274 BEQ MOINS	343 DEY
206	BEQ FINCH	275 RFP #S88	344 BNE SSC01
207	STA CARAC	276 RFO MOTNS	345 SSC00
208	JSK ENVOI	277 CMP #S95	346 LDA CARAC
209	DEY	278 BEO PC	347 CMP #13
210	BNE BCL2	279 CMP #S0D	348 DNE SSC02
211 FINCH		280 BEQ PC	349 DEC DRAP1
212	JMP BCL	281 M1 JSR BEEP	350 SSC02
213 FIN		282 JMP TOUCH	351 LDA CARAC
214	JSR ENVOI	283 MOINS	352 CMP #17
215	JMP BCL	284 SEC	353 BEQ SSC022
216		285 LDA PAGE	354 CMP #S20
217 *-----		286 SBC #2	355 DCC CO1
218 * Mode lecture		287 BEQ PRPAGE	356 SSC022 LDY #0
219 *-----		288 SEC	357 STA (STOCK), Y
220		289 SBC #2	358 ORA #S1000000U
221 MESLEC		290 STA PAGE	359 STA S7F7
222	JSK VTAB	291 JSR CLS	360 INC STOCK
223	LDY #>ML	292 JMP PC	361 BNE SSC2
224	LDA #<ML	293 PRPAGE	362 INC STOCK+1
225	JMP STROUT	294 JSR BEEP	363 LDA STOCK+1
226 LECTURE		295 JSR VTAB	364 CMP #S96
227	JSR MESLEC	296 LDY #>MP	365 BNE SSC2
228	JSR CLS	297 LDA #<MP	366 SSC03 DEC STOCK
229	LDY #0	298 JSK STROUT	367 DEC STOCK+1
230	STY PAGE	299 JMP D1	368 JSR VTAB
231	STY STUCK	300 RTS	369 LDY #>MM
232	LDA #S25	301 DERPAGE	370 LDA #<MM
233	STA PAGE+1	302 JSR BEEP	371 JSR STROUT
234	STA STOCK+1	303 JSR VTAB	372 JSR KEYIN
235	INC STOCK+1	304 LDY #>MD	373 JMP FINCOPIE
236	BNE LEC	305 LDA #<MU	374 SSC2
237 PC		306 JSR STROUT	375 RTT DRAP1
238	LDY #1	307 D1 JSR KEYIN	376 BPL SSC3
239	LDA (PAGE), Y	308 JSR MESLEC	377 INC DRAP1
240	CMP #SFF	309 JMP TOUCH	378 INC DRAP
241	BEO DERPAGE	310 FINFICH	379 LDA DRAP
242	STA STOCK+1	311 JSR VTAB	380 CMP #24
243	DEY	312 LDY #>MF	381 BEQ PAUSE
244	LDA (PAGE), Y	313 LDA #<MF	382 SSC3
245	STA STOCK	314 JSK STROUT	383 JMP CO1
246 LEC	INC PAGE	315 JSR KEYIN	384 PAUSE
247	INC PAGE	316 AND #S11011111	385 LDA STOCK+1
248 LEC1		317 CMP #0	386 JSR CONTP
249	LDY #0	318 DEQ F	387 LDY #S0
250	LDA (STOCK), Y	319 SR MESLEC	388 LDA #SFE
251	BMI DERPAGE	320 JMP TOUCH	389 STA (STOCK), Y
252	STA CARAC	321 F' JMP VTAB	390 INC STUCK
253	ORA #S10000000	322	391 BNE PAUSE2
254	STA S7F7	323 *-----	392 INC STOCK+1
255	JSR ENVOI	324 * Mode enregistrement	393 LDA STOCK+1
256	INC STUCK	325 * de copies d'écian	394 CMP #S96
257	RNE LEC0	326 *-----	395 BNE PAUSE2
258	INC STOCK+1	327	396 BEQ SSC03
259 LECO	LDY #0	328 COPIECR	397 PAUSE2 JSR VTAB
260	LDA STOCK	329 JSR INITAD	398 LDY #>MA
261	CMP (PAGE), Y	330 JMP PAUSE	399 LDA #<MA
262	BNE LEC1	331 CO1 JSR GETDATA	400 JSR STROUT
263	INY	332 BCU SSCU	401 JSR KEYIN
264	LDA STOCK+1	333 CO2 CMP #S1B	402 CMP #S9B
265	CMP (PAGE), Y	334 BNE CO1	403 BEQ FINCOPIE
266	RNE LEC1	335 JMP PAUSE	404 JSR INIDRAP

405	JSR	MESECR	474	JSR	INITCO	543	ENCAR		
406	JMP	COI	475	RTS		544	LDA	CARAC	
407	GETDATA	BIT	KBD	476	INITCO	545	ORA	##10000000	
408	BMI	GET1	477	LDA	#0	546	STA	CARAC	
409	IDX	#0	478	STA	COMPT	547	IDX	#0	
410	LDA	(STATUS,X)	479	STA	COMPT+1	548	ENCARI	LDA	TABLE,X
411	AND	#8	480	RTS		549	BEQ	TABLER	
412	BEQ	GETDATA	481			550	CMP	CARAC	
413	LDA	(DATA,X)	482	*-----		551	BEQ	OKCAR	
414	STA	CARAC	483	* Bientôt saturé ?		552	INX		
415	ULC		484	*-----		553	INX		
416	RTS		485			554	INX		
417	GET1	BIT	STROBE	486	CONTP	555	INX		
418	LDA	KBD	487	CMP	#\$90	556	BNE	ENCARI	
419	SEC		488	BCC	C1	557	FNCAR	JMP	ENVOI
420	RTS		489	JSR	BEEP	558	OKCAR	LDY	#3
421	MESECR		490	C1	RTS	559	OKCARI	INX	
422	JSR	VTAB	491			560	LDA	TABLE,X	
423	LDY	>MC	492	*-----		561	REQ	FTNCAR0	
424	LDA	<MC	493	* bip bref		562	STA	CARAC	
425	JMP	STROBT	494	*-----		563	JSR	ENVOI	
426	FINCOPIE		495			564	DEY		
427	LDY	#0	496	BEEP		565	BNE	OKCARI	
428	LDA	#\$FF	497	TYA		566	FNCAR0	RTS	
429	STA	(STOCK),Y	498	PHA		567			
430	JMP	VTAB	499	TXA		568	*-----		
431	INIDKAP		500	PHA		569	* Messages		
432	LDA	#0	501	IDY	#\$40	570	*-----		
433	STA	DRAP	502	C	TYA	571			
434	STA	DRAP1	503	ROR		572	ML	ASC	"MODE
435	RTS		504	TAX					LECTURE"
436			505	D	DEX	573	BRK		
437	*		506	DNE	D	574	ME	ASC	"MODE ENREGIS
438	* Caractère --> SSC		507	BIT	HP				TREMENT"
439	*-----		508	DEY		575	BRK		
440			509	BNE	C	576	MC	ASC	"MODE COPIE D
441	ENVOI		510	PLA					'FCRAN"
442	TXA		511	TAX		577	BRK		
443	PHA		512	PLA		578	MP	ASC	"PREMIERE PAG
444	ENVOI1	LDX	#0	513	TAX				E <RETURN>"
445	LDA	(STATUS,X)	514	RTS		579	BRK		
446	AND	#\$10	515			580	MD	ASC	"DERNIERE PAG
447	BEQ	ENVOI1	516	*-----					E <RETURN>"
448	LDA	CARAC	517	* Efface écran Minitel		581	BRK		
449	STA	(DATA,X)	518	*-----		582	MM	ASC	"MEMOIRE SATU
450	PLA		519						REE. ENREGIS
451	TAX		520	CLS					TREZ <RETURN
452	RTS		521	LDA	#12				>"
453			522	STA	CARAC	583	DRK		
454	*-----		523	JSR	ENVOI	584	MO	ASC	"PAUSE. <ESC>
455	* Init les compteurs		524	RTS					POUR FINIR"
456	*-----		525			585	BRK		
457			526	*-----		586	MF	ASC	"OK POUR FIN
458	INITAD		527	* Efface les messages					DE CONSULTA
459	LDA	#0	528	*-----					TION ?"
460	STA	STOCK	529			587	BRK		
461	STA	DRAP	530	VTAB		588	MA	ASC	"RETURN = COP
462	STA	DRAP1	531	LDA	#0				IE, ESC - FI
463	STA	PAGE	532	STA	CH				N "
464	LDA	#\$26	533	LDA	#23	589	BRK		
465	STA	STOCK+1	534	STA	CV	590			
466	STA	PAGE+1	535	JSR	BASCALC	591	*-----		
467	DEC	PAGE+1	536			592	* Correspondance codes		
468	LDY	#0	537	JMP	CLREOL	593	* Apple // - Minitel		
469	LDA	#0	538			594	*-----		
470	STA	(PAGE),Y	539	*-----		595			
471	LDA	#\$26	540	* Envoi de caractères		596	TABLE		
472	INY		541	*-----		597	* RC = Envoi		
473	STA	(PAGE),Y	542			598	DFB	#\$8D, \$13, \$41, 0	

599 * Ctrl-K - Connexion	617	DFD	\$82, \$13, \$43, 0	632 * *		
600	DFB	\$8B, \$1B, \$39, \$		633	DFB	\$DD, \$19, \$27, 0
		68		634 * ù		
601 * Ctrl-W = Deconnexion	620 * a			635	DFB	\$FC, \$19, \$41, \$
602	DFB	\$97, \$13, \$49, 0	621	DFR	\$00, \$19, \$41, \$	75
603 * Ctrl-I = Tab = Suite			61		DFB	0
604	DFR	\$A9, \$13, \$48, 0	622 * c			
605 * Del = Correction	623	DFD	\$DC, \$12, \$4D, \$	637 * -----		
606	DFB	\$FF, \$13, \$47, 0	63	638 * Bug de 1'ONERR GOTO		
607	DFD	\$83, \$13, \$47, 0	624 * ó	639 * -----		
608 * Ctrl-S = Sommaire	625	DFB	\$FB, \$19, \$42, \$	640		
609	DFB	\$93, \$13, \$46, 0	65	641	PLA	
610 * Ctrl-G = Guide	626 * è			642	TAY	
611	DFB	\$87, \$13, \$44, 0	627	DFB	\$FD, \$19, \$41, \$	
612 * Ctrl-X = Annulation			65	643	PLA	
613	DFB	\$98, \$13, \$45, 0	628 * "	644	IDX	SDF
614 * Ctrl R - Retour	629	DFB	\$DB, \$19, \$30, 0	645	TXS	
615	DFB	\$92, \$13, \$42, 0	630 * E ou #	646	PHA	
616 * Ctrl-B = Répétition	631	DFB	\$A3, \$19, \$23, 0	647	TYA	
				648	PHA	
				649	RTS	

## Programme 'MINBAS'

```

1 PRINT CHR$(21)
2 ONERR GOTO 5000
5 PRINT CHR$(4) "BLOADMINITEL
10 HIMEM: 8265: TEXT: HOME: COSUB 2000:
PRINT "POM'S" SPC(9) "MINITEL/APPLE" SP
C(8) "POM'S": GOSUB 2000: VTAB 22: COS
UB 2000: VIAB 10: HTAB 10: POKE 34, 4: P
OKF 35, 21
15 CX = 5: DIM A$(9): NFS = "NOM DU FICHER
"
20 DIM M1$(6), M2$(2), M3$(2), M4$(6), M5$(4),
M$(6): GOSUB 1000: AD = 8266: SI = 2: EN =
1: LE = 2: CE = 3: BE = 4: SI = 1: POKE 97
28, 255: DS = CHR$(13) + CHR$(4): DR =
0: D1$ = CHR$(4): CU = 1

