

la revue des

# SHARP ENTIERS

La revue des  
**SHAI SHA**

La revue des  
**SHARPE SHARPENTIERS**

La revue des  
**SHARPENTIERS**

La revue des  
**SHARPENTIERS**

La revue des  
**SHARPENTIERS**

La revue des  
**SHARPENTIERS**

**DANS CE NUMERO**

**GÉREZ VOTRE COMPTE**  
**LA NOUVELLE RUBRIQUE 1600**  
**3D LABY SUR 800**

**ET TOUJOURS :**  
**L'INITIATION, LES GRAPHIQUES, LE LM...**

30 F.

N° 18 - BIMESTRIEL

MARS 86-30 F.

LA REVUE DES UTILISATEURS DE MICRO-ORDINATEURS ET POCKET-COMPUTERS SHARP

DANS CE NUMERO  
RÉSULTATS DU CONCOURS  
LALES DATES  
GRAPHISME 3D  
DES PLOTTERS  
DANS L'OUEST  
TROUVEZ VOTRE MZ 700

**DIRECTEUR  
DE PUBLICATION**

Luc BURELLER

**RÉDACTEUR EN CHEF**

Jean-François VIGNAUD

**RÉDACTEURS**

Luc BURELLER  
 Simon CHAGNOUX  
 Maurice CHOUCROUN  
 Marc GIRONDOT  
 Christophe POULIN

**SECRÉTAIRES  
DE RÉDACTION**

Dominique DUBAN  
 Anne ONYME

**ONT COLLABORÉ  
A CE NUMÉRO**

Jean DUBUS  
 Jacques FRETEY  
 Jérôme GAUDIN  
 Frédéric GENAUDET  
 Jérôme GESLIN  
 Maurice GODEAUX  
 Edmond LEMAÎTRE  
 Julien LEVER  
 Daniel MAGNIN  
 Christophe NICOLAS  
 Olivier PROTTE  
 Martin RAYROLE

**REMERCIEMENTS À**

M. F. MIZZI  
 M. P. MIZZI  
 M. P.L. DE QUATREBARBES

**RÉALISATION**

IN QUARTO  
 19, rue Frédéric-Lemaître  
 75020 PARIS

**PUBLICITÉ**

Jean-François VIGNAUD

SHARP est une marque déposée. Le "CLUB DES SHARPENTIERS" et la revue "LE SHARPENTIER" sont totalement indépendants de l'importateur SBM et du groupe SHARP CORPORATION.

La revue des Sharpentiers est éditée par le Club des Sharpentiers. c/o Micro Archi 79, rue du Temple 75003 Paris Dépôt légal à parution Ce numéro a été tiré à 2 200 ex.

**EDITO**

Une nouvelle année s'annonce, et grâce à vous, le CLUB continue. Vous êtes nombreux à vous être inscrits en avance ou à nous avoir envoyés un complément d'inscription. Nous vous en remercions car ainsi vous continuerez tous à recevoir votre revue.

Une nouvelle année, ce sont de nouveaux services, le serveur qui désormais est opérationnel.

Une nouvelle année, ce sont des changements : là c'est encore un secret pour le moment.

Nous vous attendons nombreux le mercredi au Club ; la permanence se tient de 14 heures à 21 heures sans interruption. Nous espérons vous y rencontrer et que vous pourrez nous aider en donnant un peu de votre temps.

Toute la rédaction et le bureau vous souhaitent un bonne et excellente année 1987.

Luc BURELLER

**OÙ EN EST  
LE SERVEUR MINITEL ?**

Nos adhérents gastronomes savent bien que les serveurs se font toujours attendre. Le nôtre ne fait pas exception : il n'a commencé son existence que quelques semaines auparavant, au sein d'HG, celui de nos confrères (et néanmoins amis) d'"Hebdogiciel".

A l'heure (\*) où j'écris ces lignes, il ne figure pas au menu d'HG mais l'on peut déjà admirer le nouveau-né : il suffit de composer le 3615 (désolé pour la grosse erreur du dernier bulletin), de taper HG puis ENVOI, puis deux fois « sharp » validés par ENVOI à chaque fois, même (et surtout) si cela

n'a aucun rapport avec les options qu'affiche le Minitel.

Cette procédure barbare devrait rapidement être remplacée pour que lorsque vous aurez reçu le bulletin (c'est-à-dire maintenant !), l'accès se fasse par HG\*CS ou par une option « CLUB » figurant au menu.

De toutes façons, vous pouvez dès à présent nous laisser vos critiques (constructives) dans une boîte aux lettres électronique, car ça c'est déjà au point.

A suivre.

Angus MONCHOIX

(\*) 23 heures 42, le 21/12/86.

**BULLETIN D'INSCRIPTION AU CLUB DES SHARPENTIERS**

n° 21

Je m'inscris  
au CLUB DES SHARPENTIERS

Je bénéficie de tous les  
avantages du CLUB

Je suis abonné pour 1 AN  
au BULLETIN du CLUB

Je vous joins mon règlement

FRANCE : 240F  
 ETRANGER : 300F

CHEQUE N° .....

BANQUE .....

DATE .....

SIGNATURE .....

NOM ..... PRÉNOM .....

ADRESSE .....

CODE POSTAL / VILLE .....

PAYS .....

PROFESSION ..... ÂGE .....

MACHINE POSSEDÉE ..... DEPUIS .....

ACHETÉE CHEZ .....

UTILISATION PRINCIPALE DE VOTRE MACHINE .....

CLUB DES SHARPENTIERS c/o MICRO-ARCHI 79, rue du Temple, 75003 PARIS

# INITIATION

## LH-5801 III

**Nous passons enfin à des applications un peu plus utiles que ce que l'on a fait jusqu'à présent. Et la suite...**

Vous savez certainement que l'on peut mettre plusieurs types d'informations dans la MEV du PC-1500.

- le mode RESERVE
- des programmes en LM
- des programmes BASIC MERGES
- des Variables dimensionnées.

En fait chacune de ces informations se mettra à une place bien déterminée et le PC-1500 sait exactement où, grâce à d'autres octets de la MEV système.

Nous avons tout au début de notre mémoire, donc à partir de NN00, huit octets qui servent à la protection des modules. L'explication exacte de ces huit octets est donnée dans le numéro 11, page 13. Puis il y a ce que l'on appelle les gabarits réserve donc ce qui s'affiche lorsque vous appuyez sur la touche RCL. Chaque gabarit utilise 26 octets, soit 78 octets au total, plus les 8 du départ, soit 86 octets après le début de la MEV, on trouve la définition des touches réserve. Une partie des explications a déjà été donnée dans le numéro 10, page 23. Nous allons les reprendre brièvement :

A chaque touche du mode réserve est associée un code :

	I	III	II
!	1	8	17
"	2	9	18
#	3	10	19
\$	4	11	20
%	5	12	21
&	6	13	22

Ce code se trouve en mémoire avant la définition du contenu de la touche.

Cette définition est codée de manière tout à fait courante. Les caractères sont codés par leur code ASCII et les fonctions par leur code (on trouvera cette dernière liste en page 115 du TRM). Il n'y a pas de code séparateur spécial entre deux définitions ; si le PC trouve un code inférieur à &20(32), il sait que c'est la définition d'une autre touche qui commence, et s'il trouve un code 0, qu'il n'y a plus de définition.

Vous trouverez dans le numéro 16, page 26, un programme permettant de lister le mode réserve, et voici un autre utilitaire permettant de modifier un

gabarit, chose impossible en temps normal. Le gabarit est ce qui s'affiche lorsqu'on appuie sur RCL.

Voir listing 1 : Modification de Gabarits

RUN ENTER

▼ pour changer de numéro de réserve  
◀ et ▶ pour faire bouger le curseur  
ENTER pour sortir  
une touche pour modifier le signe sous le curseur.

Un dernier petit exercice pour être sûr de bien avoir saisi le codage du mode RESERVE :

Mettre sur la touche ! en réserve I le caractère de pavé noir. On annulera le contenu des autres touches.

A = PEEK &7863\*256 - Calcul du début de la MEV

A = A+86 - Calcul du 1<sup>er</sup> octet des définitions

POKE A,1 - Code de F1 en mode I

POKE A + 1,127 - Code du pavé noir

POKE A + 2,0 - Fin du mot RESERVE.

Nous allons sauter la partie correspondant au langage machine. Ce sera pour plus tard, nous passerons la prochaine fois au codage du BASIC.

Pour compléter cet article, vous trouverez la table des codes ASCII du PC-1500.

Marc GIRONDOT

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	MUL FO	SHIFT F1	SML F2	F3	F4	F5	F6	F7	+	=	↓	↑	→	ENTER	CL OFF
1	FO	F1	F2	F3	F4	F5	F6	F7	CL	RCL	CA	DEF	IMS	DEL	RSU MODE
2	SPACE	!	*	\$	\$	%	%	□	(	)	*	+	,	-	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	) ?
4	a	A	B	C	D	E	F	G	H	I	J	K	L	M	H 0
5	P	Q	R	S	T	U	U	W	X	Y	Z	J	Y	X	^ -
6	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	P	Q	r	s	t	u	v	w	x	y	z	{	}	-	■
8	DEF SP	DEF A	DEF B	DEF C	DEF D	DEF E	DEF F	DEF G	DEF H	DEF I	DEF J	DEF K	DEF L	DEF M	DEF N
9	DEF P LIST	DEF Q INPUT	DEF R GOTO	DEF S	DEF T GOSUB	DEF U CSAVE	DEF V	DEF W PRINT	DEF X RETURN	DEF Y	DEF Z	DEF SP	DEF SP	DEF =	DEF SP
A															
B															
C															
D															
E	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX
F	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX	PREFIX

Listing 1 : MODIFICATION DE GABARITS

```

10 CLEAR :DIM A$(0)*28:REM initialise A$(0) Pour contenir un gabarit
15 A=PEEK 30286:REM valeur de l'indicateur Positionnant le mode réserve
20 A=2-INT ((AAND 112)/32):REM A=0->I A=1->II A=2->III
25 A=8+PEEK &7863*A*26:REM 1er octet du gabarit
30 A$(0)="":I=0
35 IF PEEK (A+I)=0:THEN 45
40 A$(0)=A$(0)+CHR$ PEEK (A+I):I=I+1:GOTO 35
45 IF LEN A$(0)=27:THEN 55
50 FOR I=LEN A$(0)TO 27:A$(0)=A$(0)+" ":"NEXT I
55 A$(0)=A$(0)+""
60 I=0:C=0:WAIT 0:CLS
65 CURSOR :PRINT RIGHT$ (A$(0),27);
67 A$=MID$ (A$(0),I+2,1)
70 C=C+1:IF C=5CURSOR I:PRINT A$;
75 IF C=10CURSOR I:PRINT CHR$ 127::C=0
77 J=ASC INKEY$ :D=0
80 IF J=9BEEP 1:GOTO 100
85 IF J=8CURSOR I:PRINT A$::I=I-1+26*(I=0):C=9:GOTO 67
90 IF J=12CURSOR I:PRINT A$::I=I+1-26*(I=25):C=9:GOTO 67-D
91 IF J=13GOSUB 120:END
92 IF J>31LET A$(0)=LEFT$ (A$(0),I+1)+CHR$ J+RIGHT$ (A$(0),28-I-2)::J=12:D=2:GOTO 90
95 GOTO 70
100 GOSUB 120:A=(PEEK 30286AND 112)/2
105 IF (AAND 8)=8LET A=64
110 POK 30286,(PEEK 30286AND 143)OR A:GOTO 15
120 FOR K=0TO 25:POKE A+K,ASC MID$ (A$(0),K+2,1)::NEXT K:RETURN

```

## AJUSTEMENT DE COURBE

**Ce programme permet de déterminer la courbe qui passe le plus près d'une série de points donnés.**

Pour vérifier que vous n'avez pas fait d'erreur en tapant ces lignes de DATA, vous pouvez entrer ces 3 lignes que vous effacerez ensuite :

2000:X = 0.5,Y = 1,A = 1,B = 1,C = 1,  
D = 1,E = 1,F = 1,Z = 1,S = 0:ON ERROR  
GOTO 2002

2001:READ I:S = S + I:GOTO 2001

2002:PRINT S:END

Si le résultat affiché est différent de 3907.05..., c'est que vous avez fait une erreur de frappe.

### FONCTIONNEMENT DU PROGRAMME

Après avoir fait RUN, entrez les coordonnées des différents points. Pour retirer des points, faites DEF S et donnez les coordonnées des points à retirer, puis faites DEF A pour recommencer à rentrer des points.

Lorsque vous aurez entré tous les points, tapez simplement ENTER à la question «X = ».

Vous avez alors deux possibilités pour effectuer l'ajustement de la courbe :  
 - pour obtenir une courbe particulière, tapez son numéro et ENTER (1 à 25)  
 - pour obtenir la meilleure courbe, tapez simplement ENTER (attention, les courbes 24 et 25 ne sont pas prises en compte). Le numéro de la courbe choisie s'affiche alors, après 10 à 20 secondes de calcul.

Le programme commence par donner les coefficients a, b et c de l'équation de la courbe, puis le coefficient de corrélation noté r compris entre 0 et 1 et qui détermine la justesse de l'ajustement (r=1 si la courbe passe très près de tous les points).

Vous pouvez ensuite tester la courbe ajustée en donnant une valeur de X et le programme calcule Y(X).

**ATTENTION :** si tous les points que vous avez rentrés sont très éloignés de

toutes les courbes, il peut arriver que le programme donne de mauvaises valeurs et un coefficient de corrélation égal à 1. Pour se rendre compte de ces erreurs, il suffit de calculer Y(X) pour certaines valeurs de X connues.

Le programme ne peut ajuster une courbe que si toutes les valeurs de X et Y entrées vérifient les conditions inscrites à côté des courbes. Il est donc préférable de ne pas entrer le point (0,0), même si la courbe doit y passer (seules la droite et la parabole acceptent ce point).

Ce programme a été mis au point avec les formules parues dans « Curve fitting for programmable calculators » de William M. Kolb.

