

Global Applesoft Program Editor

Manuel d'utilisation

Note de l'auteur

J'ai décidé de développer ce projet après avoir vu les difficultés qu'affrontaient les programmeurs en Applesoft BASIC pour éditer leurs programmes. C'est donc pourquoi j'ai créé le programme Global Applesoft Program Editor (ou G.A.P.E).

Ce programme est écrit intégralement en assembleur 6502. En total il s'agit environ de 2300 lignes de code et est le résultat d'un grand nombre d'heures de travail.

Comme vous pouvez l'observer, le nom de commandes est en anglais. La raison de ce choix est simple, il s'agit de maintenir la consistance avec les commandes du système d'exploitation qui sont dans cette même langue. Il ne semblait pas cohérent de mélanger des mots en français avec des mots en anglais. Cependant, la documentation du programme est totalement en français pour faciliter sa compréhension.

En ce qui concerne le programme, il fut développé sur un Orange II, un ordinateur compatible Apple II. En principe ils sont identiques mais il est possible qu'il y ait quelques différences mineures et qu'elles puissent créer des problèmes si vous exécutez ce programme sur un Apple II original, mais je ne le crois pas.

Pour des raisons de manque de mémoire, le programme a du être divisé en deux parties. Pour la même raison j'ai du limiter le nombre de commentaires inclus dans le code source. J'espère cependant qu'ils seront suffisants pour comprendre comment il fonctionne.

Pour exécuter le program il suffit d'insérer la disquette dans le lecteur et allumer l'ordinateur Apple car le programme s'exécute automatiquement. J'espère qu'il vous sera utile.

Huibert Aalbers

Les commandes du G.A.P.E.

```
GLOBAL APPLESOFT PROGRAM EDITOR.  
PAR HUIBERT AALBERS  
  
>⌘  
DOS :
```

Les commandes de contrôle

Ces commandes s'appellent ainsi parce que pour les exécuter il faut presser simultanément la touche CTRL (ConTRoL) et une autre touche. Ces commandes ne fonctionnent que si on les invoque lorsque le curseur se trouve sur la première colonne de l'écran.

Le reste de cette section contient la description complète des fonctions de contrôle existantes ainsi que les touches auxquelles elles sont associées.

Les commandes du DOS

Ces fonctions permettent d'accéder directement aux fonctions principales du DOS. Si vous désirez utiliser d'autres fonctions il vous faudra quitter G.A.P.E. pour pouvoir le faire depuis le BASIC.

- CTRL-C: CATALOG
- CTRL-D: DELETE
- CTRL-L: LOAD
- CTRL-S: SAVE

Toutes ces commandes ne fonctionnent qu'avec le Drive 1. Si vous désirez travailler avec le drive 2 vous devez quitter G.A.P.E. (voir la commande QUIT).

G.A.P.E a été développé pour fonctionner avec le système d'exploitation DOS 3.3. et ne saura donc pas comment lire des disquettes formatées avec des versions plus anciennes de DOS ou avec d'autres systèmes d'exploitation comme par exemple PRODOS.

La commande QUIT

La combinaison de touches CTRL-Q permet de sortir de l'éditeur et de revenir au BASIC pour pouvoir par exemple exécuter votre programme. Si après ce-là vous désirez retourner vers l'éditeur de G.A.P.E. écrivez simplement: & <CR>

La commande NEW

Cette commande efface le programme BASIC qui se trouve en mémoire. C'est pourquoi, vous devrez confirmer l'opération avant de pouvoir continuer (poussez Y pour confirmer ou N pour annuler l'opération).

Exemple

Pressez: CTRL-N

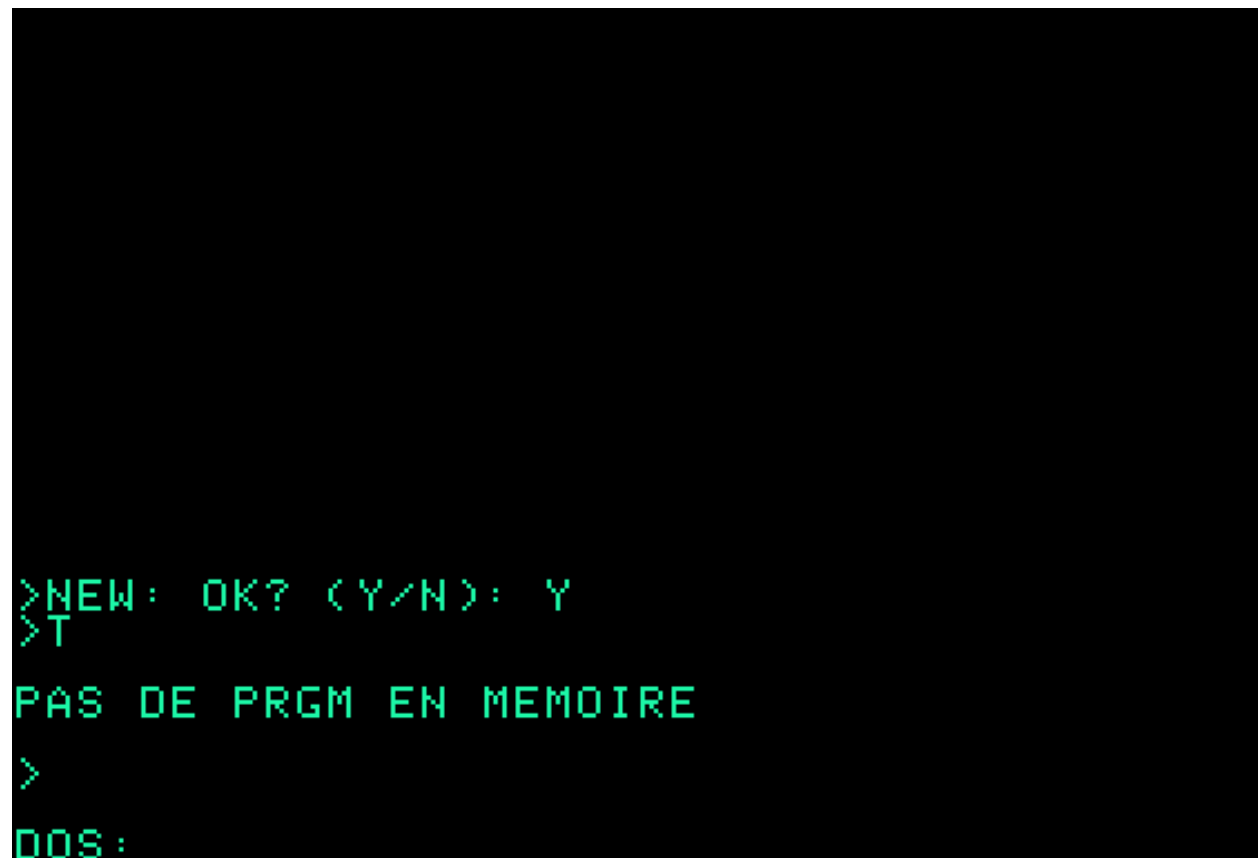
Résultat: NEW: OK? (Y/N):

Pressez: Y

Écrivez: T <CR>

Résultat: PAS DE PRGM EN MEMOIRE

(Le programme a été effacé)



```
>NEW: OK? (Y/N): Y
>T
PAS DE PRGM EN MEMOIRE
>
DOS:
```

Commandes de position

L'application G.A.P.E. considère un programme BASIC comme s'il s'agissait d'un fichier. Cela explique la façon dont on se déplace d'une ligne vers une autre mais aussi la présence du mot "GLOBAL" dans G.A.P.E. étant donné qu'il ne considère pas un programme BASIC comme un groupe de lignes indépendantes les unes des autres mais au contraire comme une structure indissoluble.

Pour tous les exemples suivants nous utiliserons ce petit programme:

```
10 HOME
20 INPUT "ANGLE EN RADIANS";A
30 A=(A*180)/3.14
40 PRINT "ANGLE EN DEGRES = ";A
50 END
```

La commande TOP

Cette commande montre la première ligne du programme et place le pointeur sur elle.

Exemple

```
Écrivez: TOP<CR>
Résultat: TOP
          10 HOME
           : VTAB 10
           : HTAB 5
```

La commande BOTTOM

Cette commande montre la dernière ligne du programme et place le pointeur sur elle.

Exemple

```
Écrivez: BOTTOM<CR>
Résultat: BOTTOM
          50 END
```

La commande NEXT

Cette commande permet de se déplacer un nombre X de lignes vers le haut ou le bas du programme, dépendant du signe de la valeur X.

La syntaxe de cette commande est la suivante:

```
N<EXT> <-> <NUMERO>
```

Exemple

```
Écrivez: TOP<CR>
Résultat: TOP
```

```

10 HOME
   : VTAB 10
   : HTAB 5
Écrivez: N<CR>
Résultat: 20 INPUT "ANGLE EN RADIAN";A
Écrivez: N 2<CR>
Résultat: 40 PRINT "ANGLE EN DEGRES = ";A
Écrivez: NEXT 5<CR>
Résultat: BOTTOM          (Le programme a atteint le "fond" du fichier)
50 END
Écrivez: N -2<CR>
Résultat: 30 A= (A*180)/3.14

```

La commande LINE

Cette commande permet de se déplacer directement sur une ligne de code déterminée.

La syntaxe de cette commande est la suivante:

```
L<LINE> <NUMERO>
```

NUMERO est une valeur 16 bits (comprise entre 1 et 65535).

Exemple

```

Écrivez: B<CR>
Résultat: BOTTOM
50 END
Écrivez: L 10<CR>
Résultat: 10 HOME
          : VTAB 10
          : HTAB 5

```

La commande PRINT

Cette commande imprime un certain nombre de lignes de code depuis la ligne actuelle (celle qui est pointée par G.A.P.E.).

La syntaxe de cette commande est la suivante:

```
P<PRINT> <NUMERO>
```

NUMERO est une valeur 16 bits (comprise entre 1 et 65535). Par défaut elle vaut 65535.

Exemple

```

Écrivez: T<CR>
Résultat: TOP
10 HOME
   : VTAB 10
   : HTAB 5
Écrivez: PRINT 2<CR>

```

```
Résultat: 10 HOME
           : VTAB 10
           : HTAB 5
           20 INPUT "ANGLE EN RADIANS";A
Écrivez:  PRINT<CR>
Résultat: 20 INPUT "ANGLE EN RADIANS";A
           30 A=(A*180)/3.14
           40 PRINT "ANGLE EN DEGRES = ";A
           50 END
           BOTTOM (Le programme a atteint le "fond" du fichier)
```

La commande PP (PinPoint)

Cette commande permet au programmeur de voir la ligne actuelle en contexte. Pour ce-là elle montre les deux lignes précédentes et les deux lignes postérieures à la ligne actuelle, sans changer cette dernière.

La syntaxe de cette commande est la suivante:

PP

Exemple

```
Écrivez: L 30<CR>
Résultat: 30 A=(A*180)/3.14
Écrivez: PP<CR>
           10 HOME
           : VTAB 10
           : HTAB 5
           20 INPUT "ENTRE EL ANGULO EN RADIOS";A
           30 A=(A*180)/3.14
           40 PRINT "ANGLE EN DEGRES = ";A
           50 END
```

Aides à la programmation

Les fonctions ESCape

L'un des principaux problèmes que peut avoir un développeur est de ne pas savoir bien taper à la machine, ce qui peut avoir un impact négatif sur sa productivité. Même pour ceux qui n'ont pas ce problème, il peut être très positif de pouvoir réduire au maximum le nombre de touches à presser pour écrire un programme. C'est pour cette raison que G.A.P.E. dispose d'un système pour écrire la plupart des commandes de l'Applesoft BASIC en n'ayant à appuyer que sur deux touches (ESC suivi par la touche correspondante à la fonction choisie).

La tableau suivant montre une liste des fonctions Applesoft, avec leur touche correspondante.

Touche	Fonction
ESC-A	ABS(
ESC-B	HYPLOT
ESC-C	CLEAR
ESC-D	DATA
ESC-E	END
ESC-F	FOR
ESC-G	GOTO
ESC-H	HOME
ESC-I	INPUT
ESC-J	CHR\$(
ESC-K	RIGHT\$(
ESC-L	LEFT\$(
ESC-M	MID\$(
ESC-N	NEXT
ESC-O	POKE
ESC-P	PEEK(
ESC-Q	CALL
ESC-R	RETURN
ESC-S	STEP
ESC-T	TEXT

Touche	Fonction
ESC-U	HTAB
ESC-V	VTAB
ESC-W	DRAW
ESC-X	XDRAW
ESC-Y	HCOLOR=
ESC-Z	RESTORE

J'ai cherché à faire que cette liste soit le plus simple possible à mémoriser. C'est pourquoi la première lettre de la fonction est en général la même que la touche qu'il faut presser pour l'obtenir. Cependant, ceci n'est pas toujours possible car il y a des cas où plusieurs noms de fonctions commencent avec la même lettre (c'est le cas par exemple pour PEEK et POKE). Dans ces cas là, les fonctions sont groupées par thème. Ainsi, PEEK, POKE et CALL (des fonctions avancées pour accéder à la mémoire de l'Apple II et aux routines de la ROM) ont été assignées à un groupe de trois lettres consécutives, ce qui devrait faciliter la mémorisation.

Restrictions

L'utilisation des fonctions ESCape n'est pas possible en mode EDIT (voir la commande EDIT) étant donné que dans ce mode, l'utilisation de la touche "ESC" a une fonction différente que nous verrons postérieurement.

La commande AUTO

Lorsque l'on écrit un programme en Applesoft BASIC il est ennuyeux d'avoir à écrire à chaque fois un nouveau numéro de ligne. Cette commande nous aide en le faisant de façon automatique.