```

### Si l/c, paramètres de l'interface modem

```

30 IF PEEK(64435) = 6 AND PEEK(64448)
- 0 THEN PRINT D$ "DRUNPARAM

```

### Menu principal

```

100 FOR I = 0 TO 6: M$(I) = M1$(I): NEXT : C
H = A0: TTS = "MENU GENERAL": GOSUB 1100
: A0 = CH: A2 = 0: CT = 0
105 ON (CH - 0 OR CH - 5) AND SL - 6 GOTO
4300
110 ON CH + 1 GOTO 150, 160, 250, 350, 200, 440
0, 3000
150 FOR I = 0 TO 2: M$(I) = M2$(I): NEXT : C
H = A1: TTS = M1$(A0): GOSUB 1100: A1 = C
H: ON CH + 1 GOTO 100, 500, 100
160 HOME: PRINT: IF PEEK(9728) = 255 O
R PEEK(9729) - 255 THEN PRINT "RIEN
A CONSULTER, FAITES RETURN...": CALL A
D, SL, BE: GOSUB 6000: A0 - 2: GOTO 100
170 GOTO 150

```

### Charger un fichier

```

200 FOR I = 0 TO 2: M$(I) = M3$(I): NEXT : C
H = A2: TTS = M1$(4): GOSUB 1100: A2 = CH
: A0 = 0: ON CH + 1 GOTO 600, 700, 100
250 FOR I = 0 TO 2: M$(I) = M2$(I): NEXT : C
H = A1: TTS = M1$(2): GOSUB 1100: A1 = CH
252 ON A1 - 2 GOTO 100
255 HOME: VTAB 8: PRINT NFS "A CHARGER OU

```

```

" CHR$(34) "?" CHR$(34): GOSUB 4000: I
F NFS = "" THEN 250
260 IF CT THEN CT = 0: GOTO 255
270 GOSUB 6100
280 PRINT D$ "BLOAD" NFS, A$2500: A0 = 1: LF
= PEEK(43617) * 256 + PEEK(43616) +
9471: POKE 8, INT(LF / 256): POKE 7, L
F - PEEK(8) * 256: GOSUB 4100: DK = A1
: FS = 0: GOTO 100

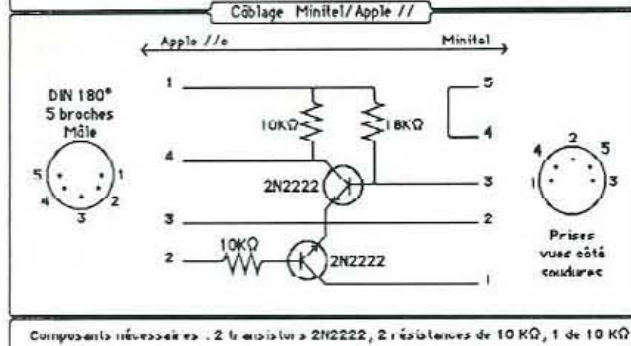
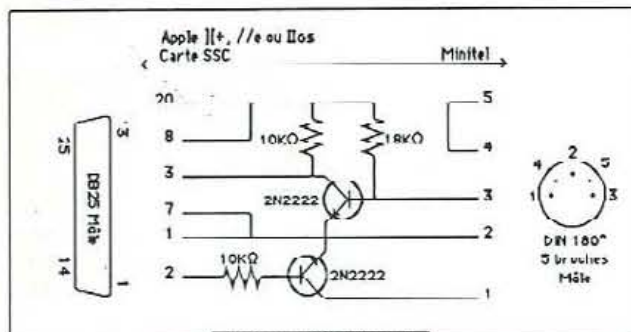
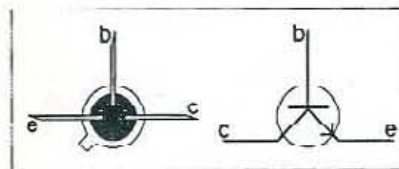
```

**Sauvegarder un fichier**

```

350 IF PEEK(9728) = 255 OR PEEK(9729)
- 255 THEN CALL AD, SL, BE: HOME: PRINT
"RIEN A SAUVEGARDER <RETURN>": WAIT 49

```



Composants révisés en 2: 2 transistors 2N2222, 2 résistances de 10 KΩ, 1 de 18 KΩ

```

152,128: POKE 49168,0:GOTO 100
351 FOR I = 0 TO 2:M$(I) = M2$(I): NEXT :C
H = A1:TT$ = M1$(3): GOSUB 1100:A1 = CH
352 ON A1 = 2 GOTO 100
353 IF A1 < > DK THEN 4200
355 HOME : VTAB 8: PRINT NF$"A SAUVEGARDER
OU " CHR$(34)"?" CHR$(34):GOSUB 4000
: IF NM$ = "" THEN 350
360 IF CT THEN CT = 0: GOTO 355
370 GOSUB 6100
376 IF DR THEN 380
378 GOSUB 4100
380 PRINT D$"BSAVE"NM$,A$2500,L"LF:A0 = 1
:FS = 0: GOTO 100
400 HOME : ON A0 + 1 GOTO 405,450

```

#### Recevoir une séquence

```

405 PRINT : PRINT " ENVOT : RETURN":
PRINT : PRINT " CORRECTION : CTRL-C/DEL
": PRINT " ANNULLATION : CTRL-X": PRINT
" SUITE : CTRL-I/TAB": PRINT " REP
ETITION : CTRL-B
410 PRINT " SOMMAIRE : CTRL-S": PRINT "
GUIDE : CTRL-G": PRINT " RETOUR :
CTRL-R": PRINT " CONNEXION : CTRL-K":
PRINT "FIN DE PAGE : CTRL-E"
412 PRINT "DECONNEXION : CTRL-W": PRINT "
FIN : ESC
415 GOSUD 7000
420 CALL AD,SL,EN: PRINT D$"PR#0": IF PEE
K (9720) < > 255 THEN A1 - 0: GOSUB 41
00:DR = 1:DK = 0:FS = 1: GOTO 355
425 DR - 0
430 GOTO 100

```

#### Consulter une séquence

```

450 ON SL - 6 GOTO 4300: IF A1 < > DK THE
N 4200
452 GOSUD 7000
455 M$(0) = "1200 BAUDS":M$(1) = "4800 BAUD
S":M$(2) = M3$(2):I - 3:CH - BD: COSUB
1100: ON CH = 2 GOTO 150:BD = CH: ON BD
COSUB 4600
460 HOME : PRINT : PRINT "FLECHES : PASSER
DE PAGE EN PAGE": PRINT " ESC
: ABANDONNER LA CONSULTATION
470 CALL AD,SL,LE: PRINT D$"PR#0": ON BD C
OSUB 4610
480 GOTO 100
500 HOME : ON A0 + 1 GOTO 505,550

```

#### Recevoir une copie d'écran

```

505 PRINT : PRINT "MODE COPIE D'ECRAN": PR
INT : PRINT : PRINT "FAIRE <RETURN> AVA
NT CHAQUE COPIE": PRINT "<ESC> POUR FIN
IR
515 GOSUD 7000
520 CALL AD,SL,CE: PRINT D$"PR#0": IF PEE
K (9720) < > 255 THEN A1 - 1: COSUB 41
00:DR = 1:DK = 1:FS = 1: GOTO 355
530 DR - 0: GOTO 100

```

#### Consulter une copie d'écran

```

550 ON A1 < > DK GOTO 4200: FOR I = 0 TO
3:M$(I) = M5$(I): NEXT :CH - A3:TT$ - H
1$(1) + " - " + M2$(1): GOSUB 1100:A3 =
CH
555 ON CH < 2 AND SI = 6 GOTO 4300
560 ON A3 + 1 GOTO 750,750,750,150

```

```

600 FOR I = 0 TO 6:M$(I) = M4$(I): NEXT :C
H = SL - 1:TT$ = M1$(4) + " - " + M3$(A
2): GOSUB 1100: IF CH < 6 THEN SL = CH
+ 1
610 A2 = 2: GOTO 200
700 FOR I = 0 TO 6:M$(I) = M4$(I): NEXT :C
H = SI - 1:TT$ = M1$(4) + " - " + M3$(A
2): GOSUB 1100: IF CH < 6 THEN SI = CH
+ 1
710 A2 - 2: GOTO 200
750 ON A3 + 1 GOTO 775,775,760
760 HOME : VTAB 8: PRINT NF$"TEXT' RECEPT
EUR OU " CHR$(34)"?" CHR$(34): GOSUB
4000: IF NM$ - "" THEN 550
770 IF CT THEN CT = 0: GOTO 160
775 HOME : PRINT : PRINT "TRANSFERT EN COU
RS...": PRINT : PRINT "<ESC> POUR L'INT
ERRROMPRE": IF A3 < > 2 THEN 795
780 PRINT D$"OPEN"NM$: PRINT D$"WRITE"NM$
: GOTO 800
795 PRINT D$"PR#"SI: IF A3 = 0 THEN PRINT
CHR$(27) CHR$(69) CHR$(14)
800 AC = 9729
810 OC = PEEK (AC): IF OC = 255 THEN 900
820 IF OC = 254 THEN PRINT : PRINT : FOR
I = 0 TO 39: PRINT "-": NEXT : PRINT :
PRINT :AC = AC + 1: GOTO 810
822 POKE 2039,OC + 128
825 IF PEEK (49152) = 155 THEN POKE 4916
8,0: GOTO 900
830 PRINT CHR$(OC):AC = AC + 1: GOTO 81
0
900 PRINT D$"CLOSE": IF A3 = 0 THEN PRINT
CHR$(15)
910 PRINT CHR$(12): PRINT D$"PR#0": POKE
2039,160: GOTO 100

```

#### Initialise les tableaux pour les menus

```

1000 DATA MINITEL VERS APPLE,CONSULTATION,
CHARGER,SAUVEGARDER,OPTIONS,ENVOI D'UN
FICHER,FIN
1010 FOR I = 0 TO 6: READ M1$(I): NEXT
1020 DATA SEQUENCES,PORT MINITEL,COPIES D'
ECRAN,PORT IMPRIMANTE,RETOUR,RETOUR
1030 FOR I = 0 TO 2: READ M2$(I),M3$(I): N
EXT
1040 FOR I = 0 TO 4:M4$(I) = "PORT " + ST
RS (I + 1): NEXT :M4$(6) = "RETOUR":M4
$(5) = "NEANT
1050 DATA VERS IMAGEWRITER/DMP,VERS AUTRE
IMPRIMANTE,VERS UN FICHER 'TEXT',RETO
UR
1060 FOR I = 0 TO 3: READ M5$(I): NEXT
1070 DATA 27,58,107,118,32,27,58,107,100
,32
1080 FOR I = 0 TO 9: READ A$(I): NEXT
1090 RETURN

```

#### SP Affichage des menus

```

1100 HOME :TT$ = " " + TT$ + " ": PRINT S
PC( (40 - LEN (TT$)) / 2): INVERSE :
PRINT TT$: NORMAL :MA = I - 1: FOR I =
0 TO MA: VTAB I * 2 + 8: PRINT SPC(
12)M$(I): NEXT :ESC = 0
1105 IF LEFTS (M$(0),1) = "M" AND FS THEN
VTAB 14: HTAB 12: INVERSE : PRINT "
"MS(3) " ": NORMAL.