Martin RAYROLE

#### AJUSTEMENT

```

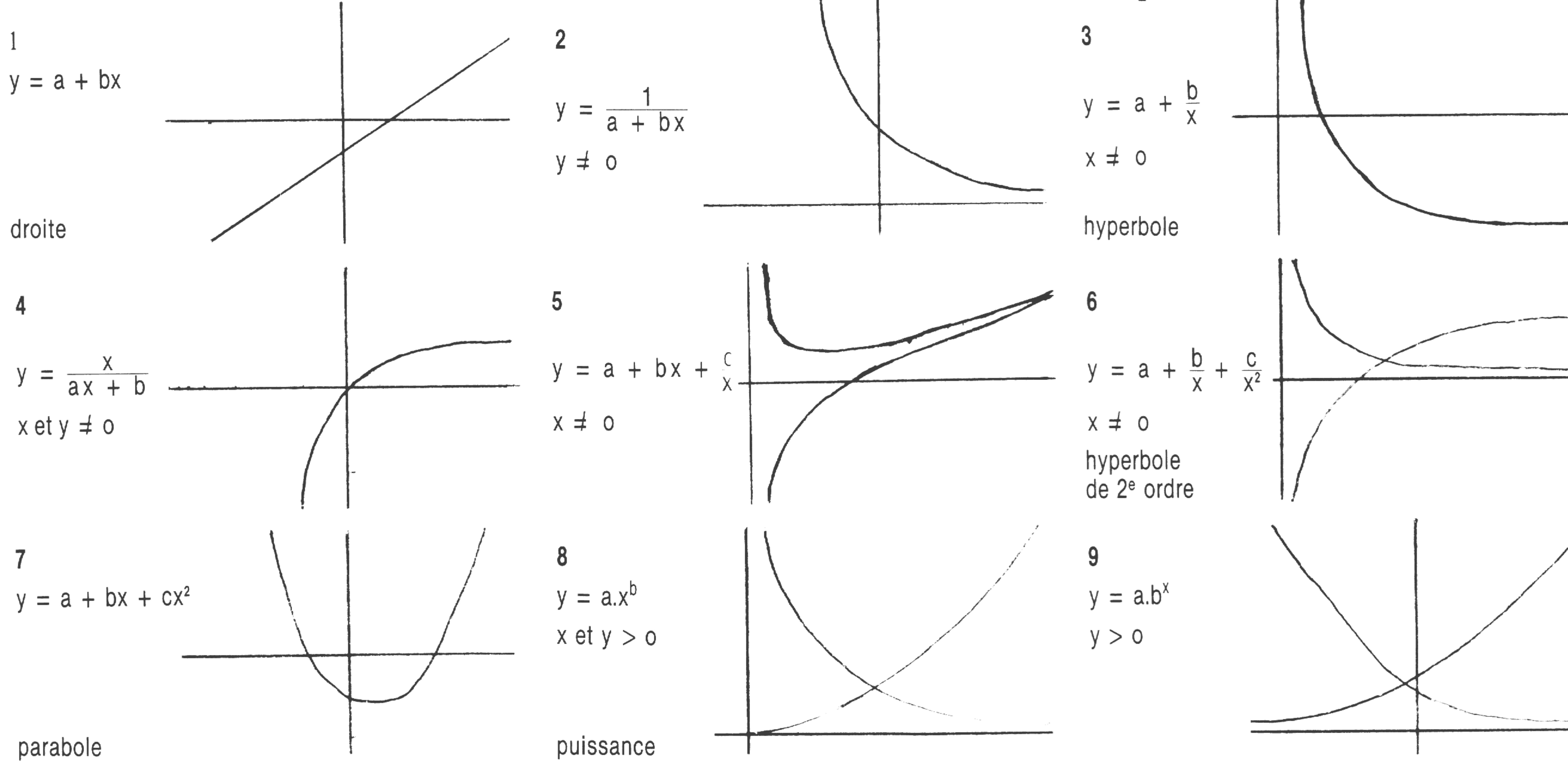
10: DIM R(46), T(6): FOR I=0 TO 6: T(I)=1: NEXT I
12: "A"K=1: WAIT 0: GOTO 20
16: "S"K=-1: WAIT 0
20: CLS : BEEP 1,80,30: A$="Point "+STR$(R(5)+1): PRINT A$:
  INPUT " X="; X: CLS : PRINT A$:: INPUT " Y="; Y: GOTO 40
30: GOTO 200
40: T(1)=T(1) AND X>0, T(2)=T(2) AND X>0, T(3)=T(3) AND Y>0,
  T(4)=T(4) AND Y>0
50: T(5)=T(5) AND X>0 AND X<1, T(6)=T(6) AND X>0 AND (X-INT X)=0
  AND X<30
55: A=X*X, B=Y*Y: RESTORE : GOSUB 800
60: IF T(1)GOSUB 800
70: IF T(3)RESTORE 902: GOSUB 800: IF T(1)LET R(10)=R(10)+K/
  X*Y
80: IF T(2)RESTORE 903: C=LN X, E=C*C: GOSUB 800: IF T(3)LET R
  (36)=R(36)+K*C/Y
90: IF T(4)RESTORE 904: D=LN Y, F=D*D: GOSUB 800: IF T(1)LET R
  (31)=R(31)+K*D/X
100: IF T(2)AND T(4)GOSUB 800
110: IF T(5)AND T(4)LET Z=LN (1-X): GOSUB 800
120: IF T(6)AND T(4)GOSUB 700: F=LN F: RESTORE 907: GOSUB 800
130: GOTO 20
200: CLS : WAIT : INPUT " Numéro de la courbe "; J: GOTO 300
210: ZM=0: FOR J=1 TO 23: GOSUB 810: IF Z>ZMLET ZM=Z, NM=J
220: NEXT J: J=NM: BEEP 1: PRINT USING ; " Courbe numero "; J
300: GOSUB 810: IF Z=0PRINT " Points hors du domaine": GOTO 20
  0


```

```

310: READ S: IF NN=6LET M=C+D*S, D=E+F*S, U=(M*A-B*B*S)/L, V=0
  *N-M*B*S I=L, S=3: GOTO 320
320: A=U, B=V, C=W: GOTO 600+S
600: A=W, B=U/2-W, C=U-B*U/2: GOTO 605
601: A=EXP (U-W*LN U), B=1/U: GOTO 605
602: B=-U/2-W, A=EXP (U+B*U/2), C=1/W: GOTO 605
603: B=EXP U
604: A=EXP U
605: FOR I=1 TO T: PRINT USING ; CHR$(196+I); "="; a(I): NEXT I
610: IF J<24PRINT USING ; "#.###"; r="; SQR R+SE-5
620: ON ERROR GOTO 670
630: INPUT " X="; X: GOTO 650
640: GOTO 200
650: IF J=24GOSUB 700
660: RESTORE 1000+J: READ N, N, N: FOR I=0 TO N: READ Z: NEXT I:
  READ Y: PRINT USING ; " Y="; Y: GOTO 630
670: PRINT " Y non défini": GOTO 630
700: F=1: FOR I=1 TO X: F=F*I: NEXT I: RETURN
800: READ N: FOR I=1 TO N: READ R, S: R(R)=R(R)+S*K: NEXT I:
  RETURN
810: RESTORE 1E3+J: Z=0: READ X, Y: IF T(X)+T(Y)<2RETURN
815: R=0: ON ERROR GOTO 880
820: READ NN: T=2+(NN=9)
821: FOR Z=1 TO NN: READ W: a(Z)=R(W): NEXT Z
830: N=R(5), L=A*N-B*B, P=D*N-B*C
840: IF NN=5LET U=(A+C-B*D)/L, V=P/L, W=0: GOTO 870
850: M=F*N-C*G, D=H*N-B*G, Q=I*N-G*G
860: W=(L*M-D*P)/(L*Q-D*O), U=(P-D*W)/L, V=(C-U*B-W*G)/N
870: R=(U*C+V*D+W*F-C*C/N)/(E-C*C/N), Z=1-(1-R)*(N-1)/(N-T)
880: ON ERROR GOTO 0: RETURN
900: DATA 1, 0, X, 1, A, 2, Y, 3, B, 4, X*Y, 5, 1, 20, A*Y, 23, X*B, 24, X*A
  , 26, Y*B, 27, A*A
901: DATA 6, 6, 1/X, 7, 1, 19, Y/X, 22, Y/A, 25, 1/X/A, 28, 1/A/A
902: DATA 4, 8, 1/Y, 9, 1/B, 18, X/Y, 21, A/Y
903: DATA 9, 12, C, 13, E, 29, C/X, 32, X*C, 33, A*C, 35, Y*C, 37, E/A, 39
  , E*C, 40, E*E
904: DATA 4, 14, D, 15, F, 30, X*D, 38, A*D
905: DATA 4, 16, C*D, 34, X*C*D, 41, E*D, 42, C*D*X
906: DATA 4, 43, Z, 44, Z*Z, 45, C*Z, 46, D*Z
907: DATA 2, 11, F, 17, X*F
1001: DATA 0, 0, 5, 1, 0, 2, 4, 3, 5, A+B*X
1002: DATA 0, 3, 5, 1, 0, 8, 18, 9, 5, 1/(A+B*X)
1003: DATA 1, 0, 5, 7, 6, 2, 19, 3, 5, A+B*X
1004: DATA 1, 3, 5, 7, 6, 8, 10, 9, 5, X/(A*X+B)
1005: DATA 1, 0, 9, 1, 0, 2, 4, 3, 19, 6, 5, 7, 5, A+B*X+C*X
1006: DATA 1, 0, 9, 7, 6, 2, 19, 3, 22, 7, 25, 28, 5, A+(B+C*X)/X
1007: DATA 0, 0, 9, 1, 0, 2, 4, 3, 20, 1, 24, 27, 5, A+(B+C*X)**X
1008: DATA 2, 4, 5, 13, 12, 14, 16, 15, 4, A*X*B
1009: DATA 0, 4, 5, 1, 0, 14, 30, 15, 3, A*B*X
1010: DATA 1, 4, 5, 7, 6, 14, 31, 15, 3, A*B^(1/X)
1011: DATA 2, 4, 5, 33, 32, 14, 34, 15, 4, A*X^(B*X)
1012: DATA 2, 4, 5, 37, 29, 14, 42, 15, 4, A*X^(B*X)
1013: DATA 0, 4, 5, 1, 0, 14, 30, 15, 4, A*EXP (B*X)
1014: DATA 1, 4, 5, 7, 6, 14, 31, 15, 4, A*EXP (B*X)
1015: DATA 2, 0, 5, 13, 12, 2, 35, 3, 5, A+B*LN X
1016: DATA 2, 3, 5, 13, 12, 8, 36, 9, 5, 1/(A+B*LN X)
1017: DATA 2, 4, 9, 1, 0, 14, 30, 15, 16, 12, 32, 13, 3, A*B*X*X*C
1018: DATA 2, 4, 9, 7, 6, 14, 31, 15, 16, 12, 29, 13, 3, A*B^(1/X)*X*C
1019: DATA 0, 4, 9, 1, 0, 14, 30, 15, 38, 1, 24, 27, 2, A*EXP ((X-B)^2/C)
1020: DATA 2, 4, 9, 13, 12, 14, 16, 15, 41, 13, 39, 40, 2, A*EXP ((B-LN X)
  )^2/C
1021: DATA 5, 4, 9, 13, 12, 14, 16, 15, 43, 45, 44, 4, A*X*B*(1-X)^C
1022: DATA 2, 4, 9, 1, 0, 14, 30, 15, 16, 12, 32, 13, 1, A*X*B^(1/C)*EXP (X
  /B)
1023: DATA 0, 3, 9, 1, 0, 8, 18, 9, 21, 1, 24, 27, 0, 1/A*X*B^(1/2+C)
1024: DATA 6, 4, 6, 1, 0, 11, 14, 17, 30, 1, A*B*X*F
1025: DATA 2, 4, 6, 1, 0, 14, 12, 32, 30, -1, A*X*B*X

```



**ASTUCE 1****ARRÊTER EN MARCHE**

Voici un petit truc simple que j'ai découvert sur mon PC-1500. Depuis pas mal de temps, je dispose en permanence d'un petit DRIVER CLAVIER qui, en particulier, remplace avantageusement la touche OFF par un SJP E33F (« Sharpentier » n° 11, page 21). Cependant, avec cette méthode, la fonction ARUN n'est plus disponible car elle ne se déclenche qu'au retour d'un vrai OFF. Mais il y a beaucoup mieux !

Essayez donc ceci : Placez un DRIVER du type : SJP &E24A

CPI A,OF  
BZR +03  
JMP E33F  
RTN

et lancez-le en mettant dans 785B-C l'adresse et dans 79D4 la valeur &55.

Maintenant rentrez le programme BASIC suivant :

10:CALL &E24A:GOTO 20

Lancez-le par RUN 10.

Le programme reboucle à chaque appui sur une touche autre que BREAK. Mais si on exécute la séquence SHIFT OFF, ô miracle ! le PC s'éteint et mieux encore, si on rallume la machine le programme tourne toujours.

Cela fonctionne sur n'importe quel programme sous DRIVER.

Si on place une routine E24A sous BASIC et l'on répond à cette routine par SHIFT OFF, au lieu de continuer le programme, la machine s'éteint. Puis quand on rallume, le PC attend toujours une réponse au CALL &E24A.

Avec cette méthode, voilà une façon simple de placer des mots de passe à l'allumage et d'autres astuces de ce genre. On peut aussi faire démarrer automatiquement le PC sur n'importe quel programme mis en mémoire (il suffit de placer le CALL &E24A au bon endroit).

La seule contrainte par rapport à un ARUN moins puissant est de devoir appuyer sur une autre touche après le ON, mais cela tout le monde le fait en allumant sa machine...

**ASTUCE 2****D'OÙ VIENT-IL ?**

Outre les 4 types de ROM, il y a deux sortes de PC-1500. Ceux provenant du Japon et ceux destinés à l'exportation.

L'ordinateur se sert de cette indication pour savoir s'il faut qu'il affiche le signe KATAKANA ou le mot SMALL.

Si le bit 3 de #&F00F est à 0, la machine provient du marché Japonais, et s'il est à 1, elle est destinée à l'exportation.

Christophe NICOLAS

# COMPTE BANCAIRE

**Ce programme est une petite aide pour gérer son compte bancaire. Il permet 16 retraits et 16 entrées, avec bien entendu une sauvegarde sur K7 et une sortie sur imprimante. Comme le logiciel comporte 13 fonctions différentes appelées toutes par DEF, il est préférable d'utiliser un normographe fourni avec le PC.**

Une fois le programme chargé, il faut effectuer un RUN et entrer la date. Le curseur revient et vous pouvez choisir la commande voulue.

DEF A : création d'entrées  
 DEF Z : lecture des entrées  
 DEF S : annulation d'une entrée  
 DEF X : annulation de toutes les entrées  
 DEF D : annulation d'un retrait  
 DEF C : annulation de tous les retraits  
 DEF F : création des retraits  
 DEF V : lecture des retraits  
 DEF N : imprime la date  
 DEF M : modification de la date  
 DEF SPC : sauvegarde des informations  
 DEF L : affiche le total des retraits, des entrées et le solde  
 DEF = : rappel d'anciennes données sur K7  
 A : créations d'entrées

Lorsque l'écran affiche « CODE 24 CART. » (ne pas faire ENTER), entrez directement ce à quoi correspond la somme en entrée. Le libellé doit comporter 24 caractères au plus. Ensuite, la machine affiche « SOMME: » et on entre la somme. Alors le PC affiche le nombre d'entrées restant. A la question « JE POURSUIS (O/N) », répondre O pour continuer la saisie des entrées.