La syntaxe de cette commande est la suivante:

```
A<UTO> <LIGNE>,<INCREMENT>
```

LIGNE est une valeur 16 bits (comprise entre 1 et 63999).

INCREMENT est une valeur 8 bits (comprise entre 1 et 255).

Après avoir utilisé cette commande, chaque fois que vous pressez la barre d'espace lorsque le curseur est sur la première colonne de l'écran, un nouveau numéro de ligne apparaîtra. Vous n'avez donc plus qu'à écrire la fin de votre ligne de code et presser RETURN. Pour écrire une autre ligne, pressez à nouveau la barre d'espace et un nouveau numéro de ligne apparaîtra automatiquement. Répétez l'opération jusqu'à ce que vous terminiez d'écrire votre programme.

Exemple

Écrivez: AUTO 10,5<CR>

Résultat: >

Pressez: <Espace>

Résultat: >10

Écrivez: HOME: CLEAR<CR>

Pressez: <Espace>

Résultat: >15

La commande AUTOFF

Cette commande permet de désactiver les effets de la commande AUTO.

La syntaxe de cette commande est la suivante:

```
AUTOFF<F>
```

Exemple

Exécutez l'exemple précédent et après,

Écrivez: AUTOFF<CR>

Résultat: >

Écrivez: <Espace>

Résultat: >

(Le mode AUTO a été désactivé)

La commande COLUMN

Parfois, lorsqu'on écrit un programme il arrive que l'on doive centrer un texte sur l'écran en utilisant la fonction PRINT. Dans ces cas il est utile de pouvoir savoir exactement dans quelle colonne on est en train d'écrire. Pour résoudre ce problème, la fonction COLUMN énumère au bas de l'écran chacune des colonnes.

La syntaxe de cette commande est la suivante:

```
C<OLUMN>
```

Note

Étant donné que les deux dernières lignes de l'écran sont protégées, il est impossible que vous puissiez les effacer par erreur.

```
50 PRINT D$;"BLOAD P4,A$8272"  
60 CALL 28672  
BOTTOM.  
  
>N  
  
BOTTOM.  
60 CALL 28672  
>NEW: OK? (Y/N): Y  
>  
>AUTO 10,10  
>10 HOME  
>COL  
>  
1234567890123456789012345678901234567890  
DOS:
```

La commande NOCOLUMN

Elle désactive les effets de la commande COLUMN et libère l'avant-dernière ligne de l'écran.

La syntaxe de cette commande est la suivante:

```
NO<COLUMN>
```

La commande H\$

En Applesoft BASIC, toutes les valeurs doivent s'exprimer en décimal. Cependant, un grand nombre de livres utilise des valeurs en hexadécimal pour représenter les positions mémoire importantes de l'Apple II. C'est pour ce-là que lorsqu'on programme en BASIC il est important de disposer d'un outil qui nous permette de convertir facilement et rapidement des valeurs hexadécimales en décimal. C'est exactement ce que fait la commande H\$.

La syntaxe de cette commande est la suivante:

```
H$ VALEUR
```

VALEUR est une valeur hexadécimale 16 bits (comprise entre 0 et FFFF).

Exemple

Écrivez: H\$E5<CR>

Résultat: =229

Écrivez: H\$C050

Résultat: =49232

La commande SEARCH

Lorsque l'on écrit ou corrige un programme assez long il est facile d'oublier le numéro de la ligne où l'on a écrit telle ou autre chose. C'est pour cette raison que cette commande est si utile car elle nous permet de trouver une chaîne de caractères spécifique dans notre code et elle nous montre sur l'écran toutes les lignes qui la contiennent.

La syntaxe de cette commande est la suivante:

```
SEARCH CHAINE</NUMERO>
```

CADENA est une chaîne de caractères longue de moins de 128 lettres.

NUMERO contient le nombre de lignes sur lesquelles va s'effectuer la recherche. Sa valeur par défaut est 255. NUMERO doit être un nombre entier compris entre 1 et 255.

Note

La dernière ligne de code sur laquelle se réalise la recherche devient la ligne actuelle.

Exemple

Copiez le programme suivant:

```
10 HOME  
20 INPUT "HOLA! QUE FAITES-VOUS?";A$  
30 END
```

Écrivez: T<CR>

Résultat: 10 HOME

Écrivez: S HO<CR>

Résultat: 10 HOME

```
20 INPUT "HOLA! QUE FAITES VOUS ?";A$
```

BOTTOM (G.A.P.E. a trouvé "HO" sur les lignes 10 et 20 avant d'atteindre la fin du programme)

Écrivez: T<CR>

Résultat: 10 HOME

Écrivez: SEARCH B\$<CR>

Résultat: BOTTOM

(G.A.P.E n'a pas trouvé la chaîne de caractères "B\$")

Éditer une ligne de programme

Cette fonctionnalité est au coeur du système G.A.P.E. car elle permet d'améliorer de façon significative la façon de modifier une ligne d'un programme Applesoft BASIC. En fait, l'Apple II ne permet pas vraiment de modifier une ligne de code existante, l'utilisateur doit en gros écrire à nouveau toute la ligne pour remplacer la version antérieure. C'est beaucoup de travail inutile si l'on ne veut faire qu'une petite modification. Avec G.A.P.E. modifier une ligne de code est beaucoup plus facile puisque l'on peut faire des modifications sans avoir à taper à nouveau toute la ligne.

La commande EDIT

Cette commande permet de modifier facilement une ligne de programme. De plus, elle vous montre toujours la ligne formatée de façon optimisée comme le fait la commande PRINT ce qui facilite la lecture du code. Ceci est important parce que ça simplifie l'édition des lignes de code longues et complexes tout en maintenant la consistance avec les listings en papier (si vous disposez d'une imprimante).

La syntaxe de cette commande est la suivante:

```
E<DIT> <LIGNE>
```

LIGNE est une valeur comprise entre 1 et 63999 (les valeurs permises pour une ligne de programme écrite en Applesoft BASIC). Sa valeur par défaut est celle de la ligne courante (celle sur laquelle pointe le programme G.A.P.E.).

Lorsque l'on édite une ligne de code G.A.P.E. la montre à l'écran et place le curseur au début de la ligne, attendant que vous envoyiez une commande. Le tableau suivant montre une liste des différentes commandes dont vous disposez en mode EDIT.

Commande	Description
D	Efface le caractère placé sous le curseur, déplaçant les autres caractères vers la gauche pour remplir l'espace resté vide.
I	Insère un espace sous le curseur, déplaçant les autres caractères vers la droite.
F	Cherche un caractère dans la ligne de code et déplace le curseur sur celui-ci. Après avoir pressé la touche F il faut presser la touche correspondante au caractère que vous voulez trouver.
B	Place le curseur au début (Beginning) de la ligne.
E	Place le curseur à la fin (End) de la ligne.
ESC	Permet de changer entre le mode TYPE où l'on peut écrire librement et le mode EDIT où l'on peut utiliser les commandes que montre ce tableau.
CTRL-X	Annule toutes les modifications faites depuis le début de l'opération et laisse à nouveau la ligne dans son état initial, sans changements.
RETURN	Termine l'édition de la ligne et conserve tous les changements effectués par le programmeur.

Note

Huibert Francisco-José Aalbers Indaberea - Prix Hollande (1985)

Les flèches fonctionnent normalement et permettent de se déplacer au long de la ligne de code tant en mode EDIT comme en mode TYPE. La seule différence est qu'il est impossible d'avancer au-delà du dernier caractère de la ligne. De plus il n'est pas possible d'utiliser la commande I (Insert) si votre ligne de code a déjà atteint le nombre maximum de caractères (250).

La dernière ligne de l'écran montre à tout moment si l'on se trouve en mode EDIT (EDIT: COMMAND) ou en mode TYPE (EDIT: TYPE).

Annexes

Annexe A - Messages d'erreur

Messages d'erreur du DOS

Ces messages apparaissent sur la dernière ligne de l'écran. Pour connaître leur signification exacte, veuillez consulter le manuel d'utilisation du DOS 3.3 édité par Apple. Les messages d'erreur les plus fréquents sont les suivants:

FILE NOT FOUND	L'ordinateur ne réussit pas à trouver le programme Applesoft sur la disquette. Vous avez probablement mal écrit le nom du programme ou inséré une disquette où le programme ne se trouvait pas.
I/O ERROR	L'ordinateur ne réussit pas à lire les données de la disquette. La disquette peut être endommagée ou vous n'avez peut-être pas fermé correctement le lecteur de disquettes..
DISK FULL	Votre disquette n'a pas assez d'espace libre pour sauvegarder votre programme.
WRITE PROTECTED	Vous avez essayé de sauvegarder un programme sur une disquette protégée contre l'écriture. Pour pouvoir sauvegarder votre programme vous devez déprotéger votre disquette en enlevant l'étiquette qui couvre l'encoche du côté droit de votre disquette.

Messages d'erreur du G.A.P.E.

SORRY	G.A.P.E n'a pas compris votre commande. Veuillez vérifier que vous n'avez pas commis une erreur de syntaxe en écrivant la commande.
NO SUCH LINE	La ligne de programme vers laquelle vous voulez vous déplacer n'existe pas.
BAD ARG #	L'argument utilisé pour votre commande contient une valeur incorrecte. Veuillez vérifier que vous n'avez pas commis une erreur de syntaxe en écrivant la commande.
TOP	Le programme a atteint la première ligne du programme.
BOTTOM	Le programme a atteint la dernière ligne du programme.

Annexe B - Bibliographie

Livres

- **La pratique de l'Apple II (Volume III)** de Nicole Bréaud Pouliquen et Daniel-Jean David. Édition P.S.I.
- **Clefs pour l'Apple II** de Nicole Bréaud Pouliquen. Édition P.S.I.
- **Beneath Apple DOS** de Don Worth et Pieter Lechner. Quality Software.

Magazines

- **NIBBLE Magazine Number 6, Vol. 3 (March 1985)** Ed MicroSPARC, Inc.

Annexe C - Ressources utilisées

Hardware

- **Apple II+** 48K RAM
- **Apple II Disk Drive** DOS 3.3.

Software

- **LISA 2.5 Assembler**. Édité par PROGRAMMA International Inc.
- **Apple Writer II**. Édité par Apple Computer Inc. (traitement de textes utilisé pour imprimer le manuel d'utilisation).

Global Applesoft Program Editor

Code source

Archivo PHILIPS.1

```
1 ;
2 *****
3 *
4 *          GLOBAL          *
5 * APPLESOFT PROGRAM EDITOR *
6 *
7 *    DEVELOPPE PAR        *
8 *
9 *    HUIBERT AALBERS     *
10 *
11 *****
12 ;
13 ;-----
14 ;DEFINITION DES ROUTINES EN ROM
15 ;OU DU D.O.S
16 ;-----
17 ;
18 BUF      EQU  $200      ;BUFFER UTILIZADO POR INLIN
19 AMPER    EQU  $3F5      ;VECTOR DEL &
20 PRGM     EQU  $801      ;PRINCIPIO DEL PROGRAMA
21 LIST     EQU  $74E5     ;LISTA UNA LINEA
22 ESCODES  EQU  $75FF     ;TRATAMIENTO DE LOS COMANDOS DE ESC.
23 ENTRY2   EQU  $78D6
24 ENTRY    EQU  $7E2B
25 CLOSEALL EQU  $A316     ;CIERRA TODOS LOS FICHEROS
26 TKNTBL   EQU  $D0D0     ;TABLA DE COMANDOS DEL BASIC
27 BLTU     EQU  $D393     ;MUEVE 256 OCTETOS HACIA ARRIBA
```

28	INLIN	EQU	\$D52C	;ENTRA UNA LINEA HACIA EL BUFFER
29	PARSE	EQU	\$D559	;CODIFICA UNA LINEA ENTRADA POR INLIN
30	FNDLIN	EQU	\$D61A	;BUSCA UNA LINEA DE PROGRAMA
31	INITPTRS	EQU	\$D665	;PUNTEROS DE INIT
32	LINGET	EQU	\$DA0C	;PONE EN LINNUM EL NO DE LINEA CORRIENTE
33	CRDO	EQU	\$DAFB	;IMPRIME UN CARRIAGE RETURN
34	OUTSP	EQU	\$DB57	;IMPRIME UN ESPACIO
35	LINPRT	EQU	\$ED24	;IMPRIME X,A
36	VTAB	EQU	\$F25A	;TABULACION VERTICAL
37	PRBL2	EQU	\$F94A	;IMPRIME X ESPACIOS
38	HOME	EQU	\$FC58	;BORRA LA PANTALLA
39	KEYIN	EQU	\$FD0C	;ENTRA UN CARACTER
40	GETLN	EQU	\$FD6A	;ENTRA UNA LINEA DE CARAC. HACIA BUF
41	COUT	EQU	\$FDED	;IMPRIME EL CARACTER CONTENIDO EN A
42				;
43				;-----
44				;DEFINICION DES ADRESSES
45				;DE LA PAGE ZERO
46				;-----
47				;
48	POSCUR	EPZ	\$06	
49	ADL	EPZ	\$08	;VECTOR ENTRADA DEL FILE MANAGER
50	TEMP	EPZ	\$0F	
51	LINE	EPZ	\$18	;REGISTRO TEMPORAL SOBRE 2 OCTETOS
52	BUFF	EPZ	\$1A	;REGISTRO TEMPORAL SOBRE 2 OCTETOS
53	CH	EPZ	\$24	;POSICION DEL CURSOR (HORIZONTAL)
54	PROMPT	EPZ	\$33	;CONTIENE EL CARACTER ">"
55	LINNUM	EPZ	\$50	;CONTIENE EL NUMERO DE LINEA TRAS LINGET
56	LIST2	EPZ	\$54	;FLAG PARA HACER LIST ANTES DE CMDLP