```

```

1110 GOSUB 1300
1120 GOSUB 6000:CA - PEEK (49152)
1130 IF CA = 13 THEN RETURN
1140 IF CA - 8 OR CA - 11 THEN CH - CH - 1
: IF CH < 0 THEN CH = MA: GOTO 1110
1150 IF CA - 21 OR CA - 10 THEN CH - CH - 1
: IF MA < CH THEN CH = 0: GOTO 1110
1160 IF CA - 27 THEN CH - MA: RETURN
1170 GOTO 1110
1200 RETURN

```

### Affiche la flèche des menus

```

1300 FOR I = 0 TO MA: VTAB I * 2 + 8: POKE
36,7: IF CH = I THEN PRINT "***->";
GOTO 1330
1320 PRINT SPC(4)
1330 NEXT: RETURN
2000 FOR I = 1 TO 5: PRINT "-----": NE
XT: RETURN

```

### Fin. ESC ne permet pas de sortir

```

3000 ON CA = 27 GOTO 100: END

```

### Saisie le nom du fichier

```

4000 IF UK = 1 THEN CT = 1:UK = 0: RETURN
4002 VTAB 10: POKE 36,10: CALL - 958: PR
NT ".....": POKE 36,10
: INPUT "":NMS: IF NMS = "" THEN RETU
RN
4003 IF LEN (NMS) > 20 THEN : CALL AD,SL,
BE: GOTO 4000
4030 IF LEN (NMS) < 3 THEN 4050
4040 IF ( LEFT$ (NMS,2) = "C." OR LEFT$ (
NMS,2) = "S.") AND CU THEN NMS = RIGH
T$ (NMS, LEN (NMS) - 2)
4050 IF ( ASC ( LEFT$ (NMS,1)) < 65 OR AS
C ( LEFT$ (NMS,1)) > 90) AND NM$ < >
"?" THEN CALL AD,SL,BE: GOTO 4000
4055 FOR I = 0 TO 1:M$(I) - "LECTEUR " +
STR$ (I + 1): NEXT :M$(2) = M2$(2):I =
3:CH - PEEK (43624) - 1: GOSUB 1100:
IF CH = 2 THEN CT = 1: RETURN
4056 POKE 43624,CH + 1: HOME
4057 IF NMS = "?" THEN VTAB 10: PRINT D$
"CATALOC": PRINT : PRINT "<RETURN>": CO
SUB 6000: HOME :CT = 1
4060 RETURN
4100 LF = PEEK (8) * 256 + PEEK (7) - 94/
1: RETURN

```

### Erreur de type de fichier

```

4200 HOME : PRINT : PRINT "IL N'Y A PAS DE
"M2$(1 - DK): PRINT "EN MEMOIRE... <R
ETURN>": CALL AD,SL,BE
4210 GOSUB 6000:A1 = DK: GOTO 100

```

### L'une des interfaces manque

```

4300 HOME : PRINT : PRINT "UTILISEZ LE MEN
U 'OPTIONS' POUR": PRINT "INDIQUER LA
CONFIGURATION <RETURN>": CALL AD,SL,BE
:A0 = 4: GOSUB 6000: GOTO 100

```

### Envoi d'un fichier au Minitel

```

4400 HOME : VTAB 8: PRINT NMS'"TEXT' A ENV
OYER":CU - 0: COSUB 4000:CU - 1: IF NM
$ = "" THEN 100
4410 IF CT THEN CT = 0: GOTO 4400
4420 HOME : PRINT : PRINT "TRANSFERT EN CO

```

```

URS...": PRINT : PRINT "<ESC> POUR I.T
NTERRROMPRE

```

```

4430 PRINT D$"PR#"SL: PRINT D$"OPEN"NMS: P
RINT D1$"READ"NM$
4440 GET RS: IF RS = CHR$ (13) THEN POKE
6,19: CALL AD,SL,CX: POKE 6,72: CALL
AD,SL,CX: FOR I = 1 TO 2400: NEXT : GO
TO 4440
4445 FOR I = 1 TO 35: NEXT
4450 POKE 6,ASC (R$): CALL AD,SL,CX: IF
PEEK (49152) = 155 THEN POKE 49168,0
: GOTO 4500
4460 GOTO 4440
4500 PRINT D$"CLOSE": PRINT D1$"PR#0": GOT
O 100

```

### Minitel et interface en 4800 bauds

```

4600 VTAB 17: FOR J = 0 TO 4: POKE 6,A$(J)
: CALL AD,SL,CX: NEXT : PRINT : PRINT
D$"PR#"SL: PRINT " " CHR$ (1)"12B": PR
INT D$"PR#0": VTAB 17: PRINT SPC(4):
PRINT
4605 RETURN

```

### Minitel et interface en 1200 bauds

```

4610 VTAB 17: FOR J = 5 TO 9: POKE 6,A$(J)
: CALL AD,SL,CX: NEXT : PRINT : PRINT
D$"PR#"SL: PRINT " " CHR$ (1)"8B": PRI
NT D$"PR#0": VTAB 17: PRINT SPC(4): P
RINT
4615 RETURN

```

### Traitement des erreurs

```

5000 CALL 9461:U = PEEK (222): IF U = 5 T
HEN 4500
5005 CALL AD,SL,BE:U = PEEK (222): IF U =
255 THEN RESUME
5010 LI = PEEK (219) * 256 + PEEK (218):
PRINT D$"PR#0
5020 VTAB 18: PRINT : IF U = 8 THEN PRINT
"ERREUR D'ENTREE/GORTIE
5030 IF U = 6 THEN PRINT "FICHER INTROUV
ADLE
5035 IF U = 13 THEN PRINT "ERREUR DE TYPE
DE FICHER
5040 IF U = 9 THEN PRINT "PAS DE PLACE SU
R LE DISQUE
5045 PRINT " <RETURN>": WAIT 49152,128: PO
KE 49168,0
5050 IF LI = 280 THEN 255
5060 IF LI = 390 THEN 355
5070 IF LI = 780 THEN 760
5080 IF LI = 1057 THEN UK - 1: GOTO 4000
5090 IF LI = 4430 THEN 4400
5091 END

```

### SP attend une touche

```

6000 WAIT 49152,128: POKE 49168,0: RETURN

```

### Préfixes pour fichiers binaires

```

6100 IF A1 = 0 THEN NMS = "S." + NMS
6110 IF A1 = 1 THEN NMS = "C." + NMS
6120 RETURN

```

### Initialise SSC Minitel

```

7000 PRINT D$"PR#"SL: PRINT D$"PR#0": RETU
RN

```

## Récapitulation 'MINITEL'

Après avoir saisi ce code sous  
moniteur, vous le sauvegarderez par :  
BSAVE MINITEL.A\$204A.L\$4B5

204A:20 F5 E6 8A 48 18  
2050:69 C0 85 FD 68 0A 0A 0A  
2058:0A 69 89 85 EB 85 ED C6  
2060:FD A9 C0 85 EC 85 EE A9  
2068:00 85 FC 20 F5 E6 8A C9  
2070:01 F0 28 C9 02 D0 03 4C  
2078:97 21 C9 03 D0 03 4C 5C  
2080:22 C9 04 D0 03 4C 73 23  
2088:C9 05 D0 03 4C A0 23 60  
2090:20 92 23 A0 23 A9 D8 20  
2098:3A DB 60 20 90 20 20 43  
20A0:23 2C 00 C0 30 3C A2 00  
20A8:A1 FR 29 08 F0 F3 A1 FD  
20B0:A0 00 91 07 09 80 8D F7  
20B8:07 E6 07 D0 1C E6 08 A5  
20C0:08 C9 96 D0 14 C6 07 C6  
20C8:08 20 92 23 A0 24 A9 2D  
20D0:20 3A DB 20 0C FD 4C 07  
20D8:21 E6 1C D0 02 E6 1D 4C  
20E0:A1 20 2C 10 C0 AD 00 C0  
20E8:09 80 85 06 C9 9B D0 25  
20F0:20 92 23 A0 24 A9 53 20  
20F8:3A DB 20 0C FD C9 9D F0  
2100:06 20 90 20 4C A1 20 A0  
2108:00 A9 FF 91 07 C8 C8 C8  
2110:91 18 4C 92 23 A2 00 BD  
2118:A4 24 F0 6B C5 06 F0 06  
2120:E8 E8 E8 E8 D0 F1 8A C9  
2128:2D B0 49 A0 00 A5 1D D0  
2130:20 A5 1C C9 64 B0 1A 71  
2138:18 91 18 90 04 C8 38 B0  
2140:01 C0 A5 1D 71 10 91 10  
2148:20 6B 23 20 64 23 4C 74  
2150:21 18 A5 1C 71 18 48 90  
2158:04 C8 38 B0 01 C8 A5 1D  
2160:71 18 E6 18 E6 18 A0 01  
2168:91 18 20 6B 23 88 68 91  
2170:18 20 64 23 A0 03 FR RD  
2178:A4 24 F0 08 85 06 20 32  
2180:23 88 D0 F2 4C A1 20 20  
2188:32 23 4C A1 20 20 92 23  
2190:A0 23 A9 CB 4C 3A DB 20  
2198:8D 21 20 8A 23 A0 00 84  
21A0:18 84 07 A9 25 85 19 85  
21A8:08 E6 08 D0 0F A0 01 B1  
21B0:18 C9 FF F0 75 85 08 88  
21B8:B1 18 85 07 E6 18 E6 18  
21C0:A0 00 B1 07 30 64 85 06  
21C8:09 80 RD F7 07 20 32 23  
21D0:E6 07 D0 02 E6 08 A0 00  
21D8:A5 07 D1 18 D0 E2 C8 A5  
21E0:08 D1 18 D0 DB AD 00 C0  
21E8:10 FB 2C 10 C0 C9 9B F0  
21F0:4F C9 8A F0 12 C9 88 F0  
21F8:0E C9 95 F0 B0 C9 8B F0  
2200:AC 20 73 23 4C E5 21 38  
2208:A5 18 E9 02 F0 0B 38 E9  
2210:02 85 18 20 8A 23 4C AD  
2218:21 20 73 23 20 92 23 A0  
2220:23 A9 FF 20 3A DB 4C 37

2228:22 60 20 73 23 20 92 23  
2230:A0 24 A9 16 20 3A DB 20  
2238:0C FD 20 80 21 4C E5 21  
2240:20 92 23 A0 24 A9 6R 20  
2248:3A DB 20 0C FD 29 DF C9  
2250:CF F0 06 20 8D 21 4C E5  
2258:21 4C 92 23 20 43 23 4C  
2260:CA 22 20 FD 22 90 07 C9  
2268:1B D0 F7 4C CA 22 A5 06  
2270:C9 13 DU UA AU UC 20 FD  
2278:22 B0 EC 88 D0 F8 A5 06  
2280:C9 0D D0 02 C6 1E A5 06

## 80 colonnes et HGR2

Si votre carte 80 colonnes  
TEXT Apple est active  
(PR#3), l'ordre HGR2  
ne vous conduit pas  
nécessairement à effacer  
et afficher la page  
graphique 2. Le remède :  
avant de faire HGR2,  
faites POKE 49164,0 et  
POKE 49152,0 ce qui  
désactive la carte et  
commute la mémoire  
principale.