La réponse N fait apparaître « CHOIX? », puis le curseur, et on peut choisir une autre option.

Z : permet la sortie sur imprimante du détail des entrées ou alors la lecture sur l'écran de toutes les entrées.

X : efface le contenu des mémoires correspondantes aux entrées.

S : lorsque l'on presse DEF S on voit apparaître sur l'écran « CLEF 7 CRTS MAX ». A ce moment, on entre un ensemble de 7 caractères correspondant à au moins une partie du code, ou des codes d'une ou plusieurs sommes classées dans la catégorie des entrées. Alors la machine recherche tous les codes identiques à cette clef, puis les affiche et demande une confirmation d'annulation. Les codes apparaissent tour à tour avec confir-

mation à chaque fois. Si la réponse est non, il continue ses recherches, et si l'ordinateur ne trouve rien il affichera « CLEF NON TROUVEE ».

Les commandes D, C, F et V correspondent pour les retraits respectivement aux commandes S, X, A et Z.

N : imprime la date enregistrée

M : permet de changer la date, mais comme la fonction AREAD est utilisée, il faut taper la date sous la forme JJ-MM-AA avant de faire DEF M.

SPC : enregistre toutes les données de la machine sous le nom correspondant à la date.

L : permet le rappel des données sauvegardées sur K7. Pour ce faire, positionner la bande de la cassette et taper la date correspondante à la question « CODE ».

= : cette option sort sur imprimante ou sur l'écran, le retrait total, l'apport total et la somme restante.

Terminons en signalant que le programme occupe 1664 octets à vide et 3412 une fois les variables dimensionnées. J'espère qu'il vous aidera à tenir vos comptes à jour.

Frédéric GENAUDET.

```

5:CLEAR :K=16: WAIT 90
10:BIM C$(0)*8,M$(0)*8,
    W$(K)*24,W(K),E$(K)*
    24,E(K),R$(K)*24,R(K)
    ),D$(0)*24,L$(0)
20:PAUSE "INITIALISATIO
N EFFECTUEE"
30:INPUT "DATE (JJ-MM-A
A) ":"FM$(0)
40:PRINT "CHOIX ?"
50:END
70:"A" BEEP 1: PRINT "A
PPORT CREATION"
80:X=X+1
90:IF X=K+1 BEEP 3:
    PRINT "SATURE": GOTO
    40
100:INPUT "CODE 24 CART.
",E$(X),"SOMME ":"E(
X)
110:PRINT "PLACE RESTANT
E :" ;K-X

```

```

120:INPUT " JE POURSUIS
(O/N)":Q$
130:IF Q$="0" THEN 80
140:GOTO 40
150:"Z" BEEP 1: PRINT "L
ECTURE APPORT"
160:INPUT "IMPRIMANTE (O
/N)":P$
170:IF P$="0" PRINT =
    LPRINT
180:FOR L=1 TO X
190:PRINT E$(L): PRINT "
    "E(L): PRINT
    "
200:NEXT L
210:PRINT = PRINT : GOTO
    40
215:REM NETTOYAGE
220:"X" FOR L=1 TO X:E$(L)
    )="":E(L)=0: NEXT
    L:X=0: GOTO 40
230:"C" FOR L=1 TO Y:R$(L)
    )="":R(L)=0: NEXT
    L:Y=0: GOTO 40
240:REM MODIFICATION
250:INPUT "CLEF 7 CRTS M
AX.":C$
290:J= LEN (C$)
300:FOR Z=1 TO U
310:FOR H=1 TO 24-J
320:IF MID$(W$(Z),H,J)=
    C$ THEN 300
330:NEXT H: NEXT Z
340:PRINT "CLEF NON TROU
VEE"
350:RETURN
360:"S" BEEP 1: PRINT "A
PPORT MODIFICATIONS"
:U=X: FOR L=1 TO X:W
    $(L)=E$(L):W(L)=E(L)
    : NEXT L
370:GOSUB 240:X=U: FOR L
    =1 TO U:E$(L)=W$(L):
    E(L)=W(L): NEXT L:
    GOTO 40
380:"D" BEEP 1: PRINT "R
ETRAIT MODIFICATIONS
":U=Y: FOR L=1 TO Y:
    W(L)=R$(L):W(L)=R(L)
    : NEXT L
390:GOSUB 240:Y=U: FOR L
    =1 TO U:R$(L)=W$(L):
    R(L)=W(L): NEXT L:
    GOTO 40
400:"F" BEEP 1: PRINT "R
ETRAIT CREATION"
410:Y=Y+1
420:IF Y=K+1 BEEP 3:
    PRINT "SATURE": GOTO
    40
430:INPUT "CODE 24 CART.
",R$(Y),"SOMME ":"E(
Y)
440:PRINT "PLACE RESTANT
E :" ;K-Y
450:INPUT " JE POURSUIS
(O/N)":Q$
460:IF Q$="0" THEN 410
470:GOTO 40
490:"V" BEEP 1: PRINT "L
ECTURE RETRAIT"
500:INPUT "IMPRIMANTE (O
/N)":P$
510:IF P$="0" PRINT =
    LPRINT
520:FOR L=1 TO Y
530:PRINT R$(L): PRINT "
    "R(L): PRINT
    "
540:NEXT L
550:PRINT = PRINT : GOTO
    40
560:"=":A=0:B=0: BEEP 1:
    PRINT "SOMME RES."

```

```

570:INPUT "IMPRIMANTE (O
/N)":P$
580:IF P$="0" PRINT =
    LPRINT
590:FOR F=1 TO X:A=A+E(F
    ): NEXT F
600:FOR F=1 TO Y:B=B+R(F
    ): NEXT F
610:PRINT "RETR.TOTAL ";B
620:PRINT "APPO.TOTAL ";
    A: PRINT "SOMME REST
    ANTE : " ;A-B
630:PRINT = PRINT : GOTO
    40
640:" " BEEP 1: PRINT "#M
    $(*),X,E$(*),E(*),Y,
    R$(*),R(*)
645:GOTO 40
648:"L" BEEP 1: INPUT "C
ODE ":"C$(0)
650:INPUT "#M$(*),X,E$(*),
    ,E(*),Y,R$(*),R(*)
655:IF C$(0)=M$(0) THEN
    40
656:GOTO 650
660:"M": AREAD M$(0):
    GOTO 40
670:"N": LPRINT M$(0):
    GOTO 40
800:PRINT W$(Z): PRINT W
    (Z)
810:INPUT "J ANNULE L EN
    TREE (O/N)":Q$
820:IF Q$<>"0" THEN 830
830:FOR F=Z TO U:W$(F)=W
    $(F+1):A(F)=W(F+1):
    NEXT F:U=U-1: RETURN

```

POUR TOUTE  
 CORRESPONDANCE  
 AVEC LE CLUB  
 JOINDRE  
 UNE ENVELOPPE  
 TIMBRÉE

# START MOGURA

**Ce jeu est un classique au Japon, et a déjà été publié pour les PC 1401/02. Il met en scène une taupe, qu'il faut assommer au moment où elle sort de son trou.**

On peut utiliser un PC 1350 de base sans carte, mais le programme doit être le premier et le seul en mémoire. Pour cette raison, faites un NEW avant de saisir le Basic, et ne rallongez pas le programme par des modifications. Le mode d'emploi est simple, RUN, puis choisir le niveau avec 10 ou 20 touches (1 ou 2). Le jeu se charge alors et l'ordinateur demande de choisir l'option musicale ou non, et la vitesse du jeu. A ce sujet, il vaut mieux prendre une valeur autour de 150 pour commencer. La partie débute alors et il faut assommer la taupe en appuyant le plus vite possible sur la touche correspondante à l'emplacement du trou d'où elle sort. A la fin du jeu, votre score s'affiche, et vous pouvez choisir de rejouer Y (avec choix du niveau), d'arrêter par N ou de continuer ce niveau par C.

Bonne chasse !

J.F.V.

```

1:REM ****
2:REM START MOGURA
3:REM by Mr Groove
4:REM ****
10:CLEAR :A$="Y":POKE
    $6000,29,36,43,50,57
    ,1,7,12,16,19,28,35,
    42,49,56,8,6,11,15,1
    8
20:POKE $6B20,0,128,128
    ,128,128,128,128,128
    ,8,0,224,16,80,16,80
    ,16,224,0,8,254,129
30:POKE $6B35,37,17,37
    ,129,254,0,129,195,10
    ,2,188,152,188,102,19
    5,129
50:CLS : WAIT 0: PRINT
    " START MOGURA V.13
    50": GPRINT "FE81251
    12581FE"
60:CURSOR (137,7):
    GPRINT "FE8125112581
    FE": INPUT " LEVEL?
    1:10Key Z:2:20Key",Z:
    ON Z GOTO 400,420
70:IF Z<1 OR Z>2 GOTO 5
    8
80:IF B=0 GOTO 50

```

```

90:CURSOR 24: PRINT "LE
    VEL "IZ: GOSUB 670
100:CLS : ON Z GOTO 120,
    110
110:CURSOR 24: PRINT "O
    W E R T Y U I O
    P": LINE (0,8)-(149
    ,150),X,BF
120:CURSOR 72: PRINT "A
    S D F G H J K L
    =": LINE (0,24)-(149
    ,31),X,BF
130:BEEP U/16: CALL &6990
    0: PRINT " GAME OV
    ER!!"
140:PRINT " POINT=";
    PEEK ($6300+16*(Z=2)
    )+"Key"
150:INPUT "REPLAY? (Y/N)":
    ?T$:Q=(T$="Y")+2*(T
    $"="N")+3*(T$="C"):
    ON Q GOTO 50,170,100
160:CURSOR 72: GOTO 150
170:END
180:DATA $10,$6B,B,$2,0,
    $52,$11,C,$52,$11,$2
    ,0,0,$24,$90,$18,$E4,
    $B1,$10,$6B,B
190:DATA $57,$42,$52,$67
    ,,$33,$28,$2,$37,$11,
    ,0,$2,$10,$8,0,I,$1F,$7
    ,8,$69,$99,$78,$69
200:DATA $88,$4E,$FF,$43
    ,,$29,$4,$78,$69,$88,
    $64,E,$67,K,$3A,$3,$6
    75,F,$18,$6B,E
210:DATA $52,$3,$6C,$E2,
    $97,$24
220:DATA $10,$6B,G,$52,$
    2,$19,$78,$69,$52,$2
    ,,$22,$78,$69,$52,$2D
    ,,$40,$78,$69,H,$0
230:DATA V,$E4,$36,$8F,&
    DB,$10,$6B,G,$57,$07
    ,,$38,$5,$41,$29,&I,$
    37,$5B,$5B,$2,$2B
240:DATA $78,$69,H,$10,&
    6B,C,$57,$42,$52,$2,
    $9,$12,$5F,$10,$69,&
    TE,$61,U,$DF,$4E
250:DATA $0,$60,$1,$DF,&
    42,$52,$29,&B,$2D,$7
    6,$10,$6B,I,$84,$1B,
    $85,$60,$9F,$61
260:DATA $90,$25,$10,$6B
    ,I,$84,$1B,$37,$2,$2
    3,$83,$70,$E2,$B5,$2,
    $0,$3,$6B,$E2,$97
270:DATA $3,$9,$24,$30,&
    59,$26,$50,$03,$29,&
    5
280:DATA $86,$2,$32,$DB,
    $84,$62,$1,$38,$8,$8
    6
290:DATA $2,$22,$DB,$87,
    $70,$2,$84,$63,D,$29
    ,J,$37,$34,$10,$6B,E
    ,,$57,$3,$6B,$E2
300:DATA $B5,$5B,$26,$79
    ,,$69,$99,-9
400:B=&C,C=&B,D=9,E=&F,
    F=6,G=&E,H=&C5,I=&10,
    J=&1E,K=&A,L=&B:
    RESTORE :S=&68FF:
    GOSUB 700
410:GOSUB 670: CLS :
    GOTO 120
420:B=&10,C=&1B,D=&13,E=
    &1F,F=&C,G=&1E,H=&E2
    ,I=&15,J=&3B,K=&13:
    RESTORE :S=&68FF:
    GOSUB 700
430:POKE $69AF,$84,$63,9
    ,,$3A,$19,$86,2,$14,&
    DB,$84,$62,1,$38,$8,&
```

```

    86,2,4,&DB,$87,$70
440:POKE &6903,2,$63,$7A
    ,,$28,3,$71,&A,$20,$81
1: RESTORE 280:S=&69
    CB: GOSUB 700
450:GOSUB 670: CLS :
    GOTO 110
670:CURSOR 32: PRINT "
    ": INPUT "
    MUSIC(Y/N):?A$=U=16
    *(A$="Y")+U*(A$=""):
    POKE &697B,U
680:CURSOR 56: PRINT A$:
    INPUT "SPEED 1-255:";
    ?V: POKE $6956,V
690:CLS : CURSOR 27:
    PAUSE "START MOGURA": RETURN
700:CURSOR 24: PAUSE "L
    EVEL "IZ:....LOADIN
    G"
710:READ M: IF M>5 LET
    S=S+1: POKE S,M:
    GOTO 710
720:RETURN

```

## EDIT LM

**Beaucoup de nos lecteurs regrettent de ne pas avoir d'outil rapide et efficace pour entrer les programmes en langage Machine. Cet utilitaire écrit pour un PC 1350 est relogable et facilite l'introduction des Dumps que nous publions régulièrement. Et ceci, avec un contrôle permanent du Checksum, un affichage sur 4 lignes et un système d'édition avec curseur et touches redéfinies.**

Entrez le programme Basic fourni, et vérifiez-le attentivement ; une petite erreur et c'est le plantage assuré. Tapez RUN, puis une fois revenu sous Basic sauvegardez le LM sur K7 par : CSAVE M «EDITLM»;&6A00,&6BC9 ENTER.