57	ESCFLG	EPZ	\$55		;FLAG DE PULSACION DE ESC
58	INDEX	EPZ	\$5E		;REGISTRO TEMPORAL PARA MOVER MEMORIA
59	DEST	EPZ	\$60		
60	TXTTAB	EPZ	\$67		;DIRECCION PRINCIPIO DEL PROGRAMA BASIC
61	VARTAB	EPZ	\$69		;DIRECCION PRINCIPIO VARIABLES SIMPLES
62	STREND	EPZ	\$6D		;DIRECCION FIN VARIABLES DIMENSIONADAS
63	FRETOP	EPZ	\$6F		;DIRECCION FIN VARIABLES ALFANUMERICAS
64	MEMSIZ	EPZ	\$73		;DIRECCION FIN DE LA MEMORIA UTILIZABLE
65	CURLIN	EPZ	\$75		;FLAG (CONTIENE FF EN MODO INMEDIATO)
66	FORPNT	EPZ	\$85		;NUEVA DIRECCION DE UNA CADENA DESPLAZADA
67	HIGHDS	EPZ	\$94		;PARAMTROS BLTU: DESTINO
68	HIGHTR	EPZ	\$96	; 66	FIN
69	LOWTR	EPZ	\$9B	;	PRINCIPIO
70	DSCTMP	EPZ	\$9D		;REGISTRO TEMPORAL SOBRE 6 OCTETOS
71	PRGEND	EPZ	\$AF		;FINAL DE LA ZONA PROGRAMA BASIC
72	CHRGET	EPZ	\$B1		;SUBROUTINA QUE LEE EL SIGIENTE CARACTER
73	;				;DEL PROGRAMA BASIC
74	TXTPTR	EPZ	\$B8		;DIRECCION CARACTER OBTENIDO POR CHRGET
75	INC	EPZ	\$CE		
76	ERRFLG	EPZ	\$D8		;FLAG QUE INDICA SI ONERR ESTA ACTIVO
77	ADL2	EPZ	\$F9		;REGISTRO TTEMPORAL SOBRE 2 OCTETOS
78	SAVEA	EPZ	\$FB		
79	AUTOFLG	EPZ	\$FF		
80	;				
81	;				-----
82	;				PREMIO HOLANDA. PROGRAMME
83	;				COMEMNCE LE 6 DE AVRIL DE 1984
84	;				-----
85	;				

```

86          ORG $300
87 ;
88 ;-----
89 ;SUBROUTINA POR LA QUE PASA EL
90 ;PROGRAMA CADA VEZ QUE SE PULSA
91 ;UNA TECLA
92 ;-----
93 ;
94 KEYINTCP JSR $FD1B      ;LEE UN CARACTER DESDE EL TECLADO
95          PHA
96          CPX #$00      ;SI EL CURSOR NO ESTA EN LA PRIMERA
97          BEQ >1
98          JMP ANULADO
99 ^1      LDA #$00
100         STA ESCFLG
101         PLA
102         CMP #$83      ;ES UN CTRL-C ?
103         BNE NOCAT
104         LDX #$00
105 AFFCAT  LDA MESSCAT,X  ;IMPRIME "CATALOG"
106         BEQ PREPDAT
107         JSR COUT
108         INX
109         JMP AFFCAT
110 PREPDAT JSR $3DC      ;PREPARACION DE LOS DATOS NECESARIOS
111         STA ADL+1      ;PARA EFECTUAR UN CATALOG
112         STY ADL
113         LDY #$00
114         LDA #$06

```



```

115          STA  (ADL),Y
116          LDA  #$01
117          LDY  #$05
118          STA  (ADL),Y
119          LDA  #$06
120          LDY  #$06
121          STA  (ADL),Y
122          JSR  $3D6          ;LLAMADA AL FILE MANAGER
123          JMP  DOSERR       ;VERIFICA SI HA HABIDO UN ERROR
124  ;
125  MESSCAT  ASC  "CATALOG"
126          HEX  $00
127  ;
128  NOCAT    CMP  #$93          ;ES UN CTRL-S ?
129          BNE  NOSAVE
130          JMP  SAVE
131  NOSAVE    CMP  #$8C          ;ES UN CTRL-L ?
132          BNE  NOLOAD
133          JMP  LOAD
134  NOLOAD    CMP  #$84          ;ES UN CTRL-D ?
135          BNE  NODEL
136          JMP  DELETE
137  NODEL     CMP  #$8E          ;ES UN CTRL-N ?
138          BNE  NONEW
139          JMP  NEW
140  NONEW     CMP  #$91          ;ES UN CTRL-Q ?
141          BNE  NOQUIT
142          JMP  QUIT
143  NOQUIT    CMP  #" "          ;ES UN ESPACIO ?

```

144	BNE NOCOM	;SI NO, NO ES UN COMANDO.
145	LDY AUTOFLG	;ESTA PUESTA LA NUMEARION AUTOMATICA ?
146	BEQ NOCOM	;SI NO,NO ES UN COMANDO.
147	LDA INC	;CALCULO PROXIMA LINEA
148	CLC	
149	ADC LINE	
150	STA LINE	
151	TAX	
152	LDA #\$00	
153	ADC LINE+1	
154	STA LINE+1	
155	JSR LINPRT	;IMPRIME EL NUMERO EN PANTALLA
156	LDX #\$00	
157 ^2	LDA \$100,X	;Y LO SALVAGUARDA EN MEMORIA
158	STA \$200,X	
159	BEQ >3	
160	INX	
161	JMP <2	
162 ^3	LDA #" "	
163 NOCOM	JMP ANULADO+1	
164 ;		
165 ANULADO	PLA	;TRATAMIENTO DE LOS COMANDOS DE "ESCAPE"
166	CMP #\$95	
167	BEQ >1-1	
168	CMP #\$9B	
169	BNE >1	
170	STA ESCFLG	
171	RTS	
172 ^1	LDY ESCFLG	

```

173          BNE >2
174          RTS
175 ^2      CMP #\$C1
176          BCC >3
177          CMP #\$DB
178          BCS >3
179          SBC #\$C0
180          JMP ESCODES
181 ^3      LDY #\$00
182          STY ESCFLG
183          RTS
184 ;
185 VIINTCP  CMP #\$A0          ;RUTINA DE SALIDA DE CARACTERES.
186          BCS >1          ;IMPRIME LOS CARACTERES DE CONTROL
187          CMP #\$8D          ;EN INVERSO.
188          BEQ >1
189          CMP #\$88
190          BEQ >1
191          AND #%00111111
192 ^1      JSR \$FDF0
193          RTS
194 ;
195          ORG \$7000
196 ;
197 ;-----
198 ;PRINCIPIO DEL PROGRAMA.ESCRIBE
199 ;EL TITULO E INICIALIZA LAS
200 ;VARIABLES Y LAS E/S.
201 ;-----

```

```

202 ;
203 JSR HOME ;BORRA LA PANTALLA
204 ;
205 ;MODIFICA LOS VECTORES DE E/S
206 ;
207 INIT LDA #$00
208 STA $9D02 ;DESCONECTA LOS CONTROLES DEL D.O.S
209 STA $38
210 LDA #$03
211 STA $9D03
212 STA $9D05
213 STA $37
214 STA $39
215 LDA #VIINTCP ;Y TOMA EL CONTROL DE LAS ENTRADAS Y
216 STA $9D04 ;SALIDAS DE CARACTERES
217 STA $36
218 LDX #$02
219 AMPVCT LDA VECT,X ;INSTALA LOS VECTORES QUE PERMITIRAN EL
220 STA AMPER,X ;USO DEL & PARA VOLVER A ENTRAR EN EL
221 DEX ;G.A.P.E DESDE EL BASIC
222 BPL AMPVCT
223 RESETVCT LDA #RESET ;BLOQUEA LA TECLA RESET,EVITANDO QUE SE
224 STA $3F2 ;PULSE POR EQUIVOCACION
225 LDA /RESET
226 STA $3F3
227 JSR $FB6F
228 SETMEM LDA #$00 ;PROTEGE LA ZONA MEMORIA DONDE ESTA EL
229 STA MEMSIZ ;G.A.P.E PARA QUE NO SEA DESTRUIDO POR
230 LDA #$70 ;LAS VARIABLES O EL PROGRAMA DEL BASIC

```

```

231          STA MEMSIZ+1
232 ;-----
233 ;INICIALIZACION DE ALGUNOS
234 ;REGISTROS
235 ;-----
236          LDA #$00
237          STA AUTOFLG
238          STA LINE+1
239          LDA #!20
240          STA LINE
241          LDA #$0A
242          LDA #$01
243          STA $07
244          LDA #$00
245          STA LIST2
246 ;-----
247 ;IMPRIME EL TITULO
248 ;-----
249 AFFHEL   JSR CRDO
250          LDA #$05
251          STA CH
252          LDX #$00
253 ^1       LDA MES1,X
254          BEQ >2
255          JSR COUT
256          INX
257          JMP <1
258 ^2       JMP NDOERR
259 ;-----

```

```
260 ;BUCLE PRINCIPAL
261 ;-----
262 CMDLP    LDA  #$00
263          STA  ESCFLG
264          NOP
265          NOP
266          NOP
267          LDA  LIST2
268          BEQ  CMDLP2
269          JSR  LIST
270          LDA  #$00
271          STA  LIST2
272          LDA  #$00
273          STA  SAVEA
274 CMDLP2   JSR  CRDO
275          LDX  #">"
276          JSR  INLIN+2
277          STX  TXTPTR
278          STY  TXTPTR+1
279          LSR  ERRFLG
280          JSR  CHRGET
281          TAX
282          BEQ  CMDLP
283          LDX  #$FF
284          STX  CURLIN+1
285          BCC  PROCLN
286          JMP  PARSE2
287          JMP  CMDLP
288 ;-----
```

```

289 ;PROCEDE A LA ENTRADA DE UNA
290 ;LINEA DE PROGRAMA
291 ;-----
292 PROCLN   LDX PRGEND       ;PONE LOMEM=FIN DEL PROGRAMA.
293         STX VARTAB
294         LDX PRGEND+1
295         STX VARTAB+1
296         JSR LINGET       ;PONE # DE LINEA EN LINNUM
297         JSR PARSE       ;TRANSFORMA EL BUFFER EN "TOKEN"
298         STY TEMP        ;INDEX DE BUFFER (# CARACTER+5)
299         JSR FNDLIN      ;EXISTE YA LA LINEA ?
300         BCC NEWLN      ;NO, HAY QUE CREARLA
301 ;-----
302 ;DESTRUYE UNA ANTIGUA LINEA
303 ;-----
304         LDY #1          ;SI EXISTE.BORRA LA LINEA MOVIENDO EL
305         LDA (LOWTR),Y   ;RESTO DEL PROGRAMA HACIA ABAJO, ES
306         STA INDEX+1     ;DECIR ESCRIBIENDO SOBRE LA LINEA QUE
307         LDA VARTAB      ;HA DE SER BORRADA
308         STA INDEX       ;LAS LINEAS 296 A 326 CALCULAN LO
309         LDA LOWTR+1    ;SIGUIENTE:
310         STA DEST+1     ;(DEST),Y=PRIMER CARACTER DE LA LINEA
311         STA DEST+1     ;QUE HA DE SER BORRADO
312         LDA LOWTR
313         DEY
314         SBC (LOWTR),Y   ;(INDEX),Y=PRIMER CARACTER DEL RESTO
315         CLC            ;DEL PROGRAMA
316         ADC VARTAB
317         STA VARTAB     ;X-1=LONGITUD DEL RESTO DEL PROGRAMA

```

318	STA DEST	; (HI-BYTE)
319	LDA VARTAB+1	
320	ADC #\$FF	;\$100-Y=LONGITUD DEL RESTO DEL PROGRAMA
321	STA VARTAB+1	; (LO-BYTE)
322	SBC LOWTR+1	
323	TAX	
324	SEC	
325	LDA LOWTR	
326	SBC VARTAB	
327	TAY	
328	BCS PL1	
329	INX	
330	DEC DEST+1	
331 PL1	CLC	
332	ADC INDEX	
333	BCC PGMDWN	
334	DEC INDEX+1	
335	CLC	
336 PGMDWN	LDA (INDEX),Y	;MUEVE EL PROGRAMA HACIA ABAJO
337	STA (DEST),Y	
338	INY	
339	BNE PGMDWN	
340	INC INDEX+1	
341	INC DEST+1	
342	DEX	
343	BNE PGMDWN	
344	;-----	
345	;INSERTA UNA NUEVA LINEA	
346	;-----	