2288:C9 0D F0 04 C9 20 90 D2  
2290:A0 00 91 07 09 80 8D F7  
2298:07 E6 07 D0 1C E6 00 A5  
22A0:08 C9 96 D0 14 C6 07 C6  
22A8:08 20 92 23 A0 24 A9 2D  
22B0:20 3A DB 20 0C FD 4C 22  
22B8:23 24 1E 10 0A E6 1E E6  
22C0:1A A5 1A C9 18 F0 03 4C  
22C8:62 22 A5 08 20 6R 23 A0  
22D0:00 A9 FE 91 07 E6 07 D0  
22D8:0A E6 08 A5 08 C9 96 D0  
22E0:02 F0 C2 20 92 23 A0 24  
22E8:A9 89 20 3A DB 20 0C FD  
22F0:C9 9D F0 2E 20 2B 23 20  
22F8:18 23 4C 62 22 2C 00 C0  
2300:30 0E A2 00 A1 EB 29 08  
2308:F0 F3 A1 ED 85 06 18 60  
2310:2C 10 C0 AD 00 C0 38 60  
2318:20 92 23 A0 23 A9 EC 4C  
2320:3A DB A0 00 A9 FF 91 07  
2328:4C 92 23 A9 00 85 1A 85  
2330:1E 60 8A 48 A2 00 A1 EB  
2338:29 10 F0 F8 A5 06 81 ED  
2340:68 AA 60 A9 00 85 07 85  
2348:1A 85 1E 85 18 A9 26 85  
2350:08 85 19 C6 19 AU 00 A9  
2358:00 91 18 A9 26 CR 91 18  
2360:20 64 23 60 A9 00 85 1C  
2368:85 1D 60 C9 90 90 03 20

2370:73 23 60 98 48 8A 48 A0  
2378:40 98 6A AA CA D0 FD 2C  
2380:30 C0 88 DU F4 68 AA 68  
2388:AR 60 A9 0C 85 06 20 32  
2390:23 60 A9 00 85 24 A9 17  
2398:85 25 20 C1 FR 4C 9C FC  
23A0:A5 06 09 80 85 06 A2 00  
23A8:BD A4 24 F0 0A C5 06 F0  
23B0:09 D8 D0 D0 D0 F1 4C  
23B8:32 23 A0 03 E8 BD A4 24  
23C0:F0 08 85 06 20 32 23 88  
23C8:DU F2 60 CD CF C4 C5 A0  
23D0:CC C5 C3 D4 D5 D2 C5 00  
23D8:CD CF C9 C5 AU C5 CE D2  
23E0:C5 C7 C9 D3 D4 D7 C5 CD  
23E8:C5 CE D4 00 CD CF C4 C5  
23F0:A0 C3 CF D0 C9 C5 A0 C4  
23F8:A7 C5 C3 D2 C1 CD 00 D0  
2400:D2 C5 CD C9 C5 D2 C5 A0  
2408:D0 C1 C7 C5 A0 BC D2 C5  
2410:D4 D5 D2 CE BE 00 C4 C5  
2418:D2 CE C9 C5 D2 C5 A0 D0  
2420:C1 C7 C5 AU BC D2 C5 D4  
2428:D5 D2 CE BE 00 CD C5 CD  
2430:CF C9 D2 C5 A0 D3 C1 D4  
2438:D5 D2 C5 C5 AC A0 C5 CE  
2440:D2 C5 C7 C9 D3 D4 D2 C5  
2448:DA AU BC D2 C5 D4 D5 D2  
2450:CE BE 00 D0 C1 D5 D3 C5  
2458:AE A0 BC C5 D3 C3 BE A0  
2460:D0 CF D5 D2 A0 C6 C9 CE  
2468:C9 D2 00 CF CB A0 D0 CF  
2470:D5 D2 A0 C6 C9 CE A0 C4  
2478:C5 A0 C3 CF CE D3 D5 CC  
2480:D4 C1 D4 C9 CF CE A0 BF  
2488:00 D2 C5 D4 D5 D2 CE A0  
2490:BD A0 C3 CF D0 C9 C5 AC  
2498:A0 C5 D3 C3 A0 BD A0 C6  
24A0:C9 CE A0 00 8D 13 41 00  
24A8:8B 1B 39 68 97 13 42 00  
24B0:89 13 48 00 FF 13 47 00  
24B8:83 13 47 00 93 13 46 00  
24C0:87 13 44 00 98 13 45 00  
24C8:92 13 42 00 82 13 43 00  
24D0:85 0D 00 00 C0 19 41 61  
24D8:DC 19 4B 63 FR 19 42 65  
24E0:FD 19 41 65 DB 19 30 00  
24E8:A3 19 23 00 DD 19 27 00  
24F0:FC 19 41 75 00 60 A0 60  
24F8:A6 DF 9A 48 98 48 60

## Récapitulation 'PARAM'

NB : Ce fichier n'est utile que pour  
utilisation sur un Apple IIc.

Après avoir saisi ce code sous  
moniteur, vous le sauvegarderez par :  
BSAVE PARAM.A\$300.L\$27

0300:A9 38 8D 7C 04 A9 6B 8D  
0308:7D 04 A9 61 8D 7E 04 A9  
0310:7C 85 42 85 3C A9 04 85  
0318:43 85 3D A9 7E 85 3E A9  
0320:04 85 3F 38 4C 11 C3



# 1' Apple IIgs

L'objet de ces lignes de présentation n'est pas de faire une analyse exhaustive du nouveau bébé (bien beau) ni de céder à la facilité qui consiste à reprendre ligne à ligne le dossier de presse diffusé par le constructeur. Voici simplement notre premier contact et quelques caractéristiques ; gagnons toutefois que nous vous ferons profiter sans tarder de nos premières lignes de programme 16 bits.

## Le processeur

Empiler directement un registre, compter de 0 à 1000 (ou 65535) directement, disposer de branchements relatifs longs, de sauts indirects à des sous-programmes, utiliser un adressage sur 24 bits, déplacer des blocs de mémoire (32Ko en 80 ms) sont quelques-unes des possibilités du 65816. Ce processeur qui reste dans la ligne des 6502 et 65C02 séduira leurs adeptes : la lecture de ses caractéristiques ne dépaysera pas le Lecteur de Pom's comme il put l'être par l'arrivée du 68000. Le 65816 qui est une évolution, non une révolution, met à la disposition du programmeur :

- Accumulateur A 16 bits,
- Registres X et Y 16 bits,
- Registre DBR, le numéro de la 'page' DATA, c'est à dire les 8 bits de poids fort dans les adressages longs,
- Registre PBR, le numéro de la 'page' de l'instruction suivante (Program Bank Register), le PC contenant les 16 bits de poids faible,
- Registre D pour les adressages directs,
- Registre d'état P,
- Pointeur de pile S sur 16 bits.

Un petit tour en moniteur par le CALL-151 (on est en mode émulation) nous fait découvrir ce nouveau listing :

```
FF/19D8:00 00 E1 E2 30 64 A1 60-. .ab0d!  
FF/19E0:32 D5 DD A0 D5 F3 E5 E4-2U- Used  
FF/19E8:A0 EC E9 F3 F4 A0 A0 D0- list P  
FF/19F0:BD A0 D0 F5 F2 E7 E5 A0=- Purge  
FF/19F8:EC E9 F3 F4 A0 A0 C6 BD-11st F=  
FF/1A00:A0 C6 F2 E5 E5 A0 EC E9- Free li  
FF/1A08:F3 F4 A0 A0 D1 BD A0 D1-st Q- Q  
FF/1A10:F5 E9 F4 3B 8D 8D A3 AU-uit;..#  
FF/1A18:A0 A0 A0 C8 E1 EE E4 EC- Handl  
FF/1A20:E5 A0 A0 C1 E4 E4 F2 A0-e Addr  
FF/1A28:A0 C1 F4 F4 F2 A0 A0 C9- Attr I  
FF/1A30:C4 A0 A0 A0 A0 D3 E9 FA-D S1e  
FF/1A38:E5 AU AU AU DU F2 E5-e Pre  
FF/1A40:F6 A0 A0 A0 CE E5 FR F4-v Next  
FF/1A48:A0 A0 A0 C6 F2 E5 E5 35- Free5
```

A gauche, le numéro de la 'page' sous la forme "nn", on devine rapidement qu'il y a 256 'pages' mais qui n'existent évidemment qu'en fonction de la RAM disponible. A droite un équivalent en ASCII, pratique. Surprise : on est gratifié du bip mérité par FDE4C, et FC58G efface l'écran comme prévu, mais le listing

donne ceci :

```
00/FC58: 00 00 BRK 00  
00/FC5A: 00 00 BRK 00  
00/FC5C: 00 00 BRK 00  
00/FC5E: 00 00 BRK 00  
00/FC60: 00 00 BRK 00  
00/FC62: 00 00 BRK 00  
00/FC64: 00 00 BRK 00  
00/FC66: 00 00 BRK 00  
00/FC68: 00 00 BRK 00  
00/FC6A: 00 00 BRK 00  
00/FC6C: 00 00 BRK 00  
00/FC6E: 00 00 BRK 00
```

Ceci parcequ'au-dessus de SBFFF, en mode émulation, le saut se fait dans la page \$IT, là où se trouve la ROM :

```
FF/FC58: A0 05 LDY #05  
FF/FC5A: 80 CD BRA FC29 (-33)  
FF/FC5C: FB XRA  
FF/FC5D: 4C EB FC JMP FCEB  
FF/FC60: 00 00 BRK 00  
FF/FC62: A9 00 LDA #00  
FF/FC64: 85 24 STA 24  
FF/FC66: E6 25 INC 25  
FF/FC68: A5 25 LDA 25  
FF/FC6A: C5 23 CMP 23  
FF/FC6C: 90 B6 BCC FC24 (-4A)  
FF/FC6E: C6 25 DEC 25  
FF/FC70: A0 06 LDY #06  
FF/FC72: 80 B5 BRA FC29 (-4D)
```

L'exploration de cette ROM de 128Ko laisse paraître un 'tableau de bord' qui fait penser au Macintosh. Cette visite nous conduira également à une table de sauts dans la page \$E1 qui rappelle aussi le sus-cité.

## Ouvert, fermé ?

Le tableau de bord, en plusieurs menus et à l'ergonomie agréable, donne la possibilité de régler le niveau sonore, la fréquence du Bip, la couleur du fond, celle des caractères, celle de la marge de l'écran, la 'nationalité' des caractères affichés (pourquoi les Danois perdraient-ils leur à ?), celle du clavier. Bien sûr, il donne également accès à l'horloge mais il séduit surtout par la gestion de l'affectation des slots. Par défaut, nous trouvons dans ceux-ci, connectés 'soft', les interfaces intégrées suivantes :

SLOT 1	Imprimante
SLOT 2	Communication
SLOT 3	80 colonnes
SLOT 4	Souris
SLOT 5	'Smart slot'
SLOT 6	Lecteur de disquettes
SLOT 7	Libre

Ceci sans carte dans l'appareil. Notons que le slot 7 peut être affecté à l'interface AppleTalk intégrée. Dans ce cas, l'interface communication est indisponible.

Pour chacun des ports, il est possible d'imposer à la machine l'utilisation de l'interface connectée physiquement. Nul besoin de modem ? Mettre en port 2 la carte Z80, l'indiquer par ce tableau de bord et le CP/M est accessible. Le IIGS donne également le choix de l'interface qui servira au 'boot'. Il est même possible de booter sur le disque virtuel si on a pris soin de l'installer, d'y copier une disquette avant le CTRL-C RESET... Vitesse garantie.

Pour les interfaces imprimante et communication, le réglage des paramètres peut également se faire par ce tableau de commande.

Débit, parité, format et protocole de transmission des données sont accessibles sans lever le capot ce qui, vu la conception de l'ouverture, ne nous frustrera pas... Cet aspect de la machine rend plus vivable le 'système ouvert' : Pom's sait par expérience toute la souplesse tirée de multiples cartes mais sait également toute la lassitude de retirer le moniteur pour passer du mode imprimante au mode communication, pour tourner le 'Jumper Bloc' ; parfois, imiter les systèmes fermés a du bon...

## La compatibilité

Ce tableau de bord offre également le choix de la vitesse du processeur : 1 ou 2,8 MHz. Celle des Apple // est à choisir si l'ouverture de Guillaume Tell de la disquette Pom's 5 doit rester compréhensible, ou si votre Z80 est perdu par la vitesse standard du IIGS. La fréquence 2,8 MHz est à choisir pour tous les programmes indépendants du 'timing' ou ceux qui sont synchronisés sur la vidéo. L'utilisation de Big Mac par exemple devient étonnante, le gain est remarquable sur la plupart des programmes surtout si l'on considère la gestion de la carte 80 colonnes, sensiblement plus nerveuse que celle du //c.

La compatibilité semble assurée très largement, bien que notre AppleWriter DOS 3.3 soit inutilisable et que la version ProDOS soit dépassée par les nouvelles interfaces. Même Pascal (dont la portabilité semble surfaite lorsqu'on passe du //e au //c...) nous a paru s'accommoder, mais seule l'expérience en dira plus. Logiquement, les programmes qui cherchent une adresse particulière dans la carte souris ou 80 colonnes courent à l'échec mais l'émulation semble très au point, les disquettes Pom's resteront d'actualité pour celui qui gardera un lecteur 5,25 pouces car le IIGS est équipé en standard de lecteurs de disquettes 3,5 pouces 800K, ceux du Macintosh, un témoin rouge de fonctionnement en prime.