Vous pourrez le recharger par la suite en mémoire où vous voudrez. Par exemple en &3000 par : CLOAD M «EDITLM»;&3000. Soit pour une adresse de type &XXYY par CLOAD M &XXYY. L'adresse d'exécution (pour l'instant &6A00) deviendra XXYY. Le

programme est dit relogable, donc indépendant d'une quelconque adresse. Il est écrit en &6A00, mais fonctionne n'importe où en RAM. Cela est pratique pour le cas où nous aurions à entrer en mémoire un programme qui irait « écraser » EDITLM en écrivant entre &6A00 et &6BC9 : adresses de début et de fin de cet utilitaire. Au fait, le Basic n'est plus nécessaire une fois le RUN effectué. Vous pouvez faire NEW, le LM ne sera pas effacé.

L'utilisation est simple. Pour écrire en mémoire à partir d'une adresse &NNMM, il suffit de taper (en Mode RUN, PRO ou RSV) CALL &XXYY, &NNMM (ici XXYY = 6A00). Surtout respectez bien cette syntaxe. L'adresse d'exécution doit être exprimée en Hexa (&XXYY). Par exemple : CALL &6A00, &6800. L'écran affichera un Dump sur 4 lignes, présenté sous forme d'une adresse (4 car), un espace, 8 octets accolés (8 x 2 car.), deux points et le checksum (2 car.). Un pavé clignotant est placé sur la deuxième ligne, c'est le curseur d'édition que l'on peut déplacer uniquement sur cette ligne : le Scrolling est automatique. L'adresse à gauche du curseur est l'adresse courante, soit NNMM:&6800 pour l'exemple du départ.

Les touches flèches Haut, Bas, Droite, Gauche de déplacement du curseur sont opérationnelles. Comme le Dump est en Hexadécimal, seules les touches 0 à 9 et A à F ont un effet.

Tapez sur ces touches, et si vous êtes en RAM, vous verrez immédiatement la modification en mémoire et l'influence sur le Checksum. Notons l'autorépétition des touches. Hormis cela, pour faciliter la frappe, les touches (/ \* - + . sont redéfinies comme étant aussi A B C D E F).

Alors que BRK permet de sortir de l'éditeur. Vous pouvez donc «dumper» la mémoire aussi bien RAM que ROM. Vous constaterez que pour la ROM aucune modification n'est possible.

De plus, on peut sortir sur l'imprimante CE-126P avec la syntaxe suivante : CALL &XXYY,&NNMM,&TTZZ. Où &XXYY est l'adresse d'exécution de EDITLM, &NNMM l'adresse de début du Dump et &TTZZ l'adresse de fin de ce même Dump. Pour faire un Dump mémoire du programme EDITLM, faire : CALL &6A00,&6A00,&6BC9. On peut donc obtenir sur l'imprimante un Dump de l'adresse &NNMM jusqu'à &TTZZ.

Bon LM, et n'écrivez pas n'importe où au hasard, surtout pas sur le programme EDITLM, où gare au BUSY.

Jean-François VIGNAUD

```

1:REM EDITLM
10:POKE &6A00,&03,&6E,&
    &02,&B7,&E2,&97,&EA,&
    &B8,&F6,&3A,&AB,&13,&
    18,&0A,&AD,&60
20:POKE &6A10,&00,&24,&
    &67,&0D,&38,&0C,&EA,&
    &B8,&F6,&3A,&AB,&13,&
    18,&0A,&AD,&61
30:POKE &6A20,&01,&AA,&
    &02,&06,&DB,&FC,&1C,&
    FE,&0C,&AB,&03,&00,&
    02,&08,&15,&84
40:POKE &6A30,&13,&28,&
    &0A,&E2,&AA,&02,&03,&
    34,&85,&2C,&21,&84,&
    2C,&1E,&06,&02
50:POKE &6A40,&07,&34,&
    A0,&60,&00,&82,&13,&
    04,&0A,&04,&00,&00,&
    88,&35,&88,&59
60:POKE &6A50,&A0,&44,&
    88,&2C,&07,&2F,&11,&
    02,&3A,&26,&A0,&20,&
    DA,&59,&80,&DB
70:POKE &6A60,&59,&58,&
    64,&0F,&74,&30,&67,&
    3A,&3A,&03,&74,&07,&
    26,&59,&64,&0F
80:POKE &6A70,&74,&30,&
    67,&3A,&3A,&03,&74,&

```

```

    07,&26,&83,&63,&04,&
    39,&3F,&63,&05
90:POKE &6A80,&39,&46,&
    63,&08,&39,&30,&2F,&
    4F,&AD,&63,&00,&38,&
    29,&10,&6D,&18
100:POKE &6A90,&90,&00,&
    17,&18,&78,&00,&54,&
    A8,&03,&00,&02,&08,&
    14,&84,&13,&2B
110:POKE &6AA0,&04,&82,&
    13,&28,&0A,&84,&15,&
    2A,&02,&37,&85,&63,&
    00,&29,&7F,&84
120:POKE &6AB0,&63,&08,&
    2B,&84,&37,&FD,&D9,&
    02,&03,&E4,&B3,&AF,&
    60,&00,&2C,&05
130:POKE &6AC0,&6B,&01,&
    38,&62,&AF,&63,&00,&
    28,&14,&61,&01,&E2,&
    A3,&03,&00,&AA
140:POKE &6AD0,&59,&74,&
    17,&84,&14,&24,&AE,&
    DB,&00,&F9,&2C,&07,&
    60,&00,&AE,&59
150:POKE &6AE0,&80,&DB,&
    03,&70,&02,&3A,&E2,&
    B5,&AA,&59,&87,&67,&
    05,&3A,&07,&75
160:POKE &6AF0,&05,&70,&
    02,&2D,&09,&34,&86,&

```

```

    70,&06,&2F,&03,&80,&
    63,&F9,&28,&00
170:POKE &6B00,&03,&83,&
    02,&DD,&E2,&97,&2C,&
    16,&2D,&E0,&2D,&DC,&
    03,&7F,&02,&F0
180:POKE &6B10,&E2,&97,&
    03,&00,&02,&04,&34,&
    80,&59,&84,&14,&2F,&
    03,&40,&02,&04
190:POKE &6B20,&34,&25,&
    26,&2F,&03,&03,&84,&
    02,&03,&E2,&97,&E4,&
    36,&84,&44,&24
200:POKE &6B30,&67,&04,&
    39,&2B,&67,&05,&28,&
    09,&A8,&03,&00,&02,&
    08,&14,&2D,&35
210:POKE &6B40,&67,&0E,&
    28,&0C,&AA,&70,&01,&
    63,&16,&29,&84,&71,&
    10,&2D,&16,&67
220:POKE &6B50,&0F,&28,&
    0C,&AA,&71,&01,&63,&
    05,&29,&4F,&70,&10,&
    2D,&55,&00,&8A
230:POKE &6B60,&67,&28,&
    38,&39,&40,&67,&2F,&
    38,&34,&40,&67,&2A,&
    38,&2F,&40,&67
240:POKE &6B70,&2D,&38,&
    2A,&40,&67,&2B,&38,&

```

```

    25,&40,&67,&2E,&38,&
    20,&67,&30,&3A
250:POKE &6B80,&13,&67,&
    3A,&2A,&07,&75,&30,&
    80,&DB,&2C,&12,&67,&
    41,&3A,&05,&67
260:POKE &6B90,&47,&3A,&
    06,&6B,&08,&39,&2D,&
    37,&75,&37,&80,&DB,&
    86,&13,&28,&0A
270:POKE &6BA0,&03,&00,&
    AA,&59,&75,&06,&D1,&
    D2,&74,&08,&86,&14,&
    07,&F4,&19,&AA
280:POKE &6BB0,&62,&01,&
    38,&0A,&24,&64,&F0,&
    80,&47,&59,&26,&2D,&
    78,&80,&59,&58
290:POKE &6BC0,&DB,&24,&
    64,&0F,&80,&47,&59,&
    26,&2D,&85

```

tous les PC

## ON

**Après avoir vu les caractéristiques générales du traitement alphanumérique, nous verrons cette fois les traitements spécifiques à ce type de variable.**

### LEN

Nous allons tout d'abord étudier un cas simple, LEN, qui signifie longueur. Vous remarquerez qu'il n'y a pas le signe \$ après les lettres, la réponse n'est donc pas une chaîne de caractères mais le nombre de signes dans EX\$, avec une syntaxe de ce type :

VAR = LEN EX\$

Où EX\$ est une expression qui donnera pour résultat une chaîne de caractères. Cela peut être le résultat d'un calcul.

VAR est une variable numérique.

On remarquera qu'il n'est pas obligatoire de mettre des parenthèses entre le nom de EX\$ sauf si cela peut prêter à confusion, par exemple :

A = LEN A\$ + B\$

Ce cas donne une erreur 17 (PC-1500) car l'évaluateur travaille ainsi : A\$ → LEN → nombre + B\$

D'où erreur car on ne peut additionner un nombre avec une variable ALPHA. On fera dans ce cas :

A = LEN (A\$ + B\$)

L'évaluateur travaillera ainsi :

A\$ + B\$ → LEN → nombre → stockage dans A

Voyons un exemple concret :

A\$ = «CLUB» LEN A\$ = 4

B\$ = «pocket» LEN B\$ = 6

LEN (A\$ + «—» + B\$) = 4 + 1 + 6 = 11

Nous allons voir en bloc trois fonctions qui travaillent de manière très proche :

RIGHT\$ (EX\$,n)

LEFT\$ (EX\$,n)

MID\$ (EX\$,n,m)

où n et m sont des nombres dont nous allons voir les domaines de validité.

### RIGHT\$ - LEFT\$

Dans tous les cas la réponse est une chaîne alphanumérique. Ces deux

fonctions permettent d'extraire une chaîne alphanumérique d'une autre chaîne.

RIGHT\$ (EX\$,n) donne pour résultat les n caractères les plus à droite de l'expression EX\$.

LEFT\$ (EX\$,n) donne les n caractères les plus à gauche.

On remarque que n peut prendre les valeurs de 0 à 80 inclus, quelque soit la longueur de EX\$. Ainsi, si n vaut 0, le résultat est toujours une chaîne vide donc de longueur nulle : LEN RIGHT\$ (EX\$,0) = 0

De même, si n est plus grand ou égal que la longueur de EX\$, la réponse sera EX\$ :

RIGHT\$ (EX\$,80) = EX\$

### MID\$

La fonction MID\$ est plus compliquée mais aussi plus puissante. Elle permet de définir à la fois la longueur, mais aussi le point de départ de la chaîne à extraire. Remarquez que l'on ne définit pas son sens qui est toujours de la gauche vers la droite. La syntaxe est la suivante :

A\$ = MID\$ (EX\$,n,m)

Et dans A\$, il y aura les m caractères de EX\$ à partir de la position n. On remarque que n peut prendre les

valeurs de 1 à 80 inclus et m les valeurs de 0 à 80 quelque soit la longueur de EX\$. Si on demande plus de caractères que ce qu'EX\$ n'en contient à partir de sa position n, la chaîne extraite n'aura que ce que EX\$ peut effectivement fournir :

EX\$ = «ESSAI»

MID\$ (EX\$,4,10)

Soit prendre 10 caractères à partir du A, ne répondra que AI, soit deux caractères.

Nous allons voir quelques applications pratiques :

### ALIGNER LES CARACTÈRES

C'est souvent nécessaire pour le travail sur imprimante. Il faut en effet jouer sur l'esthétisme et faire des colonnes régulières. Les variables doivent avoir toutes la même longueur, quelque soit leur contenu, quitte à rajouter des espaces à la fin.

Soit une variable A\$ ayant LEN A\$ caractères. On veut qu'elle en ait 10. Il faudra donc compléter par des espaces à la droite des caractères existants.

Il faudra vérifier que A\$ n'ait pas plus de 10 caractères sinon on aboutit à une erreur :

A\$ = A\$ + LEFT\$ («\_\_\_\_\_»,  
10-LEN A\$)

Étudions un peu cette dernière ligne : 10-LEN A\$ donne pour réponse le nombre de caractères manquant à A\$ pour avoir 10 caractères.

C'est à cause de cette formule qu'il ne faut pas que A\$ ait plus de 10 caractères, par exemple 12, car  $10 - 12 = -2$  et un LEFT\$ (EX\$, -2) provoque une erreur 19 (PC-1500). On prendra donc de

la chaîne comportant 10 espaces, autant d'espaces que 10-LEN A\$. Nous verrons une prochaine fois que l'on pourrait faire plus simple avec les instructions STRING\$ et SPACE\$.

#### REPLACER UN CARACTÈRE

Soit une chaîne EX\$, si l'on veut remplacer le caractère n par un autre

caractère, par exemple «A», on fera ainsi : on prend tous les caractères avant le n, on ajoute «A» et on rajoute tous ceux situés après le n. Cela donnera :

LEFT\$ (EX\$, n - 1) + «A» + RIGHT\$ (EX\$, LEN EX\$ - n)

et tout marche parfaitement même pour les caractères situés aux extrémi-

tés car un LEFT\$ (EX\$, 0) donne une chaîne vide.

La prochaine fois, nous traiterons de STR\$, VAL et nous reviendrons sur STRING\$ et SPACE\$.

A bientôt !

Le Club

## GRAPHE LIGNE

**Pour continuer la série entamée dans le numéro 20, voici un logiciel capable de tracer simultanément plusieurs courbes. Cela permet, par exemple, d'apprécier la croissance des chiffres de vente de différents produits. Ils sont matérialisés par des lignes de couleurs et pointillés différents.**

Toujours extrait du même livre japonais, ce programme est compatible avec les PC 1350/60, 1450/60 et 2500 munis d'une carte 8 ko au moins, et d'une table traçante CE-516P ou CE-140P. Pour les PC 2500, l'imprimante intégrée fonctionne en supprimant la ligne 750. L'exemple de tracé fourni avec son tableau des valeurs vous donne un aperçu du résultat obtenu. Regardons pas à pas chacune des quatre parties composant le programme.

#### LA SAISIE

Elle est obligatoire pour une première introduction de valeurs. Par la suite, si des données sont en mémoire, on peut corriger, refaire un graphique ou sauvegarder ces données. Ou encore charger des valeurs préalablement mises sur cassette. Ce module démarre par RUN ou DEF A. A la question «Colonne=», indiquer le nombre de colonnes, (1 à 8), par exemple 6. Puis on entre le nom de chacune des colonnes (Nom X col(1) Janvier, Février, etc., jusqu'à Juin). La question «Ligne=» demande donc le nombre de lignes (2 à 10) : 5 sur l'exemple. Comme pour les colonnes, donner le nom pour chaque ligne (1980...1984). Ensuite la saisie des valeurs commence. On entre cette valeur après l'affichage du nom de la colonne, suivi de deux points et du nom de la ligne. Un ? signale cette demande, et la saisie s'effectue ligne par ligne (270, 300, etc. 156). A la fin de

cette saisie, la machine demande si on veut le tableau des valeurs entrées sur papier («Tableau (O/N)»). Si oui, répondre O ou o, autrement N ou n. Dans le cas où une erreur de frappe est décelée nous pouvons corriger grâce au module suivant.