347	NEWLN	LDA BUF	;SI NO HAY CARACTERES TRAS EL # DE LINEA
348		BEQ SETPTRS	;ABANDONA LA INSERCIÓN.
349		LDA MEMSIZ	;PONE FINAL DE LA CADENA
350		LDY MEMSIZ+1	;ESPACIO=HIMEM
351		STA FRETOP	
352		STY FRETOP+1	
353		LDA VARTAB	;PONE PARAMETROS BLTU (START=LOWTR)
354		STA HIGHTR	;FINAL (LO-BYTE)
355		ADC TEMP	
356		STA HIGHDS	;DESTINO (LO-BYTE)
357		LDY VARTAB+1	
358		STY HIGHTR+1	;FIN (HI-BYTE)
359		BCC PGMUP	
360		INY	
361	PGMUP	STY HIGHDS+1	;DESTINO (HI-BYTE)
362		JSR BLTU	;SUBE EL PROGRAMA
363		LDA LINNUM	;INSERTA #LINEA EN LO QUE SERAN LOS
364		LDY LINNUM+1	;BYTES 2-3 DE LA NUEVA LINEA.
365		STA BUF-2	
366		STY BUF-1	
367		LDA STREND	;PONE LOMEM=PRINCIPIO VAR. DIMENSIONADAS
368		LDY STREND+1	
369		STA VARTAB	
370		STY VARTAB+1	
371		LDY TEMP	
372	INSLN	LDA BUF-5,Y	;INSERTA LINEA EN LA MEMORIA LIBRE
373		DEY	;BUF-4 Y BUF-3 SERAN LOS LINK-BYTES 0-1
374		STA (LOWTR),Y	
375		BNE INSLN	

```

376 SETPTRS JSR INITPTRS ;INICIALIZA PUNTEROS
377 LDA TXTTAB ;PONE INDEX=PRINCIPIO DEL PROGRAMA
378 LDY TXTTAB+1
379 STA INDEX
380 STY INDEX+1
381 CLC
382 ;
383 ;PONE LINK BYTES
384 ;
385 FNDEOP LDY #$01
386 LDA (INDEX),Y ;FINAL DEL PROGRAMA?
387 BNE SETLINK ;NO.PONE LINK BYTES
388 LDA VARTAB ;SI. PONE FINAL PROGRAMA=LOMEM
389 STA PRGEND
390 LDA VARTAB+1
391 STA PRGEND+1
392 LDA SAVEA
393 BNE >1
394 JMP CMDLP ;VUELVE AL BUCLE PRINCIPAL
395 ^1 CMP #$01
396 BNE >2
397 JMP ENTRY
398 ^2 JMP ENTRY2
399 SETLINK LDY #$04 ;BUSCA EL FIN DE LA LINEA
400 SL1 INY
401 LDA (INDEX),Y
402 BNE SL1
403 INY ;LO HA ENCONTRADO.PONE LINK-BYTES.
404 TYA

```

```

405          ADC  INDEX
406          TAX
407          LDY  #$00
408          STA  (INDEX),Y    ;LINK (LO-BYTE)
409          LDA  INDEX+1
410          ADC  #$00
411          INY
412          STA  (INDEX),Y    ;LINK (HI-BYTE)
413          STX  INDEX        ;PONE PUNTERO AL PRINCIPIO DE LA
414          STA  INDEX+1      ;LINEA SIGUIENTE
415          BCC  FNDEOP       ;SIEMPRE
416          ;
417          ;-----
418          ;TRATAMIENTO DE LOS ERRORES
419          ;DEL D.O.S
420          ;-----
421          ;
422 NDOERR    JSR  CRDO
423          LDA  #!24
424          STA  $23
425          LDX  #!23
426          JSR  VTAB
427          LDX  #$00
428 AFFERR    LDA  MESSDOS,X
429          BEQ  SU1
430          JSR  COUT
431          INX
432          JMP  AFFERR
433 SU1      LDX  #!21

```

434		JSR	VTAB	
435		LDA	#!22	
436		STA	\$23	
437		LDA	#\$00	
438		STA	CH	
439		JSR	CLOSEALL	
440		JMP	CMDLP2+3	
441			;	
442	MESSDOS	ASC	"DOS:	"
443		HEX	00	
444			;	
445	DOSERR	LDY	#\$0A	
446		LDA	(ADL),Y	
447		BEQ	NDOSERR	
448		STA	ADL2	
449		JSR	CRDO	
450		JSR	CRDO	
451		LDA	#!24	
452		STA	\$23	
453		LDX	#!23	
454		JSR	VTAB	
455		LDA	#\$05	
456		STA	CH	
457		LDX	ADL2	
458		SEC		
459		JSR	\$A702	
460	^2	LDA	CH	
461		CMP	#!24	
462		BEQ	>3	

```

463          LDA #" "
464          JSR COUT
465          JMP <2
466 ^3       LDX #!21
467          JSR VTAB
468          LDA #!22
469          STA $23
470          LDA #$00
471          STA CH
472          JMP CMDLP+3
473 ;-----
474 ;SALVAGUARDA EL PROGRAMA
475 ;SOBRE EL FLOPPY
476 ;-----
477 ;
478 ;
479 SAVE:
480          LDX #$00
481 ^1       LDA MESSAVE,X
482          BEQ >2
483          INX
484          JSR COUT
485          JMP <1
486 ^2       LDX #" "
487          JSR INLIN+2
488          LDA #$00
489          STA ALLOW+1
490          JSR OPEN
491          LDA #$01

```

492	STA	ALLOW+1
493	SEC	
494	LDA	\$AF
495	SBC	\$67
496	TAY	
497	LDA	\$B0
498	SBC	\$68
499	STY	ADL2
500	STA	ADL2+1
501	JSR	\$A3E0
502	LDA	#\$04
503	LDY	00
504	STA	(ADL),Y
505	LDA	#\$02
506	INY	
507	STA	(ADL),Y
508	LDA	ADL2
509	LDY	#\$06
510	STA	(ADL),Y
511	LDA	ADL2+1
512	INY	
513	STA	(ADL),Y
514	LDA	#PRGM
515	INY	
516	STA	(ADL),Y
517	LDA	/PRGM
518	INY	
519	STA	(ADL),Y
520	JSR	\$3D6

```

521          BCC >3
522          LDY #$0A
523          LDA (ADL),Y
524          CMP #$06
525          BEQ >3
526          JMP DOSERR
527 ^3       LDY #$00
528          LDA #$02
529          STA (ADL),Y
530          JSR $3D6
531          JMP DOSERR
532 ;
533 ;-----
534 ;CARGA EL PROGRAMA DESDE EL
535 ;FLOPPY
536 ;-----
537 ;
538 LOAD:
539          LDX #$00
540 ^1       LDA MESLOAD,X
541          BEQ >2
542          INX
543          JSR COUT
544          JMP <1
545 ^2       LDX #" "
546          JSR INLIN+2
547          JSR CLOSEALL
548          LDA #$01
549          STA ALLOW+1

```

550	JSR OPEN
551	LDA #\$23
552	AND \$B5C2
553	BEQ FILEERR
554	JSR \$A47A
555	CLC
556	ADC \$67
557	TAX
558	TYA
559	ADC \$68
560	CMP \$74
561	BCS ERRRLONG
562	STA \$B0
563	STA \$6A
564	STX \$AF
565	STX \$69
566	LDX \$67
567	LDY \$68
568	JSR \$A471
569 ^9	JMP NDOSEERR
570 FILEERR:	
571	LDA #\$0D
572	JMP DOSERR+6
573 ERRRLONG:	
574	LDA #\$0E
575	JMP DOSERR+6
576	;-----
577	;DESTRUYE UN PROGRAMA SOBRE
578	;EL FLOPPY

579 ;-----

580 ;

581 DELETE:

582 LDX #\$00

583 ^1 LDA MESSDEL,X

584 BEQ >2

585 JSR COUT

586 INX

587 JMP <1

588 ^2 LDX #" "

589 JSR INLIN+2

590 JSR OPEN

591 LDY #\$00

592 LDA #\$05

593 STA (ADL),Y

594 JSR \$3D6

595 JMP DOSERR

596 ;-----

597 ;ABRE UN FICHERO

598 ;-----

599 OPEN:

600 LDY #\$FF

601 ^0 INY

602 LDA BUF,Y

603 BEQ ERROR

604 CMP #\$20

605 BEQ <0

606 DEY

607 LDX #\$FF

608	^1	INX
609		INY
610		LDA BUF, Y
611		EOR #%10000000
612		STA BUF, X
613		CMP #\$80
614		BNE <1
615		CPX #\$01
616		BEQ ERROR
617		LDA #\$A0
618	^2	STA BUF, X
619		INX
620		CPX #!32
621		BNE <2
622		JSR \$3DC
623		STY ADL
624		STA ADL+1
625		LDY #\$00
626	^3	LDA TABLE, Y
627		STA (ADL), Y
628		INY
629		CPY #\$0A
630		BNE <3
631	ALLOW	LDX #\$01
632		JSR \$3D6
633		BCC >4
634		LDA ALLOW+1
635		BNE ERR
636		LDY #\$0A

```
637          LDA (ADL),Y
638          CMP #$06
639          BNE ERR
640 ^4        LDA ALLOW+1
641          BEQ >6
642          LDY #$07
643          LDA (ADL),Y
644          AND #%01111111
645          CMP #$02
646          BNE >5
647          RTS
648 ^5        JMP FILEERR
649 ^6        LDY #$07
650          LDA (ADL),Y
651          CMP #$02
652          BNE >7
653          RTS
654 ^7        CMP #$82
655          BNE <5
656          LDY #$0A
657          TYA
658          STA (ADL),Y
659          JMP DOSERR
660 ERR:
661          JMP DOSERR
662 ERROR:
663          JMP CMDLP
664 TABLE:
665          HEX 01000000000106020002
```

```

666 ;-----
667 ;BORRA EL PROGRAMA QUE ESTA
668 ;EN MEMORIA CENTRAL
669 ;-----
670 ;
671 NEW:
672     LDX #$00
673 ^1   LDA MESSNEW,X
674     BEQ >2
675     JSR COUT
676     INX
677     JMP <1
678 ^2   JSR KEYIN
679     JSR COUT
680     CMP #$D9
681     BEQ >3
682     JMP CMDLP
683 ^3   LDA #$04
684     STA VARTAB
685     STA PRGEND
686     LDA #$08
687     STA VARTAB+1
688     STA PRGEND+1
689     LDA #$00
690     STA $801
691     STA $802
692     JMP CMDLP
693 ;-----
694 ;SUBROUTINA EJECUTADA CUANDO

```

```

695 ;SE APRIETA RESET
696 ;-----
697 ;
698 RESET:
699         JSR $FC58
700         JMP INIT
701 ;-----
702 ;RESTAURA LOS ANTIGUOS VECTORES
703 ;DE E/S ANTES DE VOLVER AL
704 ;BASIC
705 ;-----
706 QUIT:
707         LDA #$9E
708         STA $9D03
709         STA $9D05
710         STA $37
711         STA $39
712         LDA #$81
713         STA $9D02
714         STA $38
715         LDA #$BD
716         STA $9D04
717         STA $36
718         LDA #!24
719         STA $23
720         LDX #$00
721 ^1      LDA MESSQUIT,X
722         BEQ >2
723         JSR COUT

```

```

724          INX
725          JMP <1
726 ^2       JSR $FC42
727          DEC CH
728          JSR CRDO
729          LDA #$D0
730          STA $3F2
731          LDA #$03
732          STA $3F3
733          JSR $FB6F
734          JMP $3D0
735 ;
736 ;-----
737 ;TITULO DEL PROGRAMA
738 ;-----
739 MES1:
740          ASC "GLOBAL APPLESOFT PROGRAM EDITOR."
741          HEX 8D
742          ASC "          PAR HUIBERT AALBERS"
743          HEX 8D
744          HEX 00
745 ;-----
746 ;OTROS MENSAJES Y DATOS
747 ;-----
748 VECT:
749          HEX 4C0070
750 MESSAVE:
751          ASC "SAVE:"
752          HEX 00

```

753 MESSLOAD:

754 ASC ``LOAD:``

755 HEX 00

756 MESSDEL:

757 ASC ``DELETE:``

758 HEX 00

759 MESSNEW:

760 ASC "NEW: OK? (Y/N): "

761 HEX 00

762 MESSQUIT:

763 ASC "& RAMENE A L´EDITEUR."

764 HEX 00

765 ;-----

766 ;TRATAMIENTO DE LOS COMANDOS

767 ;DETERMINA EL COMANDO Y LO

768 ;EJECUTA

769 ;-----

770 ;

771 PARSE2:

772 RTS

773 END

Archivo PHILIPS.2 (líneas 1-999)

```
1 ;
2 *****
3 *                               *
4 * APPLESOFT PROGRAM EDITOR *
5 *                               *
6 *           BY           *
7 *                               *
8 *     HUIBERT AALBERS     *
9 *                               *
10 *****
11 ;
12 ;-----
13 ;DEFINICION DE LAS RUTINAS DEL
14 ;APPLESOFT EN ROM O DEL D.O.S
15 ;-----
16 BUF      EQU  $200      ;BUFFER UTILIZADO POR INLIN
17 AMPER    EQU  $3F5      ;VECTOR DEL &
18 PRGM     EQU  $801      ;PRINCIPIO DEL PROGRAMA
19 CMDLP    EQU  $7069     ;BUCLE PRINCIPAL DEL PROGRAMA
20 CMDLP2   EQU  $707F
21 PROCLN   EQU  $709F     ;ENTRA UNA LINEA DE PROGRAMA
22 BUF2     EQU  $9400     ;BUFFER SECUNDARIO UTILIZADO POR REPEAT
23 BUF3     EQU  $9500     ;BUFFER UTILIZADO POR SEARCH
24 CLOSEALL EQU  $A316     ;CIERRA TODOS LOS FICHEROS
25 CONVERT  EQU  $A1B9     ;CONVIERTE ASCII EN HEXADECIMAL
26 LININDEX EQU  $AA5D     ;REGISTRO UTILIZADO POR $A1B9
```