## Quelques caractéristiques du IIGS

Bien des photos des écrans créés à l'aide du programme de dessin GSPaint ont fait les couvertures. Il reste que l'utilisation est spectaculaire, toutes les facilités du Macintosh sont présentes dans ce programme auquel on pardonne d'être plus lent, vues les couleurs disponibles. Le plus étonnant est l'édition de ces couleurs : trois ascenseurs ou potentiomètres pour doser les rouge, vert et bleu, un pour régler la luminosité et un pour la saturation : les 4096 nuances sont là. Ajoutons que les 'motifs' sont éditables en couleurs mais comment obtenir une copie papier qui respecte l'œuvre ?

En résumé, le IIGS, c'est :

- Une unité centrale 128K de ROM et 256 de RAM extensible à 4 mégaoctets,
- Un clavier détachable, ou plutôt détaché, avec bloc numérique et souris,
- Moniteur couleur,
- Lecteur de disquette 3,5 pouces,
- 640 x 200 points à l'écran en 4 couleurs parmi 4096 ou 320 x 200 en 16 couleurs,
- Son 16 voix
- 8 connecteurs d'extension (dont un pour une extension mémoire),
- Deux interfaces série, une souris, une 80 colonnes, une AppleTalk.

Les Lecteurs de Pom's qui se sont penchés sur les sources des programmes Macintosh se sont alloués un avantage certain pour la programmation de leur futur GS. Le contact avec les routines graphiques des 128K de ROM leur sera familier : SetClipRgn, ObscureCursor, GetPenState, TextFace et autre GetFontInfo sont apparues dans le cahier Mac depuis bien des numéros. Les paramètres se passent ici aussi par la pille, pille qui atteint 64K.

## Le graphisme

Point fort du nouvel arrivé, le graphisme Super-Haute Résolution mérite quelques précisions.

L'écran est divisé en 200 lignes de 320 ou 640 points. A chaque ligne est affecté un octet qui en donne les caractéristiques :

- 4 bits pour indiquer quelle table, parmi les 16 tables de 16 couleurs, doit être utilisée,
- 1 bit indiquera si la ligne est en mode 320 ou 640 points,
- 1 pour activer le mode remplissage,
- 1 bit pour les interruptions.

### La table de 16 couleurs...

...choisie pour la ligne est composée de 16 fois deux octets. Dans ces 2 octets, 4 bits donnent le niveau de bleu, 4 pour le vert et 4 pour le rouge, 4 ne servent pas.

### Mode 320 ou 640

En mode 320 chaque pixel est défini non par 1 bit, mais par 4 qui donnent le numéro de la couleur de ce pixel dans la table de la ligne.

En mode 640, chaque pixel est défini par 2 bits donnant le numéro de sa couleur dans une partie de la table de la ligne, il y a donc 4 pixels par octet. Le premier pixel sera de l'une des couleurs 0 à 3, le deuxième dans l'une des couleurs 4 à 7, etc. 2 pixels adjacents ne peuvent donc être de la même couleur que si la table est composée de 4 fois les mêmes couleurs.

### Interruptions

Le tracé d'une ligne peut générer une interruption : il est donc possible d'en profiter pour changer la palette de couleurs avant. Gymnastique complexe qui rend possible l'affichage de 256 couleurs sur le même écran.

### Remplissage

Si le bit de remplissage de l'octet de contrôle d'une ligne est à 1, la couleur 0 de la table de couleur est indisponible : les pixels définis de la couleur 0 seront en fait de la dernière couleur utilisée...

De telles possibilités graphiques devraient donner l'idée d'utilitaires...

Lassé des saisies au clavier ? Choisissez une interface directe et conviviale : le crayon optique, un peu délaissé depuis l'arrivée des Macintosh, Apple //c et leur cortège de souris.

Le petit montage proposé ici ne saurait rivaliser avec la précision des petits rongeurs à roulettes, mais pour passer de menus en menus, il s'avère particulièrement efficace. De plus, le coût de l'opération (moins de 40,00 F) la rend attrayante.

## Le principe :

//e ≠ //c

Les principes retenus pour le contrôle du balayage vidéo étant différents sur les //c et //e, la routine est composée de deux parties indépendantes.

Le manuel de référence de l'Apple //c signale un octet qui change de valeur au début et à la fin du balayage vidéo : \$C019. Pour connaître la position d'un crayon optique sur l'écran, il convient de 'chronométrer' le temps écoulé entre le début du balayage et la réception de la lumière par le crayon. Cette réception sera contrôlée par un phototransistor relié au connecteur des poignées de jeu.

Le programme en assembleur respecte les étapes suivantes :

- attendre que le crayon ne soit plus éclairé pour laisser finir un éventuel balayage en cours,
- attendre le début d'une trame,
- commencer alors le comptage,
- attendre que le phototransistor soit éclairé pour stopper le comptage,
- recommencer le processus si on arrive en fin de trame sans que le crayon n'ait reçu de lumière,
- contrôler que le crayon se retrouve bien dans l'ombre après le passage du spot. Si tel n'est pas le cas, le crayon était éclairé par la lumière ambiante,
- contrôler simultanément le clavier pour que l'utilisateur puisse l'utiliser à son gré.

# Um Crayon Optique

Christian Piard

## MENU PRINCIPAL

- |   |                         |
|---|-------------------------|
| 1 | INITIALISATION          |
| 2 | CREATION                |
| 3 | MODIFICATION            |
| 4 | ANNULATION              |
| 5 | CONSULTATION            |
| 6 | IMPRESSION              |
| 7 | FIN DE LA DEMONSTRATION |

Sur le //c, on en peut plus se référer au changement de valeur de l'octet \$C019. Une interruption peut être générée à chaque début de trame, mission confiée à la carte souris. La routine installe en premier lieu le vecteur d'interruption puis exécute une boucle en attendant la première d'entre-elles (une à chaque 60<sup>ème</sup> de seconde). Le reste du traitement est similaire.

Un test permet au début de la routine de savoir si on se trouve sur l'un ou l'autre des ordinateurs.

## Utilisation du programme

Charger depuis votre programme Basic la routine :

```
PRINT CHR$(4) "BLOAD CRAYON,A3812 7"
```

À chaque utilisation du crayon, faire :

```
CALL 38127
```

Sur le //e, la routine étant relogeable, il suffit de faire :

```
CALL adresse de chargement
```

Sur le //c il faudra réassembler à l'adresse choisie ou, sous ProDOS, utiliser le relogeur d'A. Avrane.

Après cet appel, dans l'octet 6, vous disposez de la position verticale du crayon, entre 0 et 23. Si cet octet est à 255, l'utilisateur s'est servi du clavier et vous trouvez le code de la touche dans l'octet 7. Ceci est illustré dans le programme DEMO.

Comme dans le programme de démonstration, réservez une zone en vidéo inverse sur chaque option pour que le phototransistor réagisse efficacement.

Ce programme peut fonctionner sous ProDOS sur l'un et l'autre appareil.

## Réalisation pratique

Le hardware (!!) se résume à :

- trois résistances qui laissent l'entrée logique 0 (équivalent à la 0) à 0,
- un phototransistor qui, en se saturant à la lumière porte cette entrée à 1.

Le phototransistor - dont le type n'est pas critique - sera installé dans un tube opaque. Quant aux résistances, elles trouveront leur place dans la prise reliée au connecteur poignée de jeu.

L'incompatibilité se faisant sentir à ce niveau là également, l'une des résistances est ajustable car le seuil de déclenchement de l'entrée diffère sensiblement selon les appareils. Pour la régler, exécutez le programme suivant :

```
10 PRINT PEEK (49249)
20 GOTO 10
```

Réglez-la pour obtenir les valeurs supérieures à 127 (généralement on obtient 160) puis revenez en arrière légèrement pour obtenir des valeurs inférieures ou égales à 127 (généralement 32), le tout, crayon à l'ombre.



## Récapitulation 'CRAYON'

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par :  
BSAVE CHAYON, A\$94EF, L\$111

```
94EF- AD
94F0- C0 FB F0 4A 2C 00 C0 30
94F8- 38 2C 61 C0 10 F6 A9 00
9500- 85 06 85 07 2C 19 C0 30
9508- FB 2C 19 C0 10 FB A2 1E
9510- 2C 00 C0 30 1C 2C 61 C0
9518 30 0C CA D0 F3 E6 06 2C
9520- 19 C0 30 EA 10 CE A2 20
9528- CA D0 FD 2C 61 C0 30 C4
9530- 60 2C 10 C0 AD 00 C0 65
9538- 07 A9 FF 85 06 60 78 A9
9540- 68 8D FE 03 A9 95 8D FF
9548- 03 A0 19 20 DE 95 A9 08
9550- A0 12 20 DE 95 A9 00 85
```

```
9558- 08 58 24 00 10 FC 78 A9
9560- 00 A0 12 20 DE 95 58 60
9568 78 A5 FE 85 19 A5 FF 85
9570- 1A A5 45 85 1B 86 1C 84
9578- 1D A9 00 A0 12 20 DE 95
9580- A0 13 20 DE 95 90 06 20
9588- F1 95 4C FB BF A9 00 85
9590- 06 85 07 85 08 A0 06 A2
9598- E3 CA D0 FD 88 D0 F8 A2
95A0- 1E 2C 00 C0 30 2A 2C 61
95A8- C0 30 17 CA D0 F3 E6 06
95B0- A5 06 C9 18 90 E9 A9 08
95B8- A0 12 20 DE 95 20 F1 95
95C0- 58 40 A2 20 CA 00 FD 2C
95C8- 61 C0 30 EA C6 08 30 E6
95D0- 2C 10 C0 AD 00 C0 85 07
95D8- A9 FF 85 06 D0 EE 40 A2
95E0- 00 86 FE A2 C4 86 FF B1
95E8- FE 85 FE A0 40 68 6C FE
95F0- 00 A5 19 85 FE A5 1A 85
95F8- FF A4 1D A6 1C A5 1B 60
```