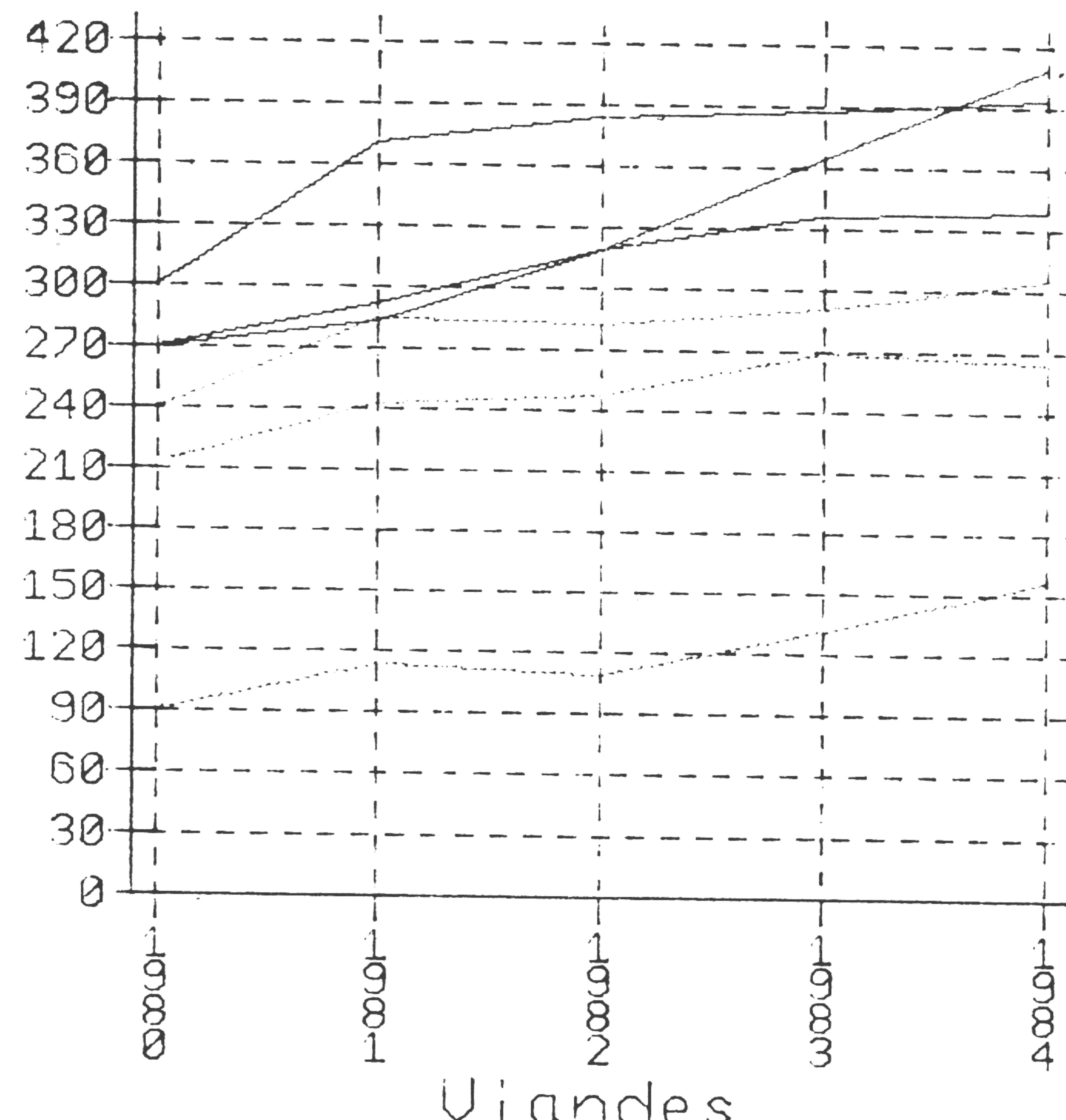
#### LA CORRECTION

Elle est permise afin de modifier un nom de colonne ou de ligne, ou une valeur numérique erronée. Attention : chaque nombre est limité à 8 caractères signe (-) compris, et seule la partie entière est conservée. Pour les noms de colonne ou ligne, un maximum de 7 caractères est pris en compte. On change le nom de la colonne 3 en entrant à la question «Variables =» C(3). Soit pour le nom d'une colonne i : C(i). De même pour le nom d'une ligne j : L(j). On modifie une valeur numérique en précisant le numéro de colonne et de ligne. Exemple : V(2,1) indique la colonne 2 et la ligne 1, soit dans l'exemple donné la valeur 300. Donc V(i,j) dans un cas général. On sort de cette partie par BRK ou DEF.

#### LA GESTION K7

DEF F affiche la question «Gestion K7(E/S)». On répond E (ou e) comme Entrée pour charger en mémoire des valeurs, ou S (ou s) comme Sortie pour sauvegarder un tableau sur cassette.

## ACHATS



Viandes

Janvier

Mars

Mai

Février

Avril

Juin

	Janvier	Février	Mars	Avril	Mai	Juin
1980	270	300	270	213	241	90
1981	284	371	292	243	285	113
1982	320	385	320	248	283	109
1983	365	388	337	269	291	131
1984	410	394	339	265	305	156

## LE TRACÉ

Nous avons deux points d'entrée dans ce module : DEF C permettant d'entrer les titres, et DEF D de refaire un tracé en conservant les titres déjà saisis. «Titre=» demande le titre général du graphique, ici «ACHATS», «Titre(1)=» correspond à l'inscription au bas du tracé, relative aux lignes, soit «Vian-des» dans l'exemple. Alors que «Titre(2)=» demande une information sur les colonnes, c'est-à-dire l'unité utilisée ou la monnaie, etc... Ici «Frs». Mais vous êtes libres de choisir comme il vous plaira. Le tracé débute après la saisie de ces dernières précisions. A ce stade, on est au niveau de l'entrée par DEF D. On supprime l'instruction OPEN pour le PC-2500 avec son imprimante, et on détermine le OPEN requis en manuel avant de faire DEF C ou D.

Pour la CE-140P, on garde les valeurs d'origine, et pour la CE-516P on peut prendre OPEN «1200,N,8,1,A,L». Une fois le graphique achevé, l'ordinateur rend la main.

La prochaine fois, nous verrons un autre type de graphique, mais en attendant faites-nous part de vos souhaits, et n'hésitez pas à nous envoyer vos réalisations.

Jean-François VIGNAUD

```

10. A1 WAIT 64. CLEAR CONSOLE 39
20. SP$=1. T$=0. Y$= CHR$ 27
30. INPUT "Colonnes : KK
40. IF KK<1 OR KK>8 BEEP 2. GOTO 30
50. DIM X$(KK),T$(30*22+KK)
60. FOR I=1 TO KK
70. PAUSE "Nom Colonne" . STR$ I. Y$
80. INPUT Y$11
90. IF LEN Y$11>2 BEEP 2. GOTO 80
100. NEXT I
110. INPUT "Lignes" . YK
120. IF YK>20 BEEP 2. GOTO 110
130. DIM Y$(YK),DD$YK,YK
140. FOR I=1 TO YK
150. PAUSE "Nom ligne" . STR$ I. YI
160. INPUT Y$11
170. IF LEN Y$11>2 BEEP 2. GOTO 160
180. NEXT I
190. FOR YI=1 TO YK. FOR KI=1 TO KK
200. PAUSE LEFT$ (Y$YI+SP$ ,2) . LEFT$ (Y$YI+SP$ ,2)
210. D$= INPUT D
220. D$= INT D. IF LEN STR$ D$ BEEP 2.
GOTO 210
230. DD$YK,YK
240. NEXT K. NEXT Y
250. K$=1. INPUT "Tableau" . MN
260. COSUB 220. IF K$=1 THEN 250
270. IF K END
280. COSUB 250
290. S1= INT (KK*4*YK). X$0
300. FOR O=1 TO S1
310. S2=3. IF O=S1 LET S2=KK-4*O+4
320. LPRINT "M0.0"
330. LPRINT " "
340. K2=108*Y$2+1. LPRINT "D1-K2.0"
350. FOR I=0 TO S2-1. LPRINT "M1-K2+1" 10
360. LPRINT "D1-K2+1" 10. NEXT I
370. FOR I=0 TO S2. LPRINT "M1-12+108" 10
380. LPRINT "P1-K$XH1". NEXT I
390. FOR YI=1 TO YK. LPRINT "M0.0"
400. LPRINT "D1-K2.0"
410. LPRINT "D1-K2+1" 10. NEXT I
420. FOR I=0 TO S2. LPRINT "M1-108" 10
430. LPRINT "P1-Y$YI". GOTO 45

```

```

440. LPRINT "P1-Y$YI". GOTO 45
450. NEXT I. NEXT Y
460. LPRINT "M1-K2.0" 10. LPRINT "D0.0"
470. LPRINT " "
480. B$=$1. INPUT "Unité de fréq." . K$
490. IF LEFT$ (K$,2)=UCM THEN 540
500. IF LEFT$ (K$,2)=UCL THEN 590
510. IF LEFT$ (K$,2)=UCU THEN 850
520. IF K$="" THEN 250
530. BEEP 2. GOTO 480
540. IF VAL MID$ (K$,3,1)>16. IF I<1 OR I>X
K. THEN 530
550. PAUSE "CC" . STR$ I. Y$11
560. INPUT X$11
570. IF LEN X$11>2 BEEP 2. GOTO 560
580. GOTO 480
590. W$=LEN K$-L$1. IF W$>LET L$2
600. IF VAL MID$ (K$,3,1)>16. IF KK1 OR KK
K. THEN 530
610. PAUSE "CC" . STR$ I. Y$11
620. INPUT Y$11
630. IF LEN Y$11>2 BEEP 2. GOTO 620
640. GOTO 480
650. Y$=VAL MID$ (K$,3,1). IF KK1 OR KK
K. THEN 530
660. W$=LEN K$-L$1. IF W$>LET L$2
670. IF VAL MID$ (K$,3,1)>16. IF KK1 OR YY
K. THEN 530
680. PAUSE "CC" . STR$ X. Y$11. STR$ Y$11
. STR$ DD$YK,YD
690. INPUT D
700. D$= INT D. IF LEN STR$ D$ BEEP 2.
GOTO 690
710. DD$YK,YD. GOTO 480
720. IF K$=0 OR K$=1. LET K$=0. RETURN
730. IF K$=1 OR K$=0. LET K$=1. RETURN
740. BEEP 2. K$=1. RETURN
750. CLOSE . OPEN
760. LPRINT E$. P1. LPRINT "M1-Y$11"
770. LPRINT E$. P1. LPRINT " "
780. D$= INPUT "Titre" . T$=D$0
790. IF LEN T$0>10 BEEP 2. GOTO 790
800. INPUT T$=E$10. T$=E$0
810. IF LEN T$0>10 BEEP 2. GOTO 800
820. INPUT T$=E$20. T$=E$0
830. IF LEN T$0>10 BEEP 2. GOTO 820
840. P1. SING= COLOR 790. YK=480. ZK=520
ZK=500
850. YN=10000000. KK=YN
860. FOR YI=1 TO YK. FOR KI=1 TO KK. D$=DD$YK
. YD
870. IF DD$YK LET KK=0
880. IF DD$YK LET YY=0
890. NEXT K. NEXT Y
900. YY=40. ZY
910. MS$=1. LEN STR$ YK$12. T$=MS$12. LET
MS$=0
920. IF 2*MS$12>KK LET ZK=KK. MS$=0
930. PK$=KK. ZK=5
940. YY=YK. ZK=KK. ZD$=YY
950. D$= INT (D0. KK-1)*1000. T$=PK$*1000
960. IF PK$=12 LET PK$=0. GOTO 1000
970. IF PK$=20 LET PK$=0. GOTO 1000
980. IF PK$=30 LET PK$=0. GOTO 1000
990. P1. D$=0
1000. M$=PK$*1000. T$=PK$*1000. Y$=PK$*1000
1010. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1020. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1030. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1040. Y$=ME. PA(Y$11)
1050. RM A. TRACE X
1060. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1070. LPRINT E$. P1. LPRINT "M1-Y$11"
1080. LPRINT E$. P1. LPRINT "M1-Y$11"
1090. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1100. FOR LL=0 TO RP1. KMM$=LL. Y$=PK$*1000
1110. IF SW=1. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1120. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1130. SW=SW. NEXT LL
1140. LPRINT "L0" . LPRINT "M0.0"
1150. KM=XH*10. YU=0. YD=0. NY=50
1160. FOR KI=1 TO YK. KM=XH
1170. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1180. X=KM-4. Y=12. T$=C30=E$*KK. ST=14
1190. COSUB 120
1200. IF YK NY LET NY=Y
1210. NEXT K
1220. LPRINT E$. P1. MS$=LEN T$(10). X=
INT ((ZX-MS$18)/20). Y=NY-25
1230. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1240. LPRINT E$. P1. LPRINT "M1-Y$11"
1250. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1260. FOR KI=1 TO YK
1270. IF SW=1. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1280. LPRINT "M1-KM-YD" . LPRINT "M1-Y$11"
1290. X=KM. SW=SW. NEXT K
1300. LPRINT "L0" . LPRINT "M1-Y$11"
1310. RP$= INT ((MX-MN)/YD)