27	TKNTBL	EQU	\$D0D0	;TABLA DE COMANDOS DEL BASIC
28	BLTU	EQU	\$D393	;MUEVE 256 OCTETOS HACIA ARRIBA
29	INLIN	EQU	\$D52C	;ENTRA UNA LINEA HACIA EL BUFFER
30	PARSE	EQU	\$D559	;CODIFICA UNA LINEA ENTRADA POR INLIN
31	FNDLIN	EQU	\$D61A	;BUSCA UNA LINEA DE PROGRAMA
32	INITPTRS	EQU	\$D665	
33	LINGET	EQU	\$DA0C	;PONE EN LINNUM EL NO DE LINEA CORRIENTE
34	CRDO	EQU	\$DAFB	;IMPRIME UN CARRIAGE RETURN
35	OUTSP	EQU	\$DB57	;IMPRIME UN ESPACIO
36	OUTDO	EQU	\$DB5C	;IMPRIME EL CARACTER CONTENIDO EN A
37	ISLETC	EQU	\$E07D	;COMPRUEBA SI A ES UNA LETRA (A-Z)
38	LINPRT	EQU	\$ED24	;IMPRIME X,A
39	VTAB	EQU	\$F25A	;TABULACION VERTICAL
40	PRBL2	EQU	\$F94A	;IMPRIME X ESPACIOS
41	UP	EQU	\$FC1A	;SUBE DE UNA LINEA EN LA PANTALLA
42	HOME	EQU	\$FC58	;BORRA LA PANTALLA
43	KEYIN	EQU	\$FD0C	;ENTRA UN CARACTER
44	KEYIN2	EQU	\$FD1B	;ESPERA QUE SEA PULSADA UNA TECLA
45	GETLN	EQU	\$FD6A	
46	COUT	EQU	\$FDED	;IMPRIME EL CARACTER CONTENIDO EN A
47	PROUT	EQU	\$FE95	;INICIALIZA EL SLOT (A)
48				;
49				;-----
50				;DEFINICION DE LAS DIRECCIONES
51				;EN PAGINA CERO
52				;-----
53				;
54	POSCUR	EPZ	\$06	;POS. CURSOR TRAS LINPRT DURANTE LIST

55	ALLOWPTR	EPZ	\$07	
56	ADL	EPZ	\$08	;VECTOR ENTRADA DEL FILE MANAGER
57	CRDNB	EPZ	\$0F	
58	LINE	EPZ	\$18	;REGISTRO TEMPORAL SOBRE 2 OCTETOS
59	CURLIGNE	EPZ	\$1A	;CONTIENE LA LINEA CORRIENTE
60	ADRNXT	EPZ	\$1C	;LOW BYTE DIRECCION LINEA SIGUIENTE
61	LSTLIN	EPZ	\$1D	;DIRECCION ULTIMA LINEA DURANTE BOTTOM
62	INDEX2	EPZ	\$1F	;REGISTRO TEMPORAL.GUARDA X DURANTE PRLET
63	CH	EPZ	\$24	;POSICION DEL CURSOR (HORIZONTAL)
64	CV	EPZ	\$25	;POSICION VERTICAL DEL CURSOR
65	PROMPT	EPZ	\$33	;CONTIENE EL CARACTER ">"
66	LINNUM	EPZ	\$50	;CONTIENE EL NUMERO DE LINEA TRAS LINGET
67	LIST2	EPZ	\$54	;FLAG PARA HACE PRBUF ANTES DE CMDLP
68	ESCFLG	EPZ	\$55	;FLAG QUE CONTROLA EL MODO "ESC"
69	INDEX	EPZ	\$5E	;REGISTRO TEMPORAL PARA MOVER MEMORIA
70	DEST	EPZ	\$60	
71	TXTTAB	EPZ	\$67	;DIRECCION PRINCIPIO DEL PROGRAMA BASIC
72	VARTAB	EPZ	\$69	;DIRECCION PRINCIPIO VARIABLES SIMPLES
73	STREND	EPZ	\$6D	;DIRECCION FIN VARIABLES DIMENSIONADAS
74	FRETOP	EPZ	\$6F	;DIRECCION FIN VARIABLES ALFANUMERICAS
75	MEMSIZ	EPZ	\$73	;DIRECCION FIN DE LA MEMORIA UTILIZABLE
76	CURLIN	EPZ	\$75	;FLAG (CONTIENE FF EN MODO INMEDIATO)
77	FORPNT	EPZ	\$85	;NUEVA DIRECCION DE UNA CADENA DESPLAZADA
78	HIGHDS	EPZ	\$94	;REGISTRO TEMPORAL SOBRE 5 OCTETOS
79	HIGHTR	EPZ	\$96	;COMPONE CON HIGHDS EL REGISTRO INTERNO TEMPS1
80	LOWTR	EPZ	\$9B	;REGISTRO TEMPORAL SOBRE 5 OCTETOS
81	DSCTMP	EPZ	\$9D	;REGISTRO TEMPORAL SOBRE 6 OCTETOS
82	PRGEND	EPZ	\$AF	;FINAL DE LA ZONA PROGRAMA BASIC

83	CHRGET	EPZ	\$B1	;SUBROUTINA QUE LEE EL SIGIENTE CARACTER
84	;			;DEL PROGRAMA BASIC
85	TXTPTR	EPZ	\$B8	;DIRECCION CARACTER OBTENIDO POR CHRGET
86	ERRFLG	EPZ	\$D8	;FLAG QUE INDICA SI ONERR ESTA ACTIVO
87	INC	EPZ	\$CE	;VALOR DEL INCREMENTO EN MODO AUTO
88	LINDEX	EPZ	\$CF	;COPIA DE LININDEX ANTES DE \$A1B9
89	ADL2	EPZ	\$F9	;REGISTRO TEMPORAL SOBRE 2 OCTETOS
90	SAVEA	EPZ	\$FB	;REGISTRO TEMPORAL SOBRE 1 OCTETO
91	COM	EPZ	\$FC	;REGISTRO TEMPORAL SOBRE 1 OCTETO
92	FIRST	EPZ	\$FD	;REGISTRO TEMPORAL SOBRE 1 OCTETO
93	TEMP	EPZ	\$FD	
94	AUTOFLG	EPZ	\$FF	;FLAG RELATIVO AL MODO AUTO
95	;			
96		ORG	\$746E	
97		OBJ	\$800	
98	;			
99	;			-----
100	;			TRATAMIENTO DE LOS COMANDOS
101	;			DETERMINA EL COMANDO Y LO
102	;			EJECUTA
103	;			-----
104	;			
105	PARSE2:			
106		LDY	#\$00	
107		STY	COM	
108		LDX	#\$FF	
109	^1	INX		
110		LDA	BUF,X	

111	CMP	#\$20
112	BEQ	<1
113	LDA	BUF,X
114	AND	##00111111
115	STA	FIRST
116	TXA	
117	STA	SAVEA
118	TAY	
119	LDX	#\$00
120 ^2	LDA	COMTBL,X
121	BNE	>3
122	INC	COM
123 ^3	LDA	COMTBL,X
124	INX	
125	CMP	#\$80
126	BCS	<2
127	CMP	FIRST
128	BEQ	>4
129	CMP	#\$1D
130	BEQ	ERR
131	BNE	<2
132 ^4	DEX	
133	INY	
134	INX	
135	LDA	BUF,Y
136	BEQ	>5
137	EOR	##10000000
138	CMP	#" "

139		BEQ	>5
140		CMP	COMTBL,X
141		BNE	SUIT3
142		INX	
143		LDA	COMTBL,X
144		BNE	<4
145	^5	LDA	COM
146		ASL	
147		TAX	
148		INX	
149		LDA	TBLADR,X
150		PHA	
151		DEX	
152		LDA	TBLADR,X
153		PHA	
154		LDA	COM
155		CMP	#\$0F
156		BEQ	FUNJMP
157		LDY	#\$00
158	DEPBUF	LDA	BUF,Y
159		STA	BUF2,Y
160		INY	
161		BNE	DEPBUF
162	FUNJMP	RTS	
163	SUIT3	LDY	SAVEA
164		JMP	<2
165	ERR:		
166		LDA	#\$01

```

167          JMP  ERREUR
168 ;
169 ;-----
170 ;SOUS-PROGRAMME PERMETTANT DE
171 ;LISTER UNE LIGNE DE PROGRAMME
172 ;EN 40 COLONNES.
173 ;ENTREE: # DE LIGNE DANS LINNUM
174 ;SORTIE: IMPRESSION DE LA LIGNE
175 ;-----
176 ;
177 LIST:
178          JSR  FNDLIN      ;BUSCA DIRECCION LINEA (LINNUM)
179          BCC  NOLN       ;SI NO EXISTE, ERROR
180 ;-----
181 ;LIST+3 LISTE LA LIGNE POINTEE
182 ;PAR LOWTR.
183 ;-----
184          LDY  #$00       ;INICIALIZACION: LAS COMILLAS ESTAN
185          STY  $4D        ;CERRADAS
186          LDY  #$02       ;CARGA EN A EL PRIMER OCTETO DEL
187          LDA  (LOWTR),Y   ;NUMERO DE LINEA
188          STA  CURLIGNE    ;LO SALVAGUARDA
189          TAX                ;Y LO PASA AL REGISTRO X.
190          INY                ;CARGA EN A EL SEGUNDO OCTETO DEL
191          LDA  (LOWTR),Y   ;NUMERO DE LINEA
192          STA  CURLIGNE+1  ;Y LO SALVAGUARDA
193          STY  FORPNT
194          LDY  ALLOWPRT

```

195	CPY	#\$02	
196	BNE	>1	
197	LDX	#\$00	
198	STX	ALLOWPRT	
199	JMP	>2	
200	^1	JSR	LINPRT
201	NOP		;SALVAGUARDA EN POSCUR LA COLUNA EN LA
202	^2	STY	POSCUR ;QUE ESTA EL CURSOR TRAS IMPRIMIR EL
203	STY	INDEX2	;NUMERO DE LINEA
204	;		
205	;	PREND	CARACTERE OU TOKEN
206	;		
207	LDA	#" "	
208	LSTLN:		
209	LDY	FORPNT	
210	PRCHR:		
211	CMP	#\$22	
212	BNE	>1	
213	PHA		
214	LDA	#\$FF	
215	EOR	\$4D	
216	STA	\$4D	
217	PLA		
218	^1	CMP	#\$3A
219	BNE	>3	
220	PHA		
221	LDA	\$4D	
222	BNE	>2	

223		LDA	#\$8D	
224		JSR	PRLET	
225		LDX	POSCUR	
226		BEQ	>2	
227		JSR	PRBL2	
228	^2	PLA		
229	^3	JSR	PRLET	
230		INY		
231		LDA	(LOWTR),Y	
232		BNE	PROCHR	
233		BIT	\$C000	
234		BPL	>4	
235		LDA	\$C000	
236		BIT	\$C010	
237		CMP	#" "	
238		BEQ	PAUSE	
239		CMP	\$98	;CTRL-C
240		BEQ	STOP	
241	^4	LDX	INDEX2	;SI NO, PONE UN 0 AL FINAL DE BUF Y
242		LDA	#\$00	
243		STA	\$200,X	
244		RTS		
245	NOLN	LDA	#\$00	
246		JMP	ERREUR	
247	STOP	JSR	CRDO	
248		JMP	CMDLP	
249	PAUSE	TYA		
250		PHA		


```

251          LDA  #"  "
252          JSR  KEYIN2
253          PLA
254          TAY
255          CMP  #\$98
256          BEQ  STOP
257          RTS
258  ;
259  ;PREND UN CARACTERE DANS LA TABLE
260  ;
261  KEYCHR:
262          INY
263          BNE  S1
264          INC  DSCTMP+1
265  S1:
266          LDA  (DSCTMP),Y
267          RTS
268  ;
269  ;IMPRIME CARACTERE OU MOT-CLE.
270  ;
271  PROCHR:
272          BPL  PRCHR
273          SEC
274          SBC  #\$7F
275          TAX
276          STY  FORPNT
277          LDY  #TKNTBL
278          STY  DSCTMP

```

279	LDY	/TKNTBL-\$100
280	STY	DSCTMP+1
281	LDY	#\$FF
282	NXKEY:	
283	DEX	
284	BEQ	PRKEY
285	S2:	
286	JSR	KEYCHR
287	BPL	S2
288	BMI	NXKEY
289	PRKEY:	
290	LDA	#" "
291	JSR	PRLET
292	JSR	KEYCHR
293	BMI	S4
294	JSR	PRLET
295	BNE	PRKEY+5
296	S4:	
297	JSR	PRLET
298	LDA	#" "
299	JMP	LSTLN
300	PRLET:	
301	STA	\$4C
302	CMP	#\$8D
303	BNE	PRLET2
304	LDA	ALLOWPRT
305	BEQ	PRLET2-1
306	JSR	CRDO

307	LDA	\$4C
308	RTS	
309	PRLET2	TXA
310	PHA	
311	TYA	
312	PHA	
313	LDX	INDEX2
314	LDA	\$4C
315	AND	#%01111111
316	STA	BUF,X
317	INX	
318	STX	INDEX2
319	LDX	ALLOWPRT
320	BEQ	END2
321	PLA	
322	TAY	
323	PLA	
324	TAX	
325	LDA	\$4C
326	JSR	OUTDO
327	RTS	
328	END2	PLA
329	TAY	
330	PLA	
331	TAX	
332	LDA	\$4C
333	RTS	
334	;	

335 ;-----

336 ;SOUS-PROGRAMME DE TRAITEMENT

337 ;D'ERREUR

338 ;-----

339 ;

340 ERREUR:

341 ASL

342 TAY

343 LDA ERRTBL, Y

344 STA ADL

345 INY

346 LDA ERRTBL, Y

347 STA ADL+1

348 LDY #\$00

349 ^1 LDA (ADL), Y

350 BEQ >2

351 JSR COUT

352 INY

353 BNE <1

354 ^2 JMP CMDLP

355 ;