## Source 'T.CRAYON'

### Assembleur Big Mac

1	ORG	\$9600-273		43	INC	COMPT	Chronomètre	
2	COMPT	=	\$6	44	BIT	VBL	jusqu'à fin	
3	TOUCHE	=	\$7	45	BMI	S0E	balayage	
4	DRAP	=	\$8	46	BPL	DEB		
5	SAVE	=	\$19	47				
6	SAVA	=	\$1B	48	LDX	#\$20	Crayon de nouveau	
7	SAVX	=	\$1C	49	DEX		à l'ombre ?	
8	SAVY	=	\$1D	50	BNE	ECE		
9	ACC	=	\$45	51	BIT	CRAYON		
10	ADR	=	\$FE	52	BMI	DEB	Non = parasite	
11	VECT	=	\$3FE	53	RTS		Oui = fini	
12	VBL	=	\$C019	54				
13	CRAYON	=	\$C061	55	CLAVE	BIT	\$C010	Vide clavier
14	KBD	=	\$C000	56	LDA	KBD		Sauvegarde caractère et mot
15				57	STA	TOUCHE		compteur à \$FF
16				58	LDA	#\$FF		
17	LDA	\$FBC0	//c ou //e ?	59	STA	COMPT		
18	BEQ	IIC		60	RTS			
19	DIT	KBD	On attend une touche	61				
20	BMI	CLAVE	au clavier	62	IIC	SET		Mise en place
21	BIT	CRAYON	ou que le crayon	63	LDA	#<INTER		du vecteur
22	DPL	DEB	soit à l'ombre	64	STA	VECT		d'interruption
23				65	LDA	#>INTER		
24	LDA	#0	Initialise	66	STA	VECT+1		
25	STA	COMPT	compteurs	67				
26	STA	TOUCHE		68	LDY	#\$19		Initialise souris
27				69	JSR	SOURIS		
28	E1E	BIT	VBL	70	LDA	#8		En mode interrupt
29		BMI	E1E	71	LDY	#\$12		vidéo autorisées
30	E2E	BIT	VBL	72	JSR	SOURIS		
31		BPL	E2E	73	LDA	#0		Initialise drapeau
32				74	STA	DRAP		
33	S0E			75	CLI			
34	LDX	#\$1E	temporisation	76	BCL			
35	S00E			77	BIT	DRAP		On boucle en
36	BIT	KBD	Attend une touche	78	BPL	BCL		attendant interrupt
37	BMI	CLAVE	au clavier	79	SEI			
38	BIT	CRAYON	ou que le crayon	80	LDA	#0		Remet souris en
39	BMI	ECLAIRE	soit éclairé	81	LDY	#\$12		inactif
40	DEX			82	JSR	SOURIS		
41	BNE	S00E	boucle	83	CLI			
42				84	RTS			
				85				
				86	INTER			
				87	SEI			
				88	LDA	ADR		Sauvegarde des

```

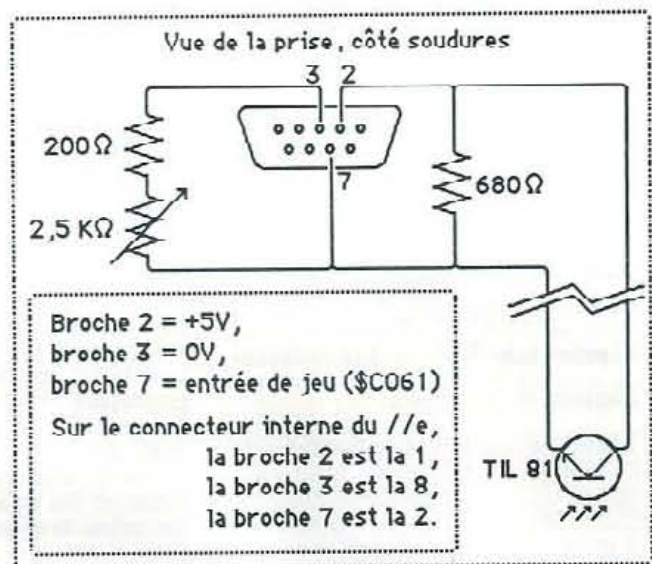
89      STA  SAVE          registres
90      LDA  ADR:1
91      STA  SAVE+1
92      LDA  ACC
93      STA  SAVA
94      STX  SAVX
95      STY  SAVY
96      LDA  #0
97      LDY  #$12          Souris inactive
98      JSR  SOURIS
99      LDY  #$13          Interr pour nous ?
100     JSR  SOURIS
101     BCC  SUITE          Oui = on la traite
102     JSR  RESTREG       Non - on restaure
103     JMP  $BFEB         et on repart
104 SUITE LDA  #0          Initialise
105     STA  COMPT         les compteurs
106     STA  TOUCHE
107     STA  DRAP
108     LDY  #6
109 RN   LDX  #$E3          Attend que le
110 B1   DEX                      premier point
111     BNE  B1              de l'écran
112     DEY                      soit allumé
113     BNE  B0
114 S0   LDX  #$1E          Temporisation
115 S00
116     BIT  KBD
117     BMI  CLAV           surveillance clavier
118     BIT  CRAYON        et crayon
119     BMI  ECLAIR
120     DEX
121     BNE  S00
122
123     INC  COMPT          On compte de 0
124     LDA  COMPT          à 23 maximum
125     CMP  #24
126     BCC  S0
127 S000
128     LDA  #8
129     LDY  #$12          Souris active pour
130     JSR  SOURIS        nouvelle interrupt
131     JSR  RESTREG
132     CLI
133     RTI
134
135 ECLAIR
136     LDX  #$20          Crayon de nouveau
137 EC   DEX                      à l'ombre ?
138     DNE  EC
139     BIT  CRAYON
140     RMI  S000
141 FIN  DEC  DRAP
142     BMI  S000
143
144 CLAV BIT  $C010        Garde caractère
145     LDA  KBD
146     STA  TOUCHE
147     LDA  #$FF          et compt à $FF
148     STA  COMPT
149     BNE  FIN
150 SOURIS
151     PHA
152     LDX  #0            Cherche dans la
153     STX  ADR           table en $C400
154     LDX  #$C4          l'adresse réelle de
155     STX  ADR+1        la routine désirée
156     LDA  (ADR),Y      puis on
157     STA  ADR           y fait un saut
158     LDY  #$40
159     PLA
160     JMP  (ADR)

```

```

161 RESTREG
162     LDA  SAVE          Restaure les
163     STA  ADR           registres
164     LDA  SAVE+1
165     STA  ADR+1
166     LDY  SAVY
167     LDX  SAVX
168     LDA  SAVA
169     RTS

```



### Programme 'DEMO'

```

5 PRINT CHR$(4) "BLOADCRAYON,A38127"
6 HIMEM: 37000: REM ProDOS page entière
10 DATA INITIALISATION, CREATION, MODIFICATION, ANNULATION, CONSULTATION, IMPRESSION, FIN DE LA DEMONSTRATION
11 FOR I = 1 TO 7: READ M$(I): NEXT
90 REM -----
100 HOME: PRINT "MENU PRINCIPAL"
110 FOR I = 1 TO 7: VTAB 2 * I + 4: PRINT
    SPC(3); INVERSE: PRINT " I ";
    NORMAL: PRINT " M$(I): NEXT
115 REM -----
120 CALL 38127
130 CHOIX = (PEEK(6) - 3) / 2
140 IF PEEK(6) = 255 THEN CHOIX = PEEK(7) - 48
150 IF CHOIX < 1 OR CHOIX > 7 THEN 120
160 ON CHOIX GOTO 200,200,200,200,200,200,300
200 HOME: PRINT " OPTION "M$(CHOIX) " ";
    VTAB 20: INVERSE: PRINT " RETOUR "; NORMAL
210 CALL 38127: GOTO 100
300 VTAB 20: PRINT "OK POUR FINIR ? "; INVERSE: PRINT " OUI "; VTAB 22: POKE
    36,16: PRINT " NON "; NORMAL
310 CALL 38127: ON PEEK(6) - 19 OR PEEK(7) = ASC("O") GOTO 320
315 IF PEEK(6) < > 21 AND PEEK(7) < > ASC("N") THEN 310
316 VTAB 20: CALL 64578: GOTO 120
320 HOME: END

```

# Micro-informations

Jean-Michel Gourévitch

**Bouleversement dans le paysage de la micro. Alors que les résultats d'IBM s'effritent, voici que ceux d'Apple connaissent un ciel de plus en plus bleu. Il se serait même vendu en août plus de Macintosh que d'IBM PC-AT. Le Macintosh Plus a incontestablement dopé Apple.**

Tandis que se profilent les livraisons de l'Apple IIGS, et que les Macintosh ouverts connaissent semble-t-il quelques retards (on ne parle plus de sortie avant le mois de mars), les Macintosh actuels se vendent particulièrement bien. Aux États-Unis, il faut dire que la baisse de prix (baisse de 200 dollars sur le 512Ko et de 400 dollars sur le Plus) n'est sûrement pas étrangère à ces succès. En France, où les baisses de prix sont rarement à l'ordre du jour chez Apple, on compte déjà 55000 Macintosh vendus, dont 30% dans les "grands comptes" qui boudaient ostensiblement le Macintosh depuis ses débuts. À ces bons résultats, probablement trois raisons : d'abord l'arrivée du Macintosh Plus avec son interface SCSI, qui fait plus "sérieux" que le Macintosh 512Ko sans sortie pour disque dur ; ensuite, le rapprochement de l'univers MS-Dos, qu'il s'agisse de l'intégration du Macintosh dans des réseaux où se trouvent déjà des PC (par exemple avec le réseau TOPS) ou de l'échange de fichiers entre les Macintosh et les PC. Et, enfin, le "boom" de l'édition personnelle. Ce que les Américains appellent le "desktop publishing" et qu'on a baptisé chez nous du nom barbare d'édition ou de PAO (publication assistée par ordinateur). Un grand nombre d'entreprises se sont avisées qu'elles pouvaient réaliser elles-mêmes, et dans leurs locaux, des documents qu'on devait auparavant confier à un imprimeur. Il vient ainsi de s'ouvrir un tout nouveau marché qu'on n'avait pas imaginé auparavant, et qui s'avère être "juteux". IBM, qui s'y intéresse de près, a donc annoncé une version spéciale de son micro scientifique : le PC-RT, spécialisé dans les tâches d'édition. En attendant, Apple dispose avec le Macintosh d'une confortable avance sur ce marché-là.

## PAO à gogo

Rien d'étonnant si les programmes d'édition électronique se multiplient. Aux "Page Maker" et autres "MacPublisher" (dont il vient de sortir une nouvelle version), il faut désormais ajouter Graphic Works de Mindscape, un véritable programme intégré comprenant un traitement de texte, un logiciel de dessins sophistiqué avec aérographe à pressions réglables, et la possibilité de mélanger textes et graphiques. Prix : 80 dollars. Ragtime d'Orange Micro, autre intégré d'édition nanti d'un traitement de texte avec colonnes multiples, possibilité de mélanger des styles et tailles, de créer des cadres de styles différents. On peut aussi "exploser" un document pour accéder simultanément à neuf sections de ce document. En prime, on dispose d'un tableur. Enfin, PS Compose de PS Publishing à usage professionnel permet de choisir des niveaux de gris et de transcrire un texte en langage de composition. Il fonctionne avec des imprimantes utilisant le langage Postscript comme la LaserWriter, (Prix 1000 dollars) mais aussi avec des systèmes de composition de Compugraphic ou d'Allied Linotype (le prix est alors de 2000 dollars).

Est-ce un traitement de texte ultra-sophistiqué ou bien un système de mise en page ? Les nombreuses fonctions de Writer Plus d'ACI permettent de l'utiliser indifféremment pour ces usages. À remarquer parmi les avantages : la possibilité de stocker par thèmes des paragraphes souvent utilisés et de les rappeler, la faculté de stocker par thèmes des images, la permutation automatique de deux caractères (pratique pour corriger une faute de frappe), la césure automatique qui divise un mot sur une syllabe et pose un tiret pour tout mot arrivant en fin de ligne, des statistiques fournissant le nombre de signes, de mots, de paragraphes et d'images d'un texte ou d'une partie de texte, le cadrage d'un texte, l'équilibrage des colonnes, la gestion des niveaux d'un texte avec génération automatique d'un sommaire, le calcul entre plusieurs cellules dans un texte, la numérotation des notes, etc. C'est vrai qu'il est difficile de construire un programme pour le Macintosh. Et, apparemment, plus le programme est inventif, plus le "debugging" est difficile. Au début

novembre, Writer Plus était ainsi encore affligé de plusieurs "bugs", qui auront certainement disparu dans la version qui sera mise en vente.

En attendant, la concurrence s'annonce déjà difficile. Avec notamment Write Now, développé par Solaster Inc (société acquise par Steve Jobs, le fondateur d'Apple) et distribué outre Atlantique par T/Maker. Write Now comporte lui aussi la possibilité d'écrire en colonnes (jusqu'à 4 par page) de contrôler l'écart intercolonnes en points. On peut aussi y mélanger textes et graphiques, changer les dimensions d'une image sans qu'elle se distorde, contrôler l'interlignage de façon précise. Le traitement de texte est évolué, avec notamment une possibilité d'ajouter des notes avec numérotation automatique. Ce traitement de texte très rapide utilise bien évidemment les fichiers de Macwrite ou de Word et comprend dans sa version américaine un correcteur orthographique. Il sera vendu 175 dollars.