```

```

1320. FOR LL=0 TO RP1. KMM$=LL. YM=YM+Y
4
1330. MS=C. LEN STR$ KM. X=-MS*12-8. YM=YM-
6
1340. LPRINT "M1-KM-YM" . LPRINT "M1-Y$11"
STR$ K
1350. LPRINT "M1-KM-YM" . LPRINT "M1-Y$11"
1360. IF KKNX LET NX=K
1370. NEXT LL
1380. LPRINT E$. P1. T$=(B1-T$C2)/Y= INT
(ZY-(ZY-MS*20)/20). KKNX=30. ST=20
1390. COSUB 1230
1400. LPRINT E$. P1. LPRINT "M0.0"
1410. CL=CL+1. LPRINT "L0" . LPRINT E$.
"1"
1420. FOR R=1 TO KK. X=10. Y= INT ((DDCR,
13-MN)*PA1. HACR=CL*100+L
1430. LPRINT "M1-KM-YM" . LPRINT "M1-Y$11"
1440. FOR G=2 TO YY. KEX=KK. Y= INT ((DDCR,
R,G)-MN)*PA1
1450. LPRINT "M1-KM-YM" . LPRINT "M1-Y$11"
1460. CL=CL+1. IF CL=4 LET CL=0. CL=1.
LPRINT "L0" . L
1470. LPRINT E$. STR$ CL. NEXT R
1480. Y-NY-65. X= INT (ZK/6). SW=1. FOR K
=1 TO KK
1490. IF SW=1. LPRINT "M1-KM-YM" . LPRINT "M1-Y$11"
GOTO 1510
1500. LPRINT "M1-ZK/2" . L
1510. CL= INT (ZK/100). LPRINT E$. CL
LPRINT "L1-ZK/100"
1520. LPRINT "L1-ZK/100" . LPRINT "R20" . L
LPRINT "P1-Y$11"
1530. IF SW=1. LET Y=Y-40
1540. SW=SW. NEXT K
1550. LPRINT "L0" . LPRINT E$.
0
LPRINT "M1-Y$11"
1560. LPRINT "M1-Y$11" . END
1570. F
1580. K$=$1. INPUT "Gestion KZ(E/S)" . K$
1590. IF K$=1 OR K$=0. THEN 1660
1600. IF K$=1 OR K$=0. THEN 1620
1610. BEEP 2. GOTO 1580
1620. COSUB 1210
1630. PRINT "#SENDATA" . KK, YK
1640. PRINT "#X$(*),Y$(*),T$(*),DD(*)"
1650. END
1660. COSUB 1210
1670. CLEAR . KK=0. YK=0. INPUT "#SENDATA"
. KK, YK
1680. DIM T$(3)*22. X$(KK), Y$(YK), DD(KK),
YK1, HAC(KK)
1690. INPUT #X$(*), Y$(*), T$(*), DD(*) . E$.
" " . CHR$ 27. SP$=
1700. END
1710. INPUT "K2 OK, PRESS ENT" . K$.
GOTO 1710
1720. RETURN
1730. MS$= LEN T$(3)
1740. FOR KK=1 TO MS. Y=Y-ST
1750. LPRINT "M1-KM-YM" . T$=KK
1760. LPRINT "M1-KM-YM" . T$=KK
1770. NEXT KK. RETURN

```

```

1780. END
1790. LPRINT "M1-KM-YM" . T$=KK
1800. LPRINT "M1-KM-YM" . T$=KK
1810. LPRINT "M1-KM-YM" . T$=KK
1820. LPRINT "M1-KM-YM" . T$=KK
1830. LPRINT "M1-KM-YM" . T$=KK
1840. LPRINT "M1-KM-YM" . T$=KK
1850. LPRINT "M1-KM-YM" . T$=KK
1860. LPRINT "M1-KM-YM" . T$=KK
1870. LPRINT "M1-KM-YM" . T$=KK
1880. LPRINT "M1-KM-YM" . T$=KK
1890. LPRINT "M1-KM-YM" . T$=KK
1900. LPRINT "M1-KM-YM" . T$=KK
1910. LPRINT "M1-KM-YM" . T$=KK
1920. LPRINT "M1-KM-YM" . T$=KK
1930. LPRINT "M1-KM-YM" . T$=KK
1940. LPRINT "M1-KM-YM" . T$=KK
1950. LPRINT "M1-KM-YM" . T$=KK
1960. LPRINT "M1-KM-YM" . T$=KK
1970. LPRINT "M1-KM-YM" . T$=KK
1980. LPRINT "M1-KM-YM" . T$=KK
1990. LPRINT "M1-KM-YM" . T$=KK
2000. LPRINT "M1-KM-YM" . T$=KK
2010. LPRINT "M1-KM-YM" . T$=KK
2020. LPRINT "M1-KM-YM" . T$=KK
2030. LPRINT "M1-KM-YM" . T$=KK
2040. LPRINT "M1-KM-YM" . T$=KK
2050. LPRINT "M1-KM-YM" . T$=KK
2060. LPRINT "M1-KM-YM" . T$=KK
2070. LPRINT "M1-KM-YM" . T$=KK
2080. LPRINT "M1-KM-YM" . T$=KK
2090. LPRINT "M1-KM-YM" . T$=KK
2100. LPRINT "M1-KM-YM" . T$=KK
2110. LPRINT "M1-KM-YM" . T$=KK
2120. LPRINT "M1-KM-YM" . T$=KK
2130. LPRINT "M1-KM-YM" . T$=KK
2140. LPRINT "M1-KM-YM" . T$=KK
2150. LPRINT "M1-KM-YM" . T$=KK
2160. LPRINT "M1-KM-YM" . T$=KK
2170. LPRINT "M1-KM-YM" . T$=KK
2180. LPRINT "M1-KM-YM" . T$=KK
2190. LPRINT "M1-KM-YM" . T$=KK
2200. LPRINT "M1-KM-YM" . T$=KK
2210. LPRINT "M1-KM-YM" . T$=KK
2220. LPRINT "M1-KM-YM" . T$=KK
2230. LPRINT "M1-KM-YM" . T$=KK
2240. LPRINT "M1-KM-YM" . T$=KK
2250. LPRINT "M1-KM-YM" . T$=KK
2260. LPRINT "M1-KM-YM" . T$=KK
2270. LPRINT "M1-KM-YM" . T$=KK
2280. LPRINT "M1-KM-YM" . T$=KK
2290. LPRINT "M1-KM-YM" . T$=KK
2300. LPRINT "M1-KM-YM" . T$=KK
2310. LPRINT "M1-KM-YM" . T$=KK
2320. LPRINT "M1-KM-YM" . T$=KK
2330. LPRINT "M1-KM-YM" . T$=KK
2340. LPRINT "M1-KM-YM" . T$=KK
2350. LPRINT "M1-KM-YM" . T$=KK
2360. LPRINT "M1-KM-YM" . T$=KK
2370. LPRINT "M1-KM-YM" . T$=KK
2380. LPRINT "M1-KM-YM" . T$=KK
2390. LPRINT "M1-KM-YM" . T$=KK
2400. LPRINT "M1-KM-YM" . T$=KK
2410. LPRINT "M1-KM-YM" . T$=KK
2420. LPRINT "M1-KM-YM" . T$=KK
2430. LPRINT "M1-KM-YM" . T$=KK
2440. LPRINT "M1-KM-YM" . T$=KK
2450. LPRINT "M1-KM-YM" . T$=KK
2460. LPRINT "M1-KM-YM" . T$=KK
2470. LPRINT "M1-KM-YM" . T$=KK
2480. LPRINT "M1-KM-YM" . T$=KK
2490. LPRINT "M1-KM-YM" . T$=KK
2500. LPRINT "M1-KM-YM" . T$=KK
2510. LPRINT "M1-KM-YM" . T$=KK
2520. LPRINT "M1-KM-YM" . T$=KK
2530. LPRINT "M1-KM-YM" . T$=KK
2540. LPRINT "M1-KM-YM" . T$=KK
2550. LPRINT "M1-KM-YM" . T$=KK
2560. LPRINT "M1-KM-YM" . T$=KK
2570. LPRINT "M1-KM-YM" . T$=KK
2580. LPRINT "M1-KM-YM" . T$=KK
2590. LPRINT "M1-KM-YM" . T$=KK
2600. LPRINT "M1-KM-YM" . T$=KK
2610. LPRINT "M1-KM-YM" . T$=KK
2620. LPRINT "M1-KM-YM" . T$=KK
2630. LPRINT "M1-KM-YM" . T$=KK
2640. LPRINT "M1-KM-YM" . T$=KK
2650. LPRINT "M1-KM-YM" . T$=KK
2660. LPRINT "M1-KM-YM" . T$=KK
2670. LPRINT "M1-KM-YM" . T$=KK
2680. LPRINT "M1-KM-YM" . T$=KK
2690. LPRINT "M1-KM-YM" . T$=KK
2700. LPRINT "M1-KM-YM" . T$=KK
2710. LPRINT "M1-KM-YM" . T$=KK
2720. LPRINT "M1-KM-YM" . T$=KK
2730. LPRINT "M1-KM-YM" . T$=KK
2740. LPRINT "M1-KM-YM" . T$=KK
2750. LPRINT "M1-KM-YM" . T$=KK
2760. LPRINT "M1-KM-YM" . T$=KK
2770. LPRINT "M1-KM-YM" . T$=KK
2780. LPRINT "M1-KM-YM" . T$=KK
2790. LPRINT "M1-KM-YM" . T$=KK
2800. LPRINT "M1-KM-YM" . T$=KK
2810. LPRINT "M1-KM-YM" . T$=KK
2820. LPRINT "M1-KM-YM" . T$=KK
2830. LPRINT "M1-KM-YM" . T$=KK
2840. LPRINT "M1-KM-YM" . T$=KK
2850. LPRINT "M1-KM-YM" . T$=KK
2860. LPRINT "M1-KM-YM" . T$=KK
2870. LPRINT "M1-KM-YM" . T$=KK
2880. LPRINT "M1-KM-YM" . T$=KK
2890. LPRINT "M1-KM-YM" . T$=KK
2900. LPRINT "M1-KM-YM" . T$=KK
2910. LPRINT "M1-KM-YM" . T$=KK
2920. LPRINT "M1-KM-YM" . T$=KK
2930. LPRINT "M1-KM-YM" . T$=KK
2940. LPRINT "M1-KM-YM" . T$=KK
2950. LPRINT "M1-KM-YM" . T$=KK
2960. LPRINT "M1-KM-YM" . T$=KK
2970. LPRINT "M1-KM-YM" . T$=KK
2980. LPRINT "M1-KM-YM" . T$=KK
2990. LPRINT "M1-KM-YM" . T$=KK
3000. LPRINT "M1-KM-YM" . T$=KK
3010. LPRINT "M1-KM-YM" . T$=KK
3020. LPRINT "M1-KM-YM" . T$=KK
3030. LPRINT "M1-KM-YM" . T$=KK
3040. LPRINT "M1-KM-YM" . T$=KK
3050. LPRINT "M1-KM-YM" . T$=KK
3060. LPRINT "M1-KM-YM" . T$=KK
3070. LPRINT "M1-KM-YM" . T$=KK
3080. LPRINT "M1-KM-YM" . T$=KK
3090. LPRINT "M1-KM-YM" . T$=KK
3100. LPRINT "M1-KM-YM" . T$=KK
3110. LPRINT "M1-KM-YM" . T$=KK
3120. LPRINT "M1-KM-YM" . T$=KK
3130. LPRINT "M1-KM-YM" . T$=KK
3140. LPRINT "M1-KM-YM" . T$=KK
3150. LPRINT "M1-KM-YM" . T$=KK
3160. LPRINT "M1-KM-YM" . T$=KK
3170. LPRINT "M1-KM-YM" . T$=KK
3180. LPRINT "M1-KM-YM" . T$=KK
3190. LPRINT "M1-KM-YM" . T$=KK
3200. LPRINT "M1-KM-YM" . T$=KK
3210. LPRINT "M1-KM-YM" . T$=KK
3220. LPRINT "M1-KM-YM" . T$=KK
3230. LPRINT "M1-KM-YM" . T$=KK
3240. LPRINT "M1-KM-YM" . T$=KK
3250. LPRINT "M1-KM-YM" . T$=KK
3260. LPRINT "M1-KM-YM" . T$=KK
3270. LPRINT "M1-KM-YM" . T$=KK
3280. LPRINT "M1-KM-YM" . T$=KK
3290. LPRINT "M1-KM-YM" . T$=KK
3300. LPRINT "M1-KM-YM" . T$=KK
3310. LPRINT "M1-KM-YM" . T$=KK
3320. LPRINT "M1-KM-YM" . T$=KK
3330. LPRINT "M1-KM-YM" . T$=KK
3340. LPRINT "M1-KM-YM" . T$=KK
3350. LPRINT "M1-KM-YM" . T$=KK
3360. LPRINT "M1-KM-YM" . T$=KK
3370. LPRINT "M1-KM-YM" . T$=KK
3380. LPRINT "M1-KM-YM" . T$=KK
3390. LPRINT "M1-KM-YM" . T
```

qu'un PC-1500 avait été programmé spécialement pour nous dire tous les jours quelles allaient être les boîtes à surveiller car allant éclore.

De plus, une fois éclos, on donnait au PC-1500 ce qu'il était advenu des œufs de chaque boîte, on sortait alors un graphique avec le pourcentage de réussite de cette boîte.

Tous les graphiques ainsi obtenus étaient conservés sur un cahier pour Chantal, une jeune fille du village qui s'est occupée entre autre de la gestion informatique de l'écloserie.

## LE RÔLE DES PC SUR PLACE

Les dernières boîtes ont été entrées en mémoire en décembre et nous avons pu sortir des statistiques de réussite suivant le mois de ponte ou le type de couveuse.

Il avait été prévu de contrôler la température grâce à une interface analogique et de sondes QCN branchées sur la prise EAR d'un PC-2500. L'interface a été réalisée par Christophe POULIN, mais je n'ai jamais réussi à faire lire une température au PC-2500. Ce dernier s'est toujours montré réticent à faire ce que je désirai de lui en langage machine et finalement nous avons dû abandonner cette idée pour retenter en 87. Il nous restait donc un PC-2500 qui ne servait pas et il a été reconvertis à une étude de comportement.

Nous avons voulu déterminer les causes qui faisaient qu'une tortue s'arrête de creuser. Dans un premier temps il a fallu déterminer le comportement exact d'une tortue « normale » en train de creuser. Il faut savoir qu'une tortue creuse alternativement avec la patte arrière droite et gauche. Et à chaque fois qu'elle descend sa patte dans le trou en formation, elle racle le sable entre 1 et 9 fois pour le décrocher. Elle le remonte ensuite et l'évacue par un coup de pied en descendant l'autre patte.