356 ESCODES:

357 TAY

358 INY

359 TXA

360 PHA

361 LDX #\$00

362 ^1 LDA ESCTBL, X

363	BEQ	>2
364	INX	
365	BNE	<1
366	^2	INX
367	DEY	
368	BNE	<1
369	PLA	
370	TAY	
371	DEX	,
372	^3	INX
373	LDA	ESCTBL,X
374	BEQ	>4
375	STA	BUF,Y
376	JSR	COUT
377	INY	
378	BNE	<3
379	^4	TYA
380	TAX	
381	LDA	#\$00
382	STA	ESCFLG
383	LDA	#" "
384	RTS	
385	;	
386	ESCTBL	HEX 00
387	ASC	"ABS("
388	HEX	00
389	ASC	"HPLOT"
390	HEX	00

391	ASC	"CLEAR"
392	HEX	00
393	ASC	"DATA"
394	HEX	00
395	ASC	"END"
396	HEX	00
397	ASC	"FOR"
398	HEX	00
399	ASC	"GOTO"
400	HEX	00
401	ASC	"HOME"
402	HEX	00
403	ASC	"INPUT"
404	HEX	00
405	ASC	"CHR\$("
406	HEX	00
407	ASC	"RIGHT\$("
408	HEX	00
409	ASC	"LEFT\$("
410	HEX	00
411	ASC	"MID\$("
412	HEX	00
413	ASC	"NEXT"
414	HEX	00
415	ASC	"POKE"
416	HEX	00
417	ASC	"PEEK"
418	HEX	00

419	ASC	"CALL"
420	HEX	00
421	ASC	"RETURN"
422	HEX	00
423	ASC	"STEP"
424	HEX	00
425	ASC	"TEXT"
426	HEX	00
427	ASC	"HTAB"
428	HEX	00
429	ASC	"VTAB"
430	HEX	00
431	ASC	"DRAW"
432	HEX	00
433	ASC	"XDRAW"
434	HEX	00
435	ASC	"HCOLOR="
436	HEX	00
437	ASC	"RESTORE"
438	HEX	00
439		;
440	ERRTBL	ADR NOLINE
441		ADR SORRY
442		ADR ERRBOT
443		ADR NOPRGM
444		ADR BADARG
445		ADR ERRHEX
446		;

447 COMTBL:

448	HEX	0E
449	ASC	"EXT"
450	HEX	000C
451	ASC	"INE"
452	HEX	0004
453	ASC	"ELETE"
454	HEX	0005
455	ASC	"DIT"
456	HEX	0010
457	ASC	"RINT"
458	HEX	000E
459	ASC	"OCOLUMN"
460	HEX	0003
461	ASC	"OLUMN"
462	HEX	0001
463	ASC	"UTON"
464	HEX	0001
465	ASC	"UTOFF"
466	HEX	0014
467	ASC	"OP"
468	HEX	0002
469	ASC	"OTTOM"
470	HEX	0013
471	ASC	"EARCH"
472	HEX	000D
473	ASC	"ODIFY"
474	HEX	0001

475	ASC	"PPEND"
476	HEX	0010
477	ASC	"P"
478	HEX	0012
479	ASC	"EPEAT"
480	HEX	0010
481	ASC	"R#"
482	HEX	0008
483	ASC	"\$"
484	HEX	0008
485	ASC	"ELP"
486	HEX	001D
487	TBLADR:	
488	ADR	NEXT-1
489	ADR	LINE1-1
490	ADR	DELETE-1
491	ADR	EDIT-1
492	ADR	PRINT-1
493	ADR	NOCOL-1
494	ADR	COLUMN-1
495	ADR	AUTO-1
496	ADR	AUTOFF-1
497	ADR	TOP-1
498	ADR	BOTTOM-1
499	ADR	SEARCH-1
500	ADR	MODIFY-1
501	ADR	APPEND-1
502	ADR	PP-1

503	ADR	REPEAT-1
504	ADR	PR-1
505	ADR	DOLLAR-1
506	ADR	HELP-1
507	HEX	00
508	NEXT:	
509	JSR	CRDO
510	JSR	LOOK
511	CLC	
512	LDX	#\$00
513	^1	INX
514	LDA	BUF,X
515	BEQ	NEXTMAS
516	CMP	#"-"
517	BNE	<1
518	JMP	NEXTMNS
519	NEXTMAS:	
520	JSR	LOOK
521	CLC	
522	LDA	CURLIGNE
523	STA	LINNUM
524	LDA	CURLIGNE+1
525	STA	LINNUM+1
526	JSR	FNDLIN
527	^1	JSR
528	LDA	\$44
529	BNE	<1
530	LDY	#\$02

531	LDA	(LOWTR),Y	
532	STA	CURLIGNE	
533	STA	LINNUM	
534	INY		
535	LDA	(LOWTR),Y	
536	STA	CURLIGNE+1	
537	STA	LINNUM+1	
538	JSR	FNDLIN	
539	BCC	>2	
540	JSR	LIST	
541	JSR	CRD0	
542	JMP	CMDLP	
543	^2	JMP	BOTTOM
544	NXTLIN:		
545	DEC	\$44	
546	LDY	#\$00	
547	LDA	(LOWTR),Y	
548	STA	LINNUM	
549	INY		
550	LDA	(LOWTR),Y	
551	STA	LOWTR+1	
552	LDA	LINNUM	
553	STA	LOWTR	
554	RTS		
555	NEXTMNS:		
556	INX		
557	STX	LININDEX	
558	JSR	CONVERT	

559	LDA	CURLIGNE
560	STA	LINNUM
561	LDA	CURLIGNE+1
562	STA	LINNUM+1
563	^1	JSR FNDLIN
564	JSR	NXTLIN2
565	BCC	>2
566	LDA	\$44
567	BNE	<1
568	LDA	LINNUM
569	STA	CURLIGNE
570	LDA	LINNUM+1
571	STA	CURLIGNE+1
572	LDA	#\$01
573	STA	ALLOWPRT
574	JSR	LIST
575	JSR	CRDO
576	JMP	CMDLP
577	^2	JMP TOP
578	NXTLIN2:	
579	CLC	
580	DEC	\$44
581	DEC	LOWTR+1
582	LDY	#\$FF
583	^1	DEY
584	CPY	#\$00
585	BEQ	>4
586	LDA	(LOWTR),Y

587	CMP	LOWTR
588	BEQ	>2
589	BNE	<1
590 ^2	DEY	
591	LDA	(LOWTR),Y
592	BEQ	>3
593	INY	
594	JMP	<1
595 ^3	INY	
596	INY	
597	INY	
598	LDA	(LOWTR),Y
599	STA	LINNUM
600	INY	
601	LDA	(LOWTR),Y
602	STA	LINNUM+1
603	SEC	
604	RTS	
605 ^4	CLC	
606	RTS	
607	LINE1:	
608	JSR	LOOK
609	JSR	CRDO
610	LDA	\$44
611	STA	\$50
612	LDA	\$45
613	STA	\$51
614	JSR	LIST

615		JSR	CRDO
616		JMP	CMDLP
617	LOOK:		
618		LDA	#\$00
619		STA	\$44
620		STA	\$45
621		JSR	POSBUF
622		LDX	#\$FF
623	BLE	INX	
624		LDA	BUF,X
625		CMP	#" "
626		BEQ	BLE
627	BLE2	INX	
628		LDA	BUF,X
629		BEQ	>3-3
630		CMP	#" "
631		BNE	BLE2
632	^2	INX	
633		LDA	BUF,X
634		CMP	#" "
635		BEQ	<2
636		LDA	BUF,X
637		BEQ	>3-3
638		CMP	#" -"
639		BEQ	<2
640		CMP	#\$B0
641		BCC	>4
642		CMP	#\$BA

643	BCS	>4	
644	STX	LININDEX	
645	STX	LINDEX	
646	JSR	CONVERT	
647	CLC		
648	LDA	\$44	
649	BNE	>3	
650	INC	\$44	;SI EL ARGUMENTO ES 0,LO CAMBIA EN 1
651	SEC		
652	^3	RTS	
653	^4	LDA #\$04	;EL ARGUMENTO NO ES UN NUMERO.
654	JMP	ERREUR	
655	DELETE:		
656	JSR	LOOK	
657	BCC	>1	
658	LDA	#\$01	
659	STA	\$44	
660	^1	LDA	CURLIGNE
661	STA	LINNUM	
662	LDA	CURLIGNE+1	
663	STA	LINNUM+1	
664	JSR	FNDLIN	
665	LDA	LOWTR	
666	STA	ADL	
667	LDA	LOWTR+1	
668	STA	ADL+1	
669	LDA	\$36	
670	PHA		

671	LDA	\$37	
672	PHA		
673	LDA	#PRLIGNE	
674	STA	\$36	
675	LDA	/PRLIGNE	
676	STA	\$37	
677	LDA	#\$00	
678	STA	\$FE	
679	LDY	#\$02	
680	LDA	(LOWTR),Y	
681	TAX		
682	INY		
683	LDA	(LOWTR),Y	
684	JSR	LINPRT	
685	LDY	\$FE	
686	LDA	#\$00	
687	STA	BUF,Y	
688	PLA		
689	STA	\$37	
690	PLA		
691	STA	\$36	
682	LDA	#\$02	
693	STA	SAVEA	
694	LDX	#\$FF	
695	LDY	#\$01	
696	JMP	CMDLP2+8	
697	ENTRY2	LDY	#\$00
698	LDA	(ADL),Y	

699	BNE	>0
700	INY	
701	LDA	(ADL),Y
702	BNE	>0
703	JMP	BOTTOM
704	LDY	#\$02
705	LDA	(ADL),Y
706	STA	CURLIGNE
707	INY	
708	LDA	(ADL),Y
709	STA	CURLIGNE+1
710	DEC	\$44
711	LDA	\$44
712	BNE	<1
713	LDA	CURLIGNE
714	STA	LINNUM
715	LDA	CURLIGNE+1
716	STA	LINNUM+1
717	JSR	LIST
718	JMP	CMDLP
719	EDIT:	
720	JSR	CRDO
721	LDA	#\$00
722	STA	COM
723	JSR	LOOK
724	BCS	>1
725	LDA	\$44
726	STA	CURLIGNE

727	LDA	\$45
728	STA	CURLIGNE+1
729 ^1	LDA	CURLIGNE
730	STA	LINNUM
731	LDA	CURLIGNE+1
732	STA	LINNUM+1
733	LDA	#\$00
734	STA	ALLOWPRT
735	JSR	LIST
736	CPX	#\$FF
737	BEQ	>0
738	LDA	#\$20
739	STA	BUF,X
740	INX	
741	LDA	#\$00
742	STA	BUF,X
743 ^0	LDY	#\$00
744 ^2	LDA	(\$28),Y
745	CMP	#" "
746	BEQ	>3
747	AND	01111111
748	STA	BUF,Y
749	INY	
750	JMP	<2
751 ^3	LDY	#\$01
752	STY	ALLOWPRT
753	DEY	
754	STY	CH

755	JSR	PRBUF
756	LDA	POSCUR
757	STA	INDEX
758	JSR	CURPOS
759	LDA	#" "
760	LDX	#!39
761 ^0	STA	\$7D0,X
762	DEX	
763	BNE	<0
764	LDX	#\$00
765 ^0	LDA	MESSEEDIT,X
766	BEQ	>4
767	STA	\$7D0,X
768	INX	
769	BNE	<0
770 ^4	LDA	#\$00
771	STA	ESCFLG
772	JSR	STAT
773	JSR	KEYIN
774	CMP	#\$95
775	BEQ	CTRLU
776	CMP	#\$88
777	BEQ	CTRLH
778	PHA	
779	LDA	COM
780	BNE	TYPE
781	PLA	
782	CMP	#\$9B

783	BNE	>5
784	LDA	#\$01
785	STA	COM
786	JMP	<4
787 ^5	CMP	#"I"
788	BNE	>5
789	JMP	INSCHR
790 ^5	CMP	#"F"
791	BNE	>5
792	JMP	FINDCHR
793 ^5	CMP	#"D"
794	BNE	>5
795	JMP	CHRDEL
796 ^5	CMP	#"B"
797	BNE	>5
798	JMP	BEGIN
799 ^5	CMP	#"E"
800	BNE	>5
801	JMP	END
802 ^5	CMP	#"C"
803	BNE	>5
804	JMP	CUT
805 ^5	CMP	#\$98
806	BNE	>5
807	JMP	CTRLX
808 ^5	CMP	#\$8D
809	BNE	<4
810	JMP	CR

811	CTRLU	LDX	INDEX
812		INX	
813		LDA	BUF,X
814		BEQ	<4
815		STX	INDEX
816		JSR	CURPOS
817		JMP	<4
818	CTRLH	LDX	INDEX
819		CPX	POSCUR
820		BEQ	<4
821		DEX	
822		STX	INDEX
823		JSR	CURPOS
824		JMP	<4
825	TYPE	PLA	
826		CMP	#\$9B
827		BNE	>1
828		LDA	#\$00
829		STA	COM
830		JMP	<4
831	^1	CMP	#\$8D
832		BNE	>2
833		JMP	CR
834	^2	AND	##01111111
835		LDY	INDEX
836		STA	BUF,Y
837		CPY	#\$FA
838		BEQ	>3

839	INY
840	LDA BUF, Y
841	BNE >3-2
842	LDA #\$20
843	STA BUF, Y
844	INY
845	LDA #\$00
846	STA BUF, Y
847	DEY
848	STY INDEX
849 ^3	JSR PRBUF2
850	JSR CURPOS
851	JMP <4
852 INSCHR	LDX #\$FF
853 ^1	INX
854	LDA BUF, X
855	BEQ >2
856	CPX #\$FA
857	BNE <1
858	JMP <4
859 ^2	LDY INDEX
860	LDA BUF, Y
861	ORA #%10000000
862	STA BUF, Y
863	INX
864	TXA
865	TAY
866	INY