## Mini-Macintosh, maxi-prix

Il ne manque plus qu'un Macintosh portable pour compléter la gamme. Certains constructeurs essaient de renouveler le coup fait par Compaq à IBM en présentant le premier PC portable. Cette fois, c'est plus dur car, compte tenu du fait qu'Apple reste l'unique source de production des ROM et du système du Macintosh, il faut obligatoirement passer sous ses fourches caudines, ou se borner à acheter des Macintosh, puis à les recarrosser, ce qui fait allègrement grimper la facture. Il y a quand même des volontaires.

Parmi eux, Colby, qui avait déjà présenté de nombreux Macintosh recarrossés, get espère présenter un portable comprenant outre les éléments d'un Macintosh Plus avec lecteur de 800K, un écran plat au plasma de 12 pouces, un disque dur intégré de 20 Mégas et la possibilité de fonctionner sur batteries. Prix de 5000 à 7000 dollars. À remarquer encore le Dynamac de Dynamac Computers, un portable style Papman lui aussi avec disque dur incorporé et écran plat électroluminescent. Prix de 4500 à 5000 dollars. Enfin, le MX Plus d'Intelitec, à écran électroluminescent, de la taille de l'écran

du Macintosh, avec modem et disque dur incorporé, possibilité de fonctionnement sur 12 volts. Prix : 5 à 7000 dollars, pour un ordinateur installé dans un attaché case tout aluminium. Devant tous ces projets, une seule question : Apple sous-traitera-t-il son Macintosh portable pour le sortir plus vite ? Ou bien tous ces fournisseurs devront-ils rengainer leurs projets pour laisser à la firme à la pomme le monopole jusqu'à présent inentamé de la construction du Macintosh ? Réponse probablement l'année prochaine.

## Logiciels : encore du génie

Lorsqu'on écrira un jour l'histoire de la micro-informatique, il faudra à coup sûr créditer le Macintosh d'un pas important réalisé dans l'intelligence des programmes. Quelques preuves ? D'abord More, la dernière version du traitement d'idées Think Tank. Ce logiciel permettant de générer automatiquement un plan, un organigramme, ou des encadrés pour présentation audiovisuelle est sans concurrence pour préparer un article, concevoir un livre, ou penser à une conférence. Ses multiples perfectionnements (possibilité de regrouper, de promouvoir ou de rétrograder les parties d'un texte, de datage ou d'horodatage, puis de présentation) ont bien mérité de la cause de la micro-informatique. C'est simplement génial.

Autre logiciel illuminé : VIP, distribué par P-Ingénierie. Il s'agit simplement d'un langage de programmation graphique. Plus besoins d'écrire des "GOTO" ou des tristouncts "IF THEN ELSE" : on clique simplement sur des icônes, ou on utilise des menus déroulants. On peut ainsi appeler directement les procédures de la "Toolbox" : ces morceaux de programmes inscrits dans les ROMs du Macintosh permettant, par exemple, d'ouvrir une fenêtre à l'écran. Une fois composé, le programme est représenté par un schéma affiché à l'écran. Le programme peut être visualisé, modifié, exécuté en mode normal ou en pas à pas. Une version "pro" permettant de gérer des variables locales ou le "list manager" sera bientôt disponible. Tout comme des bibliothèques délivrant du code source en "C" ou en Pascal.

Est-ce un système de gestion de bases de données ? Oui, d'une certaine façon. Est-ce un logiciel de traitement d'idées ? Oui aussi. Guide d'OWL International se présente comme "le premier système d'hypertexte" ce qui ne veut strictement rien dire. Plus simplement, ce logiciel permet d'établir des textes ou des graphiques et de créer entre eux des systèmes de références croisées. En cliquant sur une partie d'un dessin, ou voit ainsi s'ouvrir à l'écran le texte ou la

partie de dessin correspondante. Gageons qu'on entendra reparler de ce logiciel vendu 100 dollars.

Et puisqu'Unix est le système à la mode, voici déjà aux États-Unis Macnix distribué par Eurosoft International, qui transforme un Macintosh en station de travail Unix, en le reliant à un ordinateur "hôte" fonctionnant sous ce système. Avantage : Unix fonctionne alors avec tous les avantages graphiques du Macintosh (fenêtres, icônes, possibilité de recevoir un fichier en transférant l'icône de la fenêtre Unix sur la fenêtre du Macintosh, etc.). À remarquer que ce logiciel, qui pourrait constituer une étape importante dans la stratégie logicielle d'Apple, travaille avec les versions Unix System V et 4.2 BSD, ce qui lui permet d'être relié à des stations de travail Sun, Sun 2 et Sun 3, à des Vax de Dec, à des PC-AT et compatibles. Prix : de 600 dollars (version pour 2 utilisateurs) à 10000 dollars (nombre d'utilisateurs illimité).

## Logiciels : des perfectionnements

À remarquer que la dernière version du programme de dessin Full Paint, le successeur de MacPaint, permet d'imprimer en couleurs sur l'imprimante Image writer II. À noter aussi un nouveau langage de programmation en C : le Lightspeed C de Think Technologies. Ce langage particulièrement rapide comprend un éditeur de texte multi-fenêtres, un compilateur, un éditeur de liens et des utilitaires de programmation. Prix : 175 dollars.

Pour les architectes, Space Edit, écrit par l'auteur de Macintosh Space et distribué par Abvent, est un logiciel français de CAO avec objets volumiques combinables, bibliothèques de composants en trois dimensions, perspectives faces cachées, calques superposables, etc. Prix : 7000 Francs. Toujours distribué par Abvent, Slide 123 permet la réalisation sur le Macintosh de dossiers de permis de construire de maisons individuelles, avec notamment l'établissement des calculs de surfaces. La sortie imprimante est réalisée à l'échelle requise par l'administration : 1cm par mètre pour les plans et façades et le plan Masse. Prix : 7000 Francs.

## Des Macintosh dans les labos

Avec l'entrée possible du Macintosh dans les usines (décrite ici dans le dernier numéro de Pom's), son irruption dans les laboratoires n'a pas fini de faire du bruit. Dans un article consacré aux logiciels spéciaux, la revue MacWorld de septembre recense notamment des programmes permettant de reconstruire à

l'écran les molécules d'ADN recombinant : c'est DNA Inspector de Textco. ChemDraw de Stewart Rubinstein est un outil de dessin pour chimistes. Labview de National Instruments Corporation permet de relier

le Macintosh à des instruments et de les contrôler à l'écran du Macintosh : l'interface se réalise par un bus IEEE 488. Macintosh ADIOS de G W Instruments permet de faire fonctionner le Macintosh en analyseur spectrographique, voltmètres à huit canaux, ou d'émuler un oscilloscope à basse fréquence. À remarquer toujours dans le même domaine : Designscope de Brain Power, un instrument de simulation de circuits électroniques, à utiliser avant de construire matériellement un circuit. D'autre part, Stella, un outil pour améliorer les processus de productivité permet aux chercheurs de construire des modèles logiques pour réaliser des simulations. Qu'il s'agisse d'optimiser des quotas de lignes de production ou d'étudier les effets écologiques de la pollution par les gaz d'échappements des automobiles. Par ses possibilités graphiques et de calculs, le Macintosh devrait bientôt pénétrer en force dans les laboratoires. Une irruption facilitée par l'entrée dans la vie active de nombreux étudiants qui l'utilisent depuis déjà plusieurs années.

## Hard

Le génie souffle aussi sur le matériel. Ainsi, voici (hélas seulement pour l'instant aux États-Unis, et en 110 volts, mais on peut rêver), un système permettant de contrôler avec l'ordinateur tous les interrupteurs et tous les systèmes électriques d'une maison. De quoi, par exemple, faire couler automatiquement un bain chaud à heure fixe, réchauffer une pièce juste avant le retour du bureau, allumer ou éteindre automatiquement l'éclairage. Oui, mais je vous vois venir, à quoi bon, allez-vous me dire mobiliser un micro-ordinateur du prix d'un Macintosh pour servir de programmeur électronique ? Eh bien la nouveauté du système X 10 Powerhouse de X 10, c'est que le Macintosh sert simplement à programmer un petit boîtier de commande. On commence par réaliser un plan de sa maison avec les circuits électriques sur un logiciel de style MacPaint. On relie le boîtier au Macintosh, on programme les heures de mise en action en cliquant sur les interrupteurs sur le plan du Macintosh, et en répondant aux questions dans les fenêtres qui s'ouvrent. On peut ensuite enlever le boîtier et le connecter au système électrique. À l'heure dite, les interrupteurs électriques seront commandés. Le module de commande avec câble de liaison au Macintosh et logiciel coûte 80 dollars. Les modules

nécessaires pour commander chaque appareil électrique 20 dollars chacun.

Pour ceux qui ne disposent que d'une Imagewriter I, voici un introducteur feuille à feuille permettant d'éviter le papier perforé. Il coûte moins de 2000 Francs et est vendu par Eudimed.

## Disques durs SCSI

A noter d'abord que le premier Hyperdrive externe (le FX 20) est disponible chez P-Ingénierie pour 15300 F avec ses logiciels de "spool" imprimante, de protection des fichiers et de sauvegarde incrémentale sur disquette. Chez International Computer, on trouve pour quelques 12000 Francs l'IC 20, lui aussi d'une capacité de 20 Mégas. Les lecteurs de Pom's se souviendront peut-être de nos prévisions sur la baisse de prix des disques durs. Voici, pour eux, le premier disque dur externe SCSI de 20 Mégas disponible pour moins de 600 dollars. C'est celui vendu par Jasmine Computer. Combien de temps faudra-t-il pour arriver à ce prix en France ? Moins d'un an ?

## L'Apple // en vedette

L'Apple // est en vedette américaine. Avec, d'abord le IIGS qui est enfin sorti avec une moisson de programmes nouveaux pour la gamme Apple //. Son prix et ses possibilités devraient le réserver dans un premier temps aux marchés de l'éducation. Des traitements de texte ? Voici Multiscribe de Styleware, très ressemblant à Macwrite avec menus et polices de caractères modifiables, utilisation de la souris, et un prix de 60 dollars. Voici encore Paperclip de Batteries Included, avec toutes les fonctions de recherches, les en-têtes et bas de pages automatiques, une vision sur l'écran du document tel qu'il sera imprimé. Voici un nouveau tableur style Macintosh utilisant la souris : VIP professional de ISD Marketing avec 8192 rangs sur 256 colonnes, une mémoire utilisable de 4 Mégas, un grapheur et un langage de macros comme Lotus 1 2 3. Prix : 250 dollars. Voici encore la dernière version d'un simulateur de vol : Jet, réalisé par Sublogic l'auteur du célèbre Flight Simulator qui vous place au commandement d'un avion à réaction, avec vue de l'extérieur, décollage d'un porte avion, etc. Et si on se plante ? Alors, voici Wilderness d'Electric Transit. «Que faire après que vous vous soyez crashé avec le Flight simulator ?» demande la publicité. Utiliser Wilderness : un jeu de survie dans la jungle.

## Améliorer le //

Faut-il abandonner son // ? Il sera

probablement possible de le mettre au niveau du GS. Un kit sera disponible l'an prochain, mais pour un prix d'environ 1000 dollars. C'est que ce kit comprend une nouvelle carte logique, 256Ko de RAM. Le clavier et la souris sont vendus en supplément (pour ne pas nuire à ceux qui avaient déjà acheté une souris pour leur //e).

En attendant, voici bizarrement que sortent de nouveaux accessoires pour améliorer le // dans les domaines où le GS prime. Ce sont notamment la carte Yess de Yam Educational, permettant de transformer le // en piano électronique, et le Phasor, d'Applied Engineering, une carte synthétiseur avec 4 voies stéréophoniques, 12 canaux simultanés, 4 générateurs de bruit blanc, etc. Prix : 179 dollars.