Grâce à un programme adapté au PC-2500 nous avons déterminé pour les tortues étudiées les paramètres suivants :

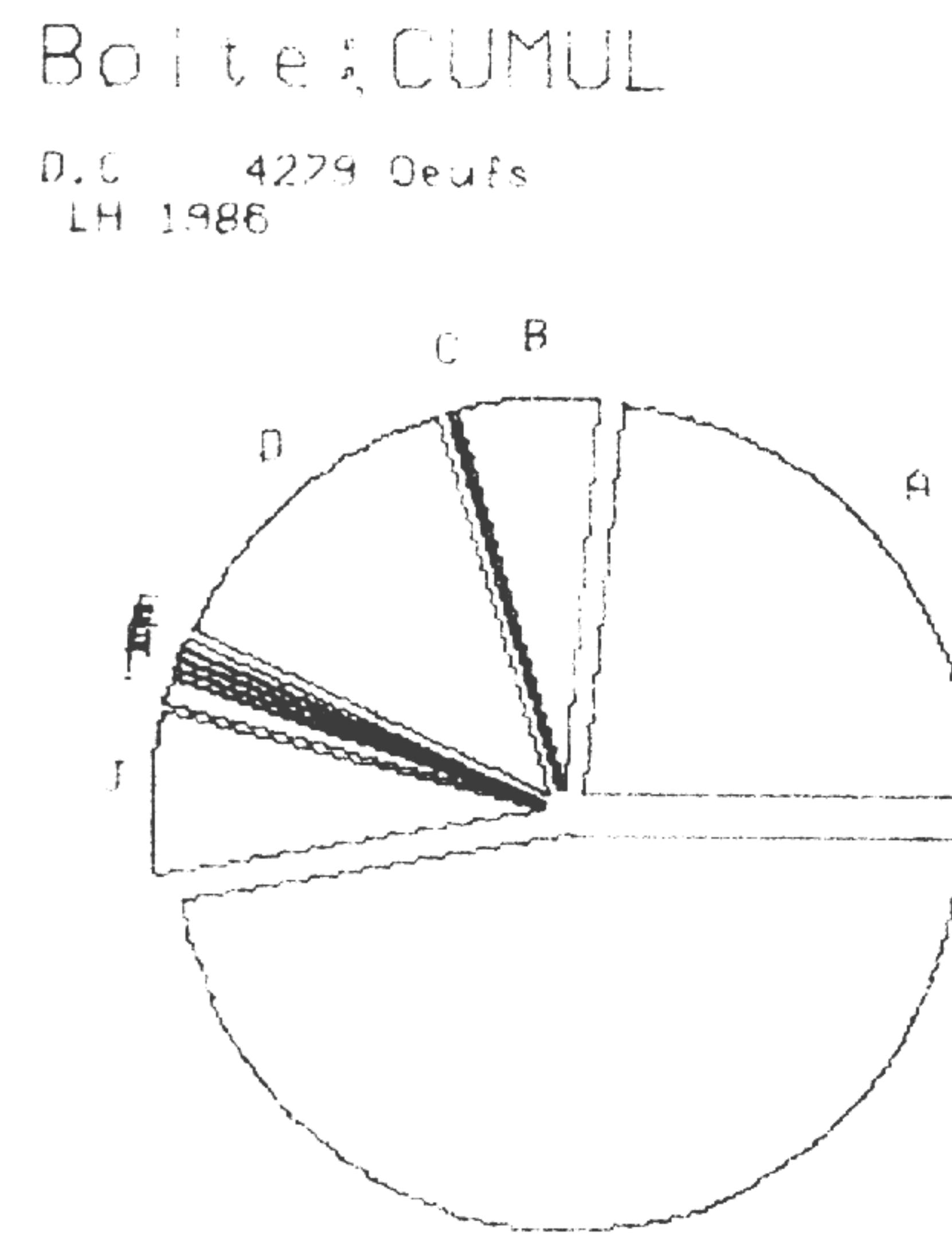
Fréquence de raclages des pattes droites et gauches en fonction du temps, Évolution du nombre de raclages de patte au fond du trou en fonction du temps,

Intervalle de temps entre deux raclages de patte au fur et à mesure du creusement.

Malheureusement nous n'avons pas eu le temps de faire assez d'expériences pour en tirer des conclusions, mais nous commençons à mieux cerner le problème.

## PRÉPARATION DE KAWANA 87

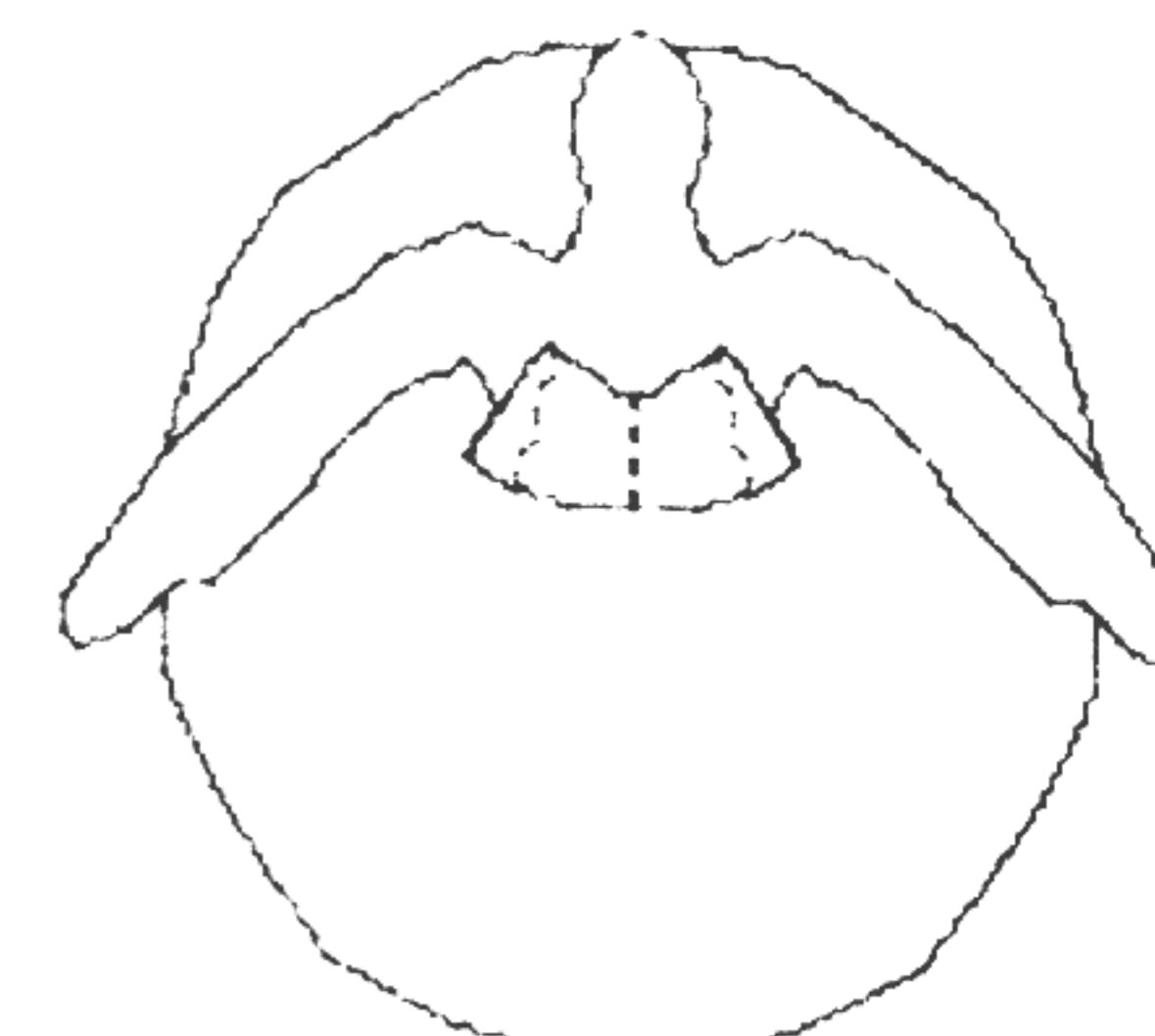
L'année prochaine marquera encore une nouvelle étape dans l'informatisation de cette mission puisque l'on devrait arriver en 87 avec 6 PC.



Expe.	10	K
Info.	4	
A. Pour.	1004	23,54 %
B. MoEcl.	251	5,88 %
C. Aben.	9	0,21 %
D. SaDev.	580	13,59 %
E. 10-20.	21	0,49 %
F. 20-30.	20	0,46 %
G. 30-40.	16	0,32 %
H. 40-50.	12	0,39 %
I. 50-60.	46	1,02 %
J. 60-70.	282	6,61 %
K. Eclos.	2019	47,33 %
Reus.	2552	59,83 %

Devenir des œufs de tortues luths en couveuses non thermorégulées - 1986.

A.G.E.O.T.M.G.



KAWANA 87

Logo et sigle de l'Association pour la gestion des écloseries d'œufs de tortues marines de Guyane - Campagne 1987.

Un PC-1500 continuera à gérer l'écloserie avec une petite innovation, nous envisageons un tableau de LED et quand une boîte devra être surveillée, la LED correspondante s'allumera.

La plus grosse innovation proviendra d'un PC-1600 qui captera sans arrêt pendant six mois la température d'une pièce entièrement thermorégulée.

Nous allons essayer de brancher huit sondes sur la seule prise analogique du 1600. Christophe POULIN sera encore mis à contribution !

Il y aura toujours un PC qui nous aidera à continuer le travail sur le comporte-

ment de creusement des tortues. Si nous en avons les moyens un PC-1600 serait l'idéal...

Trois PC-1500 serviront à noter les tortues rencontrées sur la plage. Pour chaque tortue les paramètres suivants seront pris :

Date et heure grâce à TIME, Numéro de la tortue s'il y a lieu. Ce qu'elle est en train de faire pour retrouver par extrapolation son heure d'arrivée.

Pour l'instant tout était noté sur des feuilles d'où beaucoup de manipulations pour en tirer des résultats corrects.

Avec cette méthode, au retour des patrouilles nous brancherons les

PC-1500 sur une EPF-1500-16 et les données seront sauvées sur EPROM qui seront reprises à Paris sur un IBM PC. En effet le climat nous oblige à prendre quelques précautions puisque l'humidité et la chaleur rend les cassettes pratiquement inutilisables. Nous ne savons pas ce qu'il en est des disquettes. De plus la CE-150 est tombée en panne vers la fin du séjour, la carte intérieure était couverte de moisissures ! Mis à part ceci le matériel a bien tenu ce qui est déjà en soit une bonne performance.

Jacques FRETEY  
Marc GIRONDOT

## NOUVEAUX FORMATS

**Cette petite routine adaptable à tous les PC satisfera tous les utilisateurs travaillant dans un domaine scientifique.**

Le résultat des calculs sera présenté sous une forme avec puissance de 10 et un chiffre arrondi après la virgule, ainsi que sous le format habituel pour repère.

ARUN : Lance le programme immédiatement après la mise sous tension. Cette instruction est à supprimer si on ne travaille pas sur PC-1500. Pour le mettre en route il suffira de taper RUN ENTER.

INPUT ">",A : Affiche > et attend un nombre ou un calcul dont le résultat sera stocké dans A. La virgule a pour effet de faire disparaître le > lorsque la première touche est appuyée.

USING "#.#": Formate l'affichage sous la forme SIGNE UNITE.DIZIEME.  
INT LOG ABS A : Calcule la puissance de 10 du nombre A.

A/... : Met le nombre sous le même format que l'affichage.  
+SGN A\*.05 : Permet de faire l'arrondi sur le dizième par ajout de 5 centièmes.

"E": : Affiche E pour signifier que le nombre suivant est la puissance de 10.

USING : Annule le format défini auparavant.  
INT LOG ABS A : Affiche la puissance de 10 de A.

"=>": : Affiche =>, on pourrait mettre toutes les séries de deux signes que l'on veut, ce n'est qu'une décoration.

A : Affiche la valeur non formatée.  
GOTO 1 : Retourne au début pour attendre un nouveau calcul.

Remarquez qu'il n'y a aucune définition de WAIT, il faudra donc appuyer sur ENTER après un affichage pour retourner au message d'invite.

Vous pouvez récupérer la valeur du calcul précédent qui se trouve bien évidemment dans A.

Marc GIRONDOT

```

1 ARUN : INPUT ">",A:PRINT USING "#.#
 .#":(A/10^INT LOG ABS A)+SGN A*.05;"E":0
 SING :INT LOG ABS A;"=>";A:GOTO 1

```

# KARATE K

**Extrait d'une revue japonaise PIO, ce jeu graphique a pour thème la pratique des arts martiaux. Attaqué de tous côtés, vous allez devoir vous défendre selon un rituel établi.**

Faites avant toute chose un NEW ou ALL RESET. En effet, ce programme doit être le premier en mémoire. Tapez le Basic et vérifiez-le très attentivement, surtout la première ligne 5. Remarquons de suite que D\$ contient 3 espaces. Sauvez le programme pour plus de sûreté, puis tapez POKE &2011,254,254 ENTER.

Maintenant la ligne 5 doit se trouver modifiée, et D\$ contiendra un espace et deux pavés noirs. Le jeu démarre par RUN ou DEF A, appuyez sur ENTER pour effacer l'écran de présentation et le numéro de l'étape s'affiche (STAGE). Les touches 4 et 6 permettent de se déplacer à gauche ou à droite. Les touches Z ou C agissent sur les pieds pour donner les coups.

Tout de suite vous êtes assailli des deux côtés par des combattants qu'il faut tuer pour survivre. Vous n'avez hélas pas le choix, mais votre technique de pieds est souvent mortelle. A chaque fois qu'un ennemi meurt, un autre le remplace : il y en a beaucoup. Votre énergie est matérialisée par les pavés noirs, à gauche sur l'écran.

A chaque fois que vous êtes touchés, les pavés diminuent et votre résistance aussi. Lorsque tous vos ennemis sont à terre, le Grand Maître vient pour le combat final armé d'un bâton et si vous triomphez de lui, vous passez à l'étape suivante, encore plus dure. Je vous laisse le soin de découvrir la suite du jeu.

Bon Karaté ! !