867	^3	DEX
868		DEY
869		LDA BUF,X
870		CMP #\$80
871		BCS >0
872		STA BUF,Y
873		JMP <3
874	^0	AND #%01111111
875		STA BUF,Y
876		LDA #\$20
877		STA BUF,X
878		JSR PRBUF2
879		JSR CURPOS
880		JMP <4
881	CHRDEL	LDY INDEX
882		LDA BUF,Y
883		BNE >1
884		JMP <4
885	^1	TYA
886		TAX
887		INX
888	^2	LDA BUF,X
889		BEQ >3
890		STA BUF,Y
891		INX
892		INY
893		BNE <2
894	^3	STA BUF,Y

895		JSR	PRBUF2
896		JSR	CURPOS
897		JMP	<4
898	FINDCHR	JSR	KEYIN
899		AND	##01111111
900		LDY	INDEX
901	^1	INY	
902		LDX	BUF, Y
903		BEQ	>2
904		CMP	BUF, Y
905		BNE	<1
906		STY	INDEX
907		JSR	PRBUF2
908		JSR	CURPOS
909	^2	JMP	<4
910	END	LDY	INDEX
911	^1	INY	
912		LDA	BUF, Y
913		BEQ	>2
914		JMP	<1
915	^2	DEY	
916		STY	INDEX
917		JSR	PRBUF2
918		JSR	CURPOS
919		JMP	<4
920	BEGIN	LDY	POSCUR
921		STY	INDEX
922		JSR	PRBUF2

923		JSR	CURPOS
924		JMP	<4
925	CUT	LDY	INDEX
926		LDA	#\$00
927		STA	BUF,Y
928		JSR	PRBUF2
929		JSR	CURPOS
930		JMP	<4
931	CTRLX	LDA	#\$00
932		STA	INDEX
933		JSR	CURPOS
934		JMP	EDIT+20
935	CR	JSR	PRBUF2
936		JSR	CRDO
937		LDX	#!39
938		LDA	#" "
939	^0	STA	\$7D0,X
940		DEX	
941		CPX	#\$FF
942		BNE	<0
943		LDX	#\$00
944	^1	LDA	MESSDOS,X
945		BEQ	>2
946		STA	\$7D0,X
947		INX	
948		BNE	<1
949	^2	LDY	#\$01
950		LDX	#\$FF

951		STY	TXTPTR+1
952		STX	TXTPTR
953		JSR	CHRGET
954		JMP	PROCLN
955	STAT	LDA	COM
956		BNE	>3
957		LDX	#\$00
958	^1	LDA	MESSCOM,X
959		BEQ	>2
960		STA	\$7D6,X
961		INX	
962		BNE	<1
963	^2	RTS	
964	^3	LDX	#\$00
965		LDA	MESSTYPE,X
966		BEQ	>4
967		STA	\$7D6,X
968		INX	
969		BNE	<3+2
970	^4	RTS	
971	PRBUF2	LDY	#\$00
972		STY	CH
973	^1	LDA	(\$28),Y
974		CMP	#">"
975		BEQ	>2
976		JSR	UP
977		JMP	<1
978	^2	INY	

```

979          LDA  ($28),Y
980          CMP  #" "
981          BEQ  <2
982          CMP  #"E"
983          BNE  >3
984          INC  CV
985          INC  CV
986          JSR  $FC22
987          JSR  PRBUF
988          RTS
989  ^3      JSR  UP
990          JMP  PRBUF2
991          RTS
992 PRINT:
993          LDA  #$01
994          STA  ALLOWPRT
995          JSR  CRDO
996          JSR  LOOK
997          BCC  >1
998          LDA  #$FF
999          STA  $44

```

Archivo PHILIPS.2 (líneas 1000-1830)

1000	STA	\$45
1001	^1	LDA CURLIGNE
1002	STA	LINNUM
1003	LDA	CURLIGNE+1
1004	STA	LINNUM+1
1005	^1	JSR PRINTUNA
1006	BCC	>3
1007	LDA	\$44
1008	BNE	<1
1009	LDA	\$45
1010	BEQ	>2
1011	DEC	\$45
1012	JMP	<1
1013	^2	JMP CMDLP
1014	^3	JMP BOTTOM
1015	PRINTUNA	:
1016	DEC	\$44
1017	LDA	LINNUM
1018	STA	CURLIGNE
1019	LDA	LINNUM+1
1020	STA	CURLIGNE+1
1021	JSR	FNDLIN
1022	BCC	>1
1023	JSR	LIST
1024	CLC	
1025	INY	

1026		LDA	(LOWTR),Y
1027		BNE	>0
1028		INY	
1029		LDA	(LOWTR),Y
1030		BNE	>0+1
1031		BEQ	>1
1032	^0	INY	
1033		INY	
1034		LDA	(LOWTR),Y
1035		STA	LINNUM
1036		INY	
1037		LDA	(LOWTR),Y
1038		STA	LINNUM+1
1039		JSR	CRDO
1040		SEC	
1041	^1	RTS	
1042	COLUMN:		
1043		LDX	#\$00
1044	BCLE	LDA	TBLCOL,X
1045		STA	\$750,X
1046		INX	
1047		CPX	#!40
1048		BNE	BCLE
1049		JMP	CMDLP
1050	NOCOL:		
1051		LDX	#\$00
1052		LDA	#" "
1053	BCLE2	STA	\$750,X

1054		INX
1055		CPX #!40
1056		BNE BCLE2
1057		JMP CMDLP
1058	AUTO:	
1059		LDA # \$0A
1060		STA INC
1061		JSR LOOK
1062		BCS >1
1063		LDA \$44
1064		STA LINE
1065		LDA \$45
1066		STA LINE+1
1067		BNE CHKCOM-2
1068		LDA LINE
1069		BNE CHKCOM-2
1070	^1	LDA # \$0A
1071		STA LINE
1072		LDX # \$00
1073	CHKCOM	INX
1074		LDA BUF,X
1075		BEQ NOARG
1076		CMP #", "
1077		BNE CHKCOM
1078		JSR BLE2+10
1079		LDA \$44
1080		STA INC
1081	NOARG	SEC

1082	LDA	LINE
1083	SBC	INC
1084	STA	LINE
1085	LDA	LINE+1
1086	SBC	#\$00
1087	STA	LINE+1
1088	LDA	#\$01
1089	STA	AUTOFLG
1090	JMP	CMDLP
1091	AUTOFF:	
1092	LDA	#\$00
1093	STA	AUTOFLG
1094	STA	LINE+1
1095	LDA	#!20
1096	STA	LINE
1097	LDA	#\$0A
1098	STA	INC
1099	JMP	CMDLP
1100	TOP:	
1101	JSR	LOOKPRGM
1102	LDX	#\$FF
1103	^1	INX
1104	LDA	ERRTOP,X
1105	BEQ	>2
1106	JSR	COUT
1107	JMP	<1
1108	^2	LDA \$803
1109	STA	LINNUM

1110	STA	CURLIGNE
1111	LDA	\$804
1112	STA	LINNUM+1
1113	STA	CURLIGNE+1
1114	JSR	CRDO
1115	JSR	LIST
1116	JSR	CRDO
1117	JMP	CMDLP
1118	BOTTOM:	
1119	LDA	#\$01
1120	STA	ALLOWPRT
1121	JSR	LOOKPRGM
1122	LDX	#\$FF
1123	^1	INX
1124	LDA	ERRBOT,X
1125	BEQ	>2
1126	JSR	COUT
1127	JMP	<1
1128	^2	LDA \$6A
1129	TAX	
1130	DEX	
1131	STX	LSTLIN+1
1132	LDA	\$69
1133	STA	LSTLIN
1134	LDY	#\$F8
1135	^3	DEY
1136	LDA	(LSTLIN),Y
1137	BNE	<3

1138	TYA
1139	INY
1140	INY
1141	INY
1142 ^4	PHA
1143	LDA (LSTLIN),Y
1144	STA CURLIGNE
1145	STA LINNUM
1146	INY
1147	LDA (LSTLIN),Y
1148	STA CURLIGNE+1
1149	STA LINNUM+1
1150	JSR FNDLIN
1151	BCS >5
1152	PLA
1153	TAY
1154	DEY
1155	JMP <4
1156 ^5	JSR CRDO
1157	JSR LIST
1158	JSR CRDO
1159	JMP CMDLP
1160	BOTTOM2:
1161	LDA #S02
1162	JMP ERREUR
1163	SEARCH:
1164	JSR CRDO
1165	JSR POSBUF

1166	LDA	#\$01
1167	STA	ADL+1
1168	LDX	#\$FF
1169	INX	^0
1170	LDA	BUF,X
1171	CMP	#" "
1172	BEQ	<0
1173	INX	^1
1174	LDA	BUF,X
1175	CMP	#" "
1176	BNE	<1
1177	LDY	#\$FF
1178	INY	
1179	INX	
1180	LDA	BUF,X
1181	CMP	#"]"
1182	BEQ	>3
1183	STA	BUF3,Y
1184	CMP	#\$00
1185	BNE	<2+2
1186	LDA	#\$FF
1187	STA	\$44
1188	BNE	>4
1189	LDA	#\$00
1190	STA	BUF3,Y
1191	INX	
1192	LDA	BUF,X
1193	CMP	#" "

1194	BEQ	<3+5
1195	STX	LININDEX
1196	LDA	#\$00
1197	STA	\$44
1198	STA	\$45
1199	JSR	CONVERT
1200	CLC	
1201	LDA	\$45
1202	BNE	>7
1203	LDX	\$44
1204	BEQ	>6
1205	LDA	ADL+1
1206	BEQ	>0
1207	JSR	SRCHLIN
1208	BCS	>5
1209	JMP	<4
1210	JMP	BOTTOM
1211	LDA	#\$01
1212	STA	ALLOWPRT
1213	LDA	CURLIGNE
1214	STA	ADL2
1215	LDA	CURLIGNE+1
1216	STA	ADL2+1
1217	LDA	#\$00
1218	STA	CH
1219	JSR	LIST
1220	LDA	ADL2+1
1221	STA	CURLIGNE+1

1222	LDA	ADL2
1223	STA	CURLIGNE
1224	JSR	CRDO
1225	JMP	<4
1226 ^6	LDA	LINNUM
1227	STA	CURLIGNE
1228	LDA	LINNUM+1
1229	STA	CURLIGNE+1
1230	LDA	#\$01
1231	STA	ALLOWPRT
1232	JMP	CMDLP
1233 ^7	LDA	#\$04
1234	JMP	ERREUR
1235	SRCHLIN:	
1236	CLC	
1237	LDA	#\$02
1238	STA	ALLOWPRT
1239	LDA	CURLIGNE
1240	STA	LINNUM
1241	LDA	CURLIGNE+1
1242	STA	LINNUM+1
1243	DEC	\$44
1244	JSR	FNDLIN
1245	BCS	>1
1246 ^0	LDA	#\$01
1247	STA	ALLOWPRT
1248	JMP	BOTTOM
1249 ^1	JSR	LIST

1250	INY
1251	LDA (LOWTR),Y
1252	BNE >1
1253	INY
1254	LDA (LOWTR),Y
1255	BNE >2
1256	LDA #\$00
1257	STA ADL+1
1258	JMP >2
1259 ^1	INY
1260 ^2	INY
1261	LDA (LOWTR),Y
1262	STA CURLIGNE
1263	INY
1264	LDA (LOWTR),Y
1265	STA CURLIGNE+1
1266	LDY #\$00
1267 ENTRY3	LDX #\$00
1268 ^1	LDA BUF3,X
1269	STA FIRST
1270	LDA BUF,Y
1271	EOR #%10000000
1272	CMP #\$80
1273	BEQ >3
1274	CMP FIRST
1275	BEQ >2
1276	INY
1277	JMP <1+5

1278	^2	STY	FORPNT
1279		INX	
1280		INY	
1281		LDA	BUF3,X
1282		BEQ	>4
1283		AND	##01111111
1284		CMP	BUF,Y
1285		BEQ	<2+2
1286		LDY	FORPNT
1287		INY	
1288		LDX	#\$00
1289		JMP	<1+5
1290	^3	CLC	
1291		RTS	
1292	^4	SEC	
1293		RTS	
1294	MODIFY:		
1295		JSR	CRDO
1296		JSR	POSBUF
1297		LDA	#\$01
1298		STA	ADL+1
1299		LDX	#\$FF
1300	^0	INX	
1301		LDA	BUF,X
1302		CMP	#" "
1303		BEQ	<0
1304	^1	INX	
1305		LDA	BUF,X

1306	CMP	#" "
1307	BNE	<1
1308	LDY	#\$FF
1309 ^2	INY	
1310	INX	
1311	LDA	BUF,X
1312	BNE	#"]"
1313	BEQ	>3
1314	CMP	#\$80
1315	BEQ	>7
1316	STA	BUF3,Y
1317	JMP	<2
1318 ^3	LDA	#\$00
1319	STA	BUF3,Y
1320 ^4	INX	
1321	INY	
1322	LDA	BUF,X
1323	CMP	#\$80
1324	BEQ	>8
1325	CMP	#"]"
1326	BEQ	>5
1327	STA	BUF3,Y
1328	JMP	<4
1329 ^5	LDA	#\$00
1330	STA	BUF3,Y
1331 ^6	INX	
1332	LDA	BUF,X
1333	CMP	#" "

1334		BEQ	<6
1335		STX	LININDEX
1336		LDA	#\$00
1337		STA	\$44
1338		STA	\$45
1339		JSR	CONVERT
1340		LDA	\$45
1341		BEQ	>8
1342	^7	LDA	#\$04
1343		JMP	ERREUR
1344	^8	LDA	\$44
1345		BNE	ENTRY
1346		LDA	#\$FF
1347		STA	\$44
1348	ENTRY	LDX	\$44
1349		BEQ	>0
1350		LDA	ADL+1
1351		BEQ	>9
1352		JSR	SRCHLIN
1353		BCS	>1
1354		BCC	ENTRY
1355	^9	JMP	BOTTOM
1356	^0	LDA	#\$01
1357		STA	ALLOWPRT
1358		JMP	CMDLP
1359	^1	STY	FIRST
1360		LDY	#\$00
1361		INX	