## Déjà des nouveautés pour le IIGs

A peine est-il sorti que voici déjà des logiciels et du matériel pour le GS. Côté logiciels, remarquons les trois programmes de Version Soft : un traitement de texte : GS Write, très semblable à Macwrite, mais avec la possibilité d'ouvrir jusqu'à 16 documents, un programme de dessin GS Paint (MacPaint avec de la couleur !) et un programme de communication : GS Com. Ces produits ont été achetés par la firme Activision qui compte en écarter 75000 aux États-Unis. On recense encore des programmes de publication assistée par ordinateur : Opus de Quark, Pager de Megahaus, et Front Page de Broderbund. TML a un compilateur Pascal, Electronics Art a adapté Deluxe Paint, son programme de dessin déjà vendu pour l'Amiga. Broderbund va améliorer pour le GS les programmes Print Shop et Fantavision. On parle même d'un logiciel intégrateur façon switcher baptisé Carrousel.

Sur le front des accessoires, on remarque un disque dur de 20 Mégas et une carte de 1 méga-octet de disque virtuel d'AST. Une carte d'extension mémoire pouvant contenir 4 Mégas, le RamPak4 GS d'Orange Micro et une carte interface parallèle : le ProGrappler. Orange Micro propose même un des ces accessoires avec un ventilateur et prises supplémentaires qui permettent de rendre bruyants des ordinateurs généralement silencieux. Applied Engineering a déjà concocté deux cartes d'extension : la GS Ram (1,5 Méga octets sur la carte) et la GS Ram Plus (jusqu'à 6 Mégas).

L'Apple // GS est ouvert. Grand ouvert. On va voir déferler les accessoires. Les imaginations et les fers à souder n'ont pas fini de chauffer dans la Silicon Valley.

## Adresses

Mindscape Inc - P.O. Box 1167  
Northbrook - IL 60065

Orange Micro - 1400 n Lakeview Ave  
Anaheim - CA 92807

PS Publishing - 290 Green St San  
Francisco - CA 94133

ACI - 6, avenue Franklin Roosevelt  
75008 Paris - Tél. : 43 59 89 55

T/Maker - 1973 Landings Drive  
Mountain View - CA 94043

P-Ingénierie - 226, bd Raspail  
75014 Paris - Tél. : 43 21 93 36

OWL International - 14128 NE 21st  
Street Bellevue - WA 98007

Think Technologies - 420 Bedford St  
Lexington - MA 02173

Abvent - 53, avenue de Breteuil  
75007 Paris - Tél. : 47 34 43 98

Texten - 27 Gilson Rd West Lebanon  
NH 03784

Stewart Rubinstein - 77 Sacramento St  
n° 11 - Somerville MA 02143

National Instruments - 12109  
Technology Bd Austin - TX 78727

GW Instruments - PO Box 547  
Cambridge - MA 02142

Brainpower - 24009 Ventura Bd  
Calabasas - CA 91302

High Performances Systems  
13 Darmouth College Hwy Lyme - NH  
03768

X 10 - 185 A Le Grand Ave Northvale,  
NJ 07647

Eudimed - 98, bis bd de la Reine  
78000 Versailles - Tél. : 39 02 15 30

Jasmine Computer Systems  
PO Box 1119 Mountain View - CA  
94042

Styleware - 6405 Hillcroft, Suite 201  
Houston TX 77081

Batteries Included - 30 Mural St  
Richmond Hill - Ontario Canada L4B  
1h5

ISD Marketing - 20 Steelbase Road Unit  
12 Markham Ontario Canada - L3R 1R2

Electric Transit - 501 Mann Street suite  
116 - Thousand Oaks, CA 91360

Yam Educational Software  
2028 El Camino Real San Mateo - CA  
94403

Applied Engineering - PO Box 798  
Carrollton - TX 75006



Collaborateur de Pom's vend à l'état  
neuf, un lecteur 140Ko //c (1700,00 F),  
un moniteur //c (1100,00 F), un stand //c  
(250,00 F), une housse //c (375,00 F),  
un joyetick //e //c (375,00 F) et un  
manuel de référence //e (250,00 F).  
Écrire à la rédaction qui transmettra.



# Bibliographie

Alexandre Duback

**AppleWorks au travail**  
par Alain Gargadannec et  
Jean-Michel Jégo  
Éditions du PSI - 160 FF.

Cet ouvrage est composé de trois parties : Découverte de la base de données, du traitement de texte et du tableur - Plus loin avec la base de données et le tableur - Intégration avec AppleWorks. Bien écrit et agréablement présenté, il nous emmène à la maîtrise du logiciel, grâce à des applications destinées aux PME, artisans et commerçants (paye, facturation).

Il est utile, surtout pour les non-informaticiens et les "vilains" possédant une version piratée. Le principal intérêt de ce livre est de nous entraîner au-delà de la simple utilisation du programme (ce qui est facile, AppleWorks étant un logiciel ultra-fiable, conversationnel et bien documenté par une fonction d'aide), en nous montrant jusqu'où vont les possibilités de ce logiciel intégré, c'est-à-dire les articulations entre ses trois applications.

Ce livre ne s'adresse pas aux possesseurs de la version anglaise, dont les commandes ne sont pas les mêmes.

**Dictionnaire de micro-informatique**  
par Éric Duceau et Christophe Doë - Cedic-Nathan - 191 pages.

Un livre clair, comportant de nombreuses illustrations et bien présenté. Les informations ne sont pas toujours exactes : ainsi, le langage de programmation Ada( et non A.D.A.) a été créé par Jean Ichbiah (et non Ichbiach) et, à la date de parution de l'ouvrage, il existe déjà de nombreux compilateurs, contrairement à l'affirmation de l'auteur. Un double lexique français/anglais et anglais/français termine l'ouvrage.

**MacAstuces, Macintosh et Macintosh Plus, deuxième édition**  
par Hervé Thiriez  
Éditions du PSI - 160 FF.

Cette seconde édition de MacAstuces comporte une mise à jour, l'adjonction de nouvelles astuces d'utilisation du Macintosh et, surtout, une vingtaine de pages entièrement nouvelles et consacrées au Macintosh Plus. La disquette d'accompagnement peut être commandée aux Éditions MEV.

**Guide pratique d'Excel**  
par Hervé Thiriez - Cedic-Nathan  
170 pages grand format.

Un ouvrage de base pour l'apprentissage d'Excel, à l'aide de plusieurs cas de gestion réalistes. Le seul livre sur Excel qui, à notre connaissance,

indique quels sont les bogues du logiciel et comment les contourner. Une prise en main progressive et pédagogique, couvrant toute l'utilisation de base d'Excel, y compris le calcul de tables et la programmation de macros. De nombreuses astuces d'utilisation sont mises en relief et ressortent facilement grâce à une présentation en grisé. Une disquette d'accompagnement facultative comprend tous les tableaux de l'ouvrage, ainsi qu'une nouvelle police de caractères de taille 6, permettant de voir à l'écran un grand nombre de lignes et colonnes à la fois.

**Programming pearls**  
par Jon Bentley  
Addison-Wesley - En anglais.

C'est la reprise sous la forme d'un livre d'une série d'articles publiés dans *Communications of the ACM*, une sérieuse revue professionnelle pour informaticiens. Une mine d'or pour les programmeurs, qui y trouveront de multiples astuces de programmation, des routines de tri et recherche, des idées sur la façon de structurer efficacement les données, des conseils sur la mise au point des programmes. Indispensable pour les programmeurs...



## Ludologic



Taquin, Noir & blanc, hexagone magique, trois jeux de réflexion de difficulté croissante. Ces jeux qui nécessitent des neurones aussi calmes qu'entraînés, ne devraient pas décevoir les amateurs de puzzles et autres casse-têtes.

80,00 F franco, Bon de commande page 74  
(La disquette comprend les sources)

1	2	3	4	5	6	17	
19	7	24	9	31	26	18	22

# Bon de commande

## Disquettes

HAIFA source .....	(cf. Pom's n° 5) .....	à 60,00 F .....
DISK-MANAGER .....	(cf. Pom's n° 11) .....	à 450,00 F .....
BASICIUM .....	(cf. Pom's n° 13) .....	à 150,00 F .....
E.P.E. 5.0 .....	(cf. Pom's n° 23) .....	à 200,00 F .....
Échange E.P.E. 5.0 .....	(cf. Pom's n° 23) .....	à 80,00 F .....
PASCAL .....	(cf. Pom's n° 15) .....	à 80,00 F .....
MAX (Monteur étendu) .....	(cf. Pom's n° 18) .....	à 150,00 F .....
DOMINOS .....	(cf. Pom's n° 19) .....	à 80,00 F .....
COGO .....	(cf. Pom's n° 21) .....	à 150,00 F .....
LIIDOLOGIC .....	(cf. Pom's n° 25) .....	à 80,00 F .....
ORDICO .....	(cf. Pom's n° 26) .....	à 200,00 F .....

## Recueils

N°1, recueil des revues 1 à 4 .....	à 140,00 F .....
Disquettes d'accompagnement 1 à 4 .....	à 200,00 F .....
N°2, recueil des revues 5 à 8 .....	à 140,00 F .....
Disquettes d'accompagnement 5 à 8 .....	à 200,00 F .....
N°3, recueil des revues 9 à 12 .....	à 140,00 F .....
Disquettes d'accompagnement 9 à 12 .....	à 200,00 F .....

## Revue, disquettes

Revue 4 7 8 .....	à 35,00 F .....
Revue 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 .....	à 40,00 F .....
Disquettes Apple II, //e, //c	
1/2 3 4 5 6 7 8 9 10 11 12 13 .....	à 60,00 F .....
14 15 16 17 18 19 20 21 22 23 24 25 26 27 .....	
Disquettes Macintosh	
14/15/16 groupées .....	à 150,00 F .....
17 18 19 20 21 22 23 24 25 26 27 .....	à 80,00 F .....
Mac 'A' .....	à 80,00 F .....
MacAstuces .....	à 200,00 F .....
"Raccourci" .....	à 200,00 F .....

## Abonnements

Pour 6 numéros à partir du n° .....

Abonnement à la revue seule .....	à 225,00 F .....
Abonnement revue + disquettes Apple II, //e, //c .....	à 525,00 F .....
Abonnement revue + disquettes Macintosh .....	à 625,00 F .....

Total TTC :

Supplément avion hors CEE : 15,00F par numéro et/ou disquette :

Montant du règlement :

Envoyez ce bon et votre règlement à : EDITIONS MEV, 64 rue des Chantiers 78000 VERSAILLES

Nom : \_\_\_\_\_

Adresse : \_\_\_\_\_

# infomag

La Revue des Macintosh

6, place du Cl Fabien 75018 Paris 42 40 22 01

# 100% MAC

CHEZ  
VOTRE  
MARCHAND  
DE JOURNAUX



# P.S.I.

## VISE JUSTE



### PROGRAMMER

Clefs pour Macintosh **150 FF** –  
Basic Microsoft 2.0 sur Macintosh  
**250 FF** – Basic + 80 routines sur  
Apple II **95 FF** – Les ressources de l'Apple  
IIC **95 FF** – Assembleur de l'Apple **120 FF** –  
Introduction à ProDOS sur Apple **85 FF** –  
Système ProDOS sur Apple  
**190 FF** – Programmation  
système de l'Apple II **190 FF** –  
Apple, modems et serveurs  
**130 FF** – Clefs pour l'Apple IIC  
et IIE 65CO2 **145 FF**.

### DES LIVRES POUR CRÉER

Programmation des jeux d'Arcade sur  
Apple II **140 FF** – Apple, logique et  
systèmes experts **120 FF** – Création et  
animation graphique sur Apple **335 FF**.

### UTILISER

Mac Astuces **150 FF** – Multiplan  
pour Macintosh **110 FF** – Le livre de  
Jazz **220 FF** – 50 modèles Multiplan  
pour gérer sur Apple et IBM/PC **130 FF** –  
Appleworks au travail **160 FF** –  
Photographie sur Apple et Amstrad **150 FF**.

### JOUER

102 programmes pour  
Apple **120 FF** – Super jeux  
Apple **120 FF**.