1362	^2	CPY	FORPNT
1363		BEQ	>3
1364		LDA	BUF, Y
1365		STA	\$100, Y
1366		INY	
1367		JMP	<2
1368	^3	LDA	BUF3, X
1369		BEQ	>4
1370		STA	\$100, Y
1371		INX	
1372		INY	
1373		JMP	<3
1374	^4	STY	ADL
1375		LDX	FIRST
1376	^5	LDA	BUF, X
1377		BEQ	>6
1378		STA	\$100, Y
1379		INX	
1380		INY	
1381		BNE	<5
1382		LDA	#\$00
1383		STA	\$1FF
1384	^6	STA	\$100, Y
1385		LDY	#\$00
1386	^7	LDA	\$100, Y
1387		BEQ	>8
1388		AND	##01111111
1389		STA	BUF, Y

1390	INY
1391	JMP <7
1392 ^8	STA BUF, Y
1393	LDY ADL
1394	JSR ENTRY3
1395	BCS <1
1396	LDA \$36
1397	PHA
1398	LDA \$37
1399	PHA
1400	LDA #PRLIGNE
1401	STA \$36
1402	LDA /PRLIGNE
1403	STA \$37
1404	LDA #\$00
1405	STA \$FE
1406	LDA LINNUM+1
1407	LDX LINNUM
1408	JSR LINPRT
1409	LDY \$FE
1410	LDA #\$20
1411	STA BUF, Y
1412	PLA
1413	STA \$37
1414	PLA
1415	STA \$36
1416	LDA #\$01
1417	STA SAVEA

1418	LDX	#\$FF
1419	LDY	#\$01
1420	JMP	CMDLP2+8
1421	PRLIGNE:	
1422	LDY	\$FE
1423	AND	##01111111
1424	STA	BUF,Y
1425	INC	\$FE
1426	RTS	
1427	APPEND:	
1428	LDA	#\$00
1429	STA	ALLOWPRT
1430	LDA	CURLIGNE
1431	STA	LINNUM
1432	LDA	CURLIGNE+1
1433	STA	LINNUM+1
1434	JSR	CRDO
1435	JSR	LIST
1436	LDY	#\$00
1437	^1	LDA (\$28),Y
1438	CMP	#" "
1439	BEQ	>2
1440	AND	##01111111
1441	STA	BUF,Y
1442	INY	
1443	JMP	<1
1444	^2	LDY #\$01
1445	STY	ALLOWPRT

1446	DEY
1447	STY CH
1448	LDY INDEX2
1449	LDX #\$FF
1450 ^3	INX
1451	LDA BUF2,X
1452	CMP #\$20
1453	BEQ >4-1
1454 ^3	LDA BUF2,X
1455	CMP #\$20
1456	BEQ >4-1
1457	INX
1458	JMP <3
1459	DEY
1460 ^4	INX
1461	INY
1462	LDA BUF2,X
1463	BEQ >5
1464	AND #%01111111
1465	STA BUF,Y
1466	BNE <4
1467 ^5	STA BUF,Y
1468	LDA #\$02
1469	STA LIST2
1470	LDY #\$01
1471	LDX #\$FF
1472	STY TXTPTR+1
1473	STX TXTPTR

1474		JSR	CHRGET
1475		JMP	PROCLN
1476	PRBUF:		
1477		LDA	#\$00
1478		STA	CRDNB
1479		LDX	#\$00
1480		STX	\$4D
1481	^1	LDA	BUF,X
1482		CMP	#\$22
1483		BNE	>2
1484		PHA	
1485		LDA	#\$FF
1486		EOR	\$4D
1487		STA	\$4D
1488		PLA	
1489	^2	CMP	#\$3A
1490		BNE	>3
1491		LDY	\$4D
1492		BNE	>3
1493		JSR	\$FC42
1494		LDA	#\$8D
1495		JSR	OUTDO
1496		LDA	POSCUR
1497		STA	CH
1498		LDA	#\$3A
1499		INC	CRDNB
1500	^3	CMP	#\$00
1501		BEQ	>5

1502	JSR	OUTDO
1503	LDA	CH
1504	CMP	#139
1505	BNE	>0
1506	JSR	\$FC9C
1507 ^0	LDA	CH
1508	BNE	>4
1509	INC	CRDNB
1510 ^4	INX	
1511	BNE	<1
1512 ^5	JSR	\$FC42
1513	JSR	CRDO
1514	RTS	
1515	CURPOS:	
1516	JSR	CURPOS2
1517	LDA	#\$00
1518	STA	CRDNB
1519	STA	FIRST
1520	LDX	#\$00
1521	STX	\$4D
1522 ^1	LDA	BUF,X
1523	CPX	INDEX
1524	BEQ	>5
1525	CMP	#\$22
1526	BNE	>2
1527	PHA	
1528	LDA	#\$FF
1529	EOR	\$4D

1530	STA	\$4D
1531	PLA	
1532	^2	CMP #3A
1533	BNE	>3
1534	LDY	\$4D
1535	BNE	>3
1536	INC	CRDNB
1537	LDY	POSCUR
1538	STY	FIRST
1539	^3	CMP #00
1540	BEQ	>6
1541	INC	FIRST
1542	LDA	FIRST
1543	CMP	#140
1544	BNE	>4
1545	LDA	#00
1546	STA	FIRST
1547	INC	CRDNB
1548	^4	INX
1549	BNE	<1
1550	^5	CMP #3A
1551	BEQ	>7
1552	LDA	FIRST
1553	STA	CH
1554	LDA	CRDNB
1555	CLC	
1556	ADC	ADL2
1557	STA	CV

1558	JSR	\$FC22
1559	RTS	
1560	^6	DEC INDEX
1561	JMP	<5
1562	^7	INC CRDNB
1563	LDA	POSCUR
1564	STA	FIRST
1565	JMP	<5
1566	CURPOS2:	
1567	LDY	#\$00
1568	STY	CH
1569	^1	LDA (\$28),Y
1570	CMP	#">"
1571	BEQ	>2
1572	JSR	UP
1573	JMP	<1
1574	^2	INY
1575	LDA	(\$28),Y
1576	CMP	#" "
1577	BEQ	<2
1578	CMP	#"E"
1579	BEQ	>3
1580	JSR	UP
1581	JMP	<1-2
1582	^3	LDY CV
1583	INY	
1584	INY	
1585	STY	ADL2

1586	RTS
1587	PP:
1588	JSR CRDO
1589	LDA CURLIGNE
1590	STA ADL2
1591	STA LINNUM
1592	LDA CURLIGNE+1
1593	STA ADL2+1
1594	STA LINNUM+1
1595	LDA # \$02
1596	STA \$44
1597 ^1	JSR FNDLIN
1598	JSR NXTLIN2
1599	BCC >3
1600	LDA \$44
1601	BNE <1
1602	LDA # \$05
1603	STA \$44
1604	LDA LINNUM
1605	STA CURLIGNE
1606	LDA LINNUM+1
1607	STA CURLIGNE+1
1608 ^2	JSR PRINTUNA
1609	BCC >4
1610	LDA \$44
1611	BNE <2
1612	LDA ADL2
1613	STA CURLIGNE

1614	LDA	ADL2+1
1615	STA	CURLIGNE+1
1616	JMP	CMDLP
1617 ^3	LDX	#\$FF
1618	INX	
1619	LDA	ERRTOP,X
1620	BEQ	>3
1621	JSR	COUT
1622	JMP	<3+2
1623 ^3	LDX	#\$02
1624	DEX	
1625	DEC	\$44
1626	LDA	\$44
1627	BNE	<3+2
1628	INX	
1629	INX	
1630	STX	\$44
1631	JMP	<1+16
1632 ^4	LDX	#\$FF
1633 ^5	INX	
1634	LDA	ERRBOT,X
1635	BEQ	>6
1636	JSR	COUT
1637	JMP	<5
1638 ^6	LDA	ADL2
1639	STA	CURLIGNE
1640	LDA	ADL2+1
1641	STA	CURLIGNE+1

```
1642          JMP  CMDLP
1643 REPEAT:
1644          LDY  #$00
1645 RETBUF    LDA  BUF2,Y
1646          STA  BUF,Y
1647          INY
1648          BNE  RETBUF
1649          JMP  PARSE2
1650 LOOKPRGM
      :
1651          LDA  $69
1652          CMP  #$04
1653          BNE  NOPROB
1654          LDA  $6A
1655          CMP  #$08
1656          BNE  NOPROB
1657          LDA  #$03
1658          JMP  ERREUR
1659 NOPROB    RTS
1660 PR:
1661          JSR  POSBUF
1662          LDA  #$00
1663          STA  $44
1664          STA  $45
1665          TAX
1666 ^1       INX
1667          LDA  BUF,X
1668          CMP  #"#"
1669          BNE  <1
```

1670	^2	INX
1671		LDA BUF,X
1672		CMP #" "
1673		BEQ <2
1674		CMP #B0
1675		BCC >3
1676		CMP #BA
1677		BCS >3
1678		STX LININDEX
1679		JSR CONVERT
1680		LDA \$45
1681		BNE >3
1682		LDA \$44
1683		CMP #08
1684		BCS >3
1685		JSR PROUT
1686		JMP CMDLP
1687	^3	LDA #04
1688		JMP ERREUR
1689	DOLLAR:	
1690		JSR POSBUF
1691		LDX #FF
1692	^1	INX
1693		LDA BUF,X
1694		BEQ >2
1695		CMP #" \$"
1696		BNE <1
1697		STX LININDEX

1698	JSR	CONVERT
1699	BCC	>2
1700	JSR	CRDO
1701	LDA	#"="
1702	JSR	COUT
1703	LDX	\$44
1704	LDA	\$45
1705	JSR	LINPRT
1706	JSR	CRDO
1707	JMP	CMDLP
1708 ^2	LDA	#\$05
1709	JMP	ERREUR
1710	POSBUF:	
1711	LDX	#\$FF
1712 ^1	INX	
1713	LDA	BUF,X
1714	BEQ	>2
1715	ORA	##10000000
1716	STA	BUF,X
1717	JMP	<1
1718 ^2	RTS	
1719	NOLINE:	
1720	ASC	"NO SUCH LINE."
1721	HEX	8D00
1722	SORRY:	
1723	HEX	8D
1724	ASC	"SORRY."
1725	HEX	8D00

1726 ERRBOT:

1727 HEX 8D

1728 ASC "BOTTOM."

1729 HEX 8D00

1730 NOPRGM:

1731 HEX 8D

1732 ASC "NO PROGRAM IS LOADED."

1733 HEX 8D00

1734 BADARG:

1735 HEX 8D

1736 ASC "BAD ARG. #."

1737 HEX 8D00

1738 ERRTOP:

1739 HEX 8D

1740 ASC "TOP."

1741 HEX 8D00

1742 ERRHEX:

1743 HEX 8D

1744 ASC "ERROR IN HEX. DIGITS."

1745 HEX 8D00

1746 MESSCOM:

1747 ASC "COMMAND"

1748 HEX 00

1749 MESSTYPE
:

1750 ASC "TYPE "

1751 HEX 00

1752 MESSEDT
:

1753	ASC	"EDIT: "
1754	HEX	00
1755	MESSDOS:	
1756	ASC	"DOS: "
1757	HEX	00
1758	TBLCOL:	
1759	ASC	"1234567890123456789012345678901234567890"
1760	HLPMSG1:	
1761	ASC	"HELP"
1762	HEX	00
1763	HLPMSG2:	
1764	ASC	"PRESSEZ UNE TOUCHE"
1765	HEX	00
1766	;	
1767	HELP:	
1768	LDA	#!24
1769	STA	\$23
1770	LDA	#HLPMSG
1771	STA	ADL
1772	LDA	/HLPMSG
1773	STA	ADL+1
1774	LDY	#\$00
1775	STY	TEMP
1776	^9	JSR HOME
1777	LDA	#!18
1778	STA	CH
1779	LDY	#\$00
1780	^0	LDA HLPMSG1,Y

1781	BEQ	>1
1782	JSR	\$FDED
1783	INY	
1784	JMP	<0
1785 ^1	LDX	#!23
1786	JSR	VTAB
1787	LDA	#!11
1788	STA	CH
1789	LDY	#\$00
1790 ^2	LDA	HLPMSG2,Y
1791	BEQ	>3
1792	JSR	\$FDED
1793	INY	
1794	JMP	<2
1795 ^3	LDY	TEMP
1796	LDA	(ADL),Y
1797	BEQ	>6
1798	STY	TEMP
1799	TAX	
1800	JSR	VTAB
1801	LDY	TEMP
1802	INY	
1803	BNE	>0
1804	INC	ADL+1
1805 ^0	LDA	(ADL),Y
1806	STA	CH
1807 ^4	INY	
1808	BNE	>0

1809	INC	ADL+1
1810	^0	LDA (ADL),Y
1811	BEQ	>5
1812	JSR	\$FDED
1813	JMP	<4
1814	^5	INY
1815	BNE	>0
1816	INC	ADL+1
1817	^0	STY TEMP
1818	JMP	<3
1819	^6	STY TEMP
1820	JSR	KEYIN
1821	LDY	TEMP
1822	INY	
1823	BNE	>0
1824	INC	ADL+1
1825	^0	STY TEMP
1826	LDA	(ADL),Y
1827	BNE	<9
1828	JMP	\$7000
1829	;	
1830	HLPMSG:	
1831	END	