Jean-François VIGNAUD

```

5;"A" CLEAR :WAIT 0:D$=":":PRINT ""
10:J=&6054:Z=&6040:N=&6068:H=10:A=&600F:W=8
:Y=10:UB=0:B=1:CALL &5A2
12:POKE &6014,127,120,2
0,34,65,120,127,9,9,
127,127,121,25,41,70
,120,127,9,9,127
13:POKE &6063,3,1,127,1
21,3:POKE &605E,99,7
3,73,121,127:POKE &6
054,65,34,20,120,127
14:POKE &6000,64,96,64,
64,32:POKE &6005,16,

```

```

16,28,43,83:POKE &60
80,72,107,31,107,72
15:POKE &6040,64,96,64,
64,32:POKE &604A,72,
107,31,107,72
16:POKE &6045,16,16,28,
43,83:CALL &1421
17:FOR L=&6000 TO &6027
:POKE L,0:NEXT L:FOR
L=&6067 TO &6040
STEP -1:POKE L,0:
NEXT L
18:WAIT 100:PRINT "
STAGE:";B:PRINT "
GAME START!":WAIT 0
19:PRINT D$:GOTO 40
20:POKE J-1,16,16,16,28
,43,83:FOR L=0 TO 15
:NEXT L:POKE J-1,0:
GOTO 210
21:CALL &5A2:POKE J,72,
107,31,107,72:K$=
INKEY$:
23:IF D$=20+0 THEN 350
45:IF J>N POKE N,0,0,0
,0,0:N=&6068:H=H-1:A
=A-1:POKE A,0
48:IF J<=Z POKE Z,0,0,0
,0,0:Z=&6040:H=H-1:A
=A-1:POKE A,0
50:IF K$="Z" THEN 260
55:IF K$="C" THEN 30
60:IF K$="6" LET J=J-5:
POKE J+5,0,0,0,0,0:
GOTO 95
70:IF K$="4" LET J=J+5:
POKE J-5,0,0,0,0,0
95:IF H=0 THEN 290
100:IF H>&6045 LET N=&60
63:V=0
110:IF JK>&6040 LET J=&60
40:GOTO 130
120:IF J>&6063 LET J=&60
63
130:C=RND 2:IF C=2 LET V
=1:GOTO 150
140:IF C=1 LET V=2
150:IF V=1 THEN 180
160:IF V=2 THEN 270
170:GOTO 40
180:POKE Z,0,0,0,0,0,8,
7,63,87,8
190:Z=Z+5:IF Z>&6068 LET
Z=&6040
200:GOTO 40
210:IF Z=J-5 AND K$="C"
AND JK>&6045 CALL &5
A0:CALL &5A2:POKE Z,
0,0,0,0,0:POKE Z-5,6
4,112,64,64,32
220:IF Z=J-5 AND K$="C"
AND JK>&6045 FOR L=0
TO 30:NEXT L:POKE Z-
5,0,0,0,0,0:Z=&6040:
V=0:S=S+10:0=0+1
225:GOTO 40
230:IF N=J+5 AND K$="Z"
CALL &5A0:CALL &5A2:

```

```

POKE N,0,0,0,0,0
POKE N+5,32,64,64,11
2,64:FOR L=0 TO 30:
NEXT L
240:IF N=J+5 AND K$="Z"
POKE N+5,0,0,0,0,0:V
=0:N=&6068:S=S+10:0=
0+1:GOTO 40
250:GOTO 40
260:POKE J,83,43,28,16,1
6,16:FOR L=0 TO 15:
NEXT L:POKE J+5,0:
GOTO 230
270:N=N-5:POKE N,8,87,63
,87,8,0,0,0,0,0
280:GOTO 230
290:CALL &5A0:CALL &5A2:
POKE &6054,64,96,64,
96,16:FOR L=0 TO 4:
POKE &6059,32,87,63,
87,32:FOR F=0 TO 5:
NEXT F
300:POKE &604F,32,87,63,
87,32:FOR F=0 TO 3:
NEXT F:POKE &6059,8,
87,63,87,8:FOR F=0
TO 5:NEXT F
305:POKE &604F,8,87,63,8
7,8:FOR F=0 TO 3:
NEXT F:NEXT L
306:FOR L=&6000 TO &6027
:POKE L,127:NEXT L:
FOR L=&6067 TO &6040
STEP -1:POKE L,127:
NEXT L:GOSUB 890
310:WAIT :PRINT "YOUR SC
ORE:";B:CALL &1425
320:WAIT 0:K$="":PRINT "
REPLAY?(Y/N)":
CALL &1425:K$=INKEY$:
:PRINT "":IF K$="Y"
THEN 5
330:IF K$="N" END
340:GOTO 320
350:W=8:I=&6040:CALL &5A
0:CALL &5A2:POKE Z,0
,0,0,0,0:POKE N,0,0,0
,0,0:Y=Y+E
355:POKE J,72,107,31,107
,72:POKE I,W,87,61,8
7,W:K$=INKEY$ :IF K$=
"C" THEN 370
360:GOTO 380
370:POKE J-1,16,16,16,28
,43,83:FOR L=0 TO 15
:NEXT L:POKE J-1,0:
GOTO 550
380:IF K$="Z" THEN 390
385:GOTO 400
390:POKE J,83,43,28,16,1
6,16:FOR L=0 TO 15:
NEXT L:POKE J+5,0:
GOTO 500
400:IF K$="4" LET J=J+5:
POKE J-5,0,0,0,0,0:
GOTO 420
410:IF K$="6" LET J=J-5:
POKE J+5,0,0,0,0,0
420:IF JK>&6040 LET J=&60
40:GOTO 440
430:IF JK>&6063 LET J=&60
63
440:C=RND 4:IF C=1 LET I
=I+5:POKE I-5,0,0,0,0
,0,0:GOTO 460
445:IF C=2 LET I=I-5:
POKE I+5,0,0,0,0,0:
GOTO 460
450:IF C=3 POKE I+4,8,8
,8,8,8,8:FOR L=0 TO 2
:NEXT L:POKE I+4,W,
0,0,0,0,0:GOTO 570
455:IF C=4 AND JK>&6040
POKE I-5,8,8,8,8,8

```

```

:FOR L=0 TO 20:NEXT
L:POKE I-5,0,0,0,0,0
,W:GOTO 575
460:IF I<&6040 LET I=&60
40:GOTO 355
470:IF I>&6063 LET I=&60
63
490:GOTO 355
500:IF I=J+5 AND K$="Z"
LET Y=Y-1:S=S+10:
CALL &5A0:CALL &5A2
505:IF Y<10 LET W=32
510:IF Y=0 THEN 300
520:GOTO 440
530:IF I=J-5 AND K$="C"
LET Y=Y-1:S=S+10:
CALL &5A0:CALL &5A2
555:IF Y<10 LET W=32
560:IF Y=0 THEN 300
565:GOTO 440
570:IF J=I+5 LET A=A-1:H
=H-1:BEEP 1:CALL &5A
2:POKE A,0
573:GOTO 580
575:IF J=I-5 LET A=A-1:H
=H-1:BEEP 1:CALL &5A
2:POKE A,0
580:IF H=0 THEN 290
590:GOTO 355
600:CALL &5A0:CALL &5A2:
POKE I,0,0,0,0,0
POKE J+5,64,112,64,6
4,32:FOR L=0 TO 4:
FOR F=0 TO 2
810:POKE J,80,107,31,107
,80:NEXT F:FOR F=0
TO 2:POKE J,68,107,3
1,107,68:NEXT F:NEXT
L:IF E>=20+P THEN 85
0
820:WAIT 100:PRINT "
CLEAR !":PRINT "BO
NUS:";B:30*H:S=S+30*H:
PRINT "YOUR SCORE:";
S
830:J=&6054:Z=&6040:A=&6
00F:Y=10:E=E+5:0=0:B
=B+1:PRINT " STAG
E:";B:0=0+S:N=&6068
840:H=10:WAIT 0:PRINT ""
:GOTO 19
850:WAIT 100:PRINT "***
GIVE UP! ***":P=P+5:
PRINT "SPECIAL ";E*3
0;"PTS":S=S+E*30:E=0
860:GOTO 820
880:DATA 0,0,0,0,0,8,87
,63,87,8,8,87,61,87,8
,62,121,65,73,57,120
,127,9,9,127,127,122
885:DATA 4,2,127,127,121
,73,73,99,0,0,0,0,0
,0,0,0,0,0,62,121,6
5,65,62,15,60,96,60
886:DATA 15,127,121,73,7
3,99,127,121,25,41,7
0,8,87,61,87,8,8,87
,63,87,8,0,0,0,0,0
890:CALL &5A2:RESTORE :
FOR L=&6000 TO &6028
:READ DD:POKE L,DD:
NEXT L
900:FOR L=&6067 TO &6040
STEP -1:READ DD:POKE
L,DD:NEXT L
920:POKE &6000,0
930:CALL &1425:RETURN

```

# VOCABLE

**Ce programme a été conçu sur un PC-1401 SHARP adaptable sur les autres 14xx et 12xx.**

**De nos jours, on se trouve facilement décontenancé face à un étranger, faute de vocabulaire. Comment l'améliorer et le cultiver, tel est le but que s'est donné VOCABLE, conçu et réalisé par Julien Lever.**

#### Principe d'utilisation du programme:

##### Mise en marche :

Pressez les touches DEF + F, l'ordinateur affichera : « NEW PROG. (Y/N) ? ».

Il vous demande ainsi par Y ou N si vous désirez réintroduire une nouvelle liste de mots et effacer les anciens, s'il en existait, ou si vous préférez poursuivre une liste de mots qui avait déjà été introduite dans la mémoire :

- 1<sup>er</sup> choix : Y
- 2<sup>e</sup> choix : N

##### Remarque :

L'ordinateur accepte un maximum de 38 mots dans sa mémoire. Si pour une raison ou une autre vous êtes interrompu dans votre programmation, vous pouvez, après l'avoir éteint, reprendre votre liste de mots et la compléter : pour cela, taper à la question « NEW PROG.(Y/N) ? » N. L'ordinateur affichera un nombre, puis « COMMENCEMENT ? ». Il vous suffira d'entrer le nombre préalablement affiché puis de continuer votre liste normalement.

Après donc avoir tapé Y, « NBER OF WORDS ? » s'inscrit ; vous devez alors rentrer le nombre de mots que vous désirez enregistrer. Puis l'ordinateur, par les questions « FOREIGN WORD ? » et « TRANSLATION ? » vous demande, respectivement, d'introduire le mot de la langue étrangère et sa traduction. Si le nombre de mots introduit est inférieur à 38, pour continuer le programme, tapez ENTER lors de la question : « FOREIGN WORD ? » ; vous obtiendrez ainsi la suite du programme.

Cependant, si le nombre de mots entré est égal à 38, ou au nombre précédemment introduit lors de la question « NBER OF WORDS ? », le programme affiche alors : « LEVEL(1,2,3)? ». Le choix du numéro permet de sélectionner le temps qui sera accordé pour chaque réponse lors de l'interrogation, à savoir (1=30 s, 2=20 s, 3=13 s). Puis avec la question « TIME(1,2,3)? », procédez de même,

sachant qu'ici c'est le temps de l'interrogation entière que vous sélectionnez, c'est-à-dire 1 = l'ordinateur vous interroge 19 fois, 2 = 38 fois, 3 = 76 fois.

##### L'interrogation commence :

La machine vous donne un mot tiré au hasard parmi les traductions données, puis il affiche un écran vide : à vous de taper le mot étranger, en insistant bien sur chaque touche de façon prompte. En cas de faute de frappe, appuyez sur « O », l'écran effacera tout le mot. Une fois que votre mot est orthographié, appuyez sur « = ». L'ordinateur compare votre réponse avec la bonne traduction ; son verdict tombe rapidement

- par l'affichage « GOOD !!! », l'interrogation continue, un autre mot est demandé.
- par l'affichage de « WRONG !!! », il vous donne le mot corrigé et repose ce même mot afin qu'il soit bien mémorisé.

Le **score** apparaît toutes les cinq bonnes ou mauvaises réponses avec un changement automatique de force (level (1,2,3)).

Pour 5 mauvaises réponses, vous régressez d'un niveau et inversement. Lorsque le test se termine, l'ordinateur vous le fait savoir et donne votre score, le nombre de bons et de mauvais mots et, en prime, son appréciation.

Ensuite, il vous redemande si vous voulez rejouer. Si oui, tapez Y, si non tapez N.

Un **avantage** de ce programme : après avoir éteint la machine, les mots précédemment programmés demeurent dans la mémoire. Pour vous faire réinterroger, tapez DEF G, l'interrogation s'en suivra.

##### ATTENTION :

Une erreur surviendra si vous n'avez pas déjà tapé DEF F et introduit une liste de mots. Tous les mots enregistrés restent jusqu'à renouvellement du programme en ne tapant pas Y à « NEW-PROG ? ».

D'ailleurs, pour éviter toute erreur fâcheuse, le programme vous signalera que la mémoire est déjà remplie en affichant : « SAVED MEMORY ».

##### Autre avantage :

A l'aide des instructions d'enregistrement de variables sur bande par INPUT # et PRINT #, vous pouvez vous constituer une banque de données de mots sur cassette.

Voilà, bonne chance !

Progrès immédiats assurés.

Julien LEVER

```

400:Z=2+1
404:REM IF W=J THEN GOTO
      900
410:IF G$(D)=RR$ BEEP 3:
      PAUSE "GOOD!!!";Y=Y+
      1:M=M+1:P=P+1
420:IF G$(D)<RR$ PAUSE
      "WRONG!!!";Y=Y-1:R=R+
      1:D=D+1:GOTO 450
421:IF W=J GOTO 900
422:IF M=5 GOSUB 650
448:GOTO 290
450:PAUSE "THE GOOD WORD
      IS:"
455:WAIT 200:PRINT G$(D)
456:IF R=5 GOSUB 735
465:PAUSE "TRANSLATE:";J
      =J+1
466:WAIT 200:PRINT H$(D)
      :GOTO 330
467:IF W=J GOTO 900
470:INPUT "COMMENCEMENT?
      ";X
480:GOTO 140
550:W=19
560:RETURN
570:W=38
580:RETURN
590:W=76
600:RETURN
605:GOTO 290
650:F=F+1
700:ON F GOSUB 800,820,6
      40
720:PAUSE USING "####";Y;
      X#;Z:GOTO 290
725:RETURN
730:END
735:PAUSE USING "####";Y;
      X#;Z
737:F=F+1
740:ON F GOSUB 800,820,6
      40
760:RETURN
800:D=100:R=0:M=0:PAUSE
      "LEVEL 1"
810:RETURN
820:D=65:R=0:M=0:PAUSE
      "LEVEL 2"
830:RETURN
840:D=45:R=0:M=0:PAUSE
      "LEVEL 3"
850:RETURN
900:WAIT 110:PRINT "END
      OF YOUR TEST"
910:PRINT "YOUR SCORE:";Y
920:IF Z=Y:XY WAIT 30:
      PRINT "FINENESS"
930:IF Z-Y:XY WAIT 30:
      PRINT "PRETTY GOOD."
940:IF Z-Y:XY WAIT 30:
      PRINT "INCAPABLE"
945:WAIT 200:PRINT "NB G
      OD WORDS";":PRINT
      USING "####";P#;X#;J
947:WAIT 200:PRINT "GO,B
      AD WORDS";":D
950:WAIT 50:PRINT "REPLA
      Y (Y/N)";Z#=INKEY#
960:IF Z$="Y" OR Z$="N"
      GOTO 690
970:GOTO 950
980:IF Z$="Y" GOTO 195
990:END

```

# CONVERSION DE TEMPÉRATURE

## MODE D'EMPLOI

Ce programme permet de faire différentes conversions entre plusieurs échelles de températures. Il occupe 1470 octets de mémoire. Taper R/JN pour obtenir le menu. Lorsque les échelles de températures désirées s'affichent, taper "O" (oui). Taper sur ENTER à chaque fois pour faire avancer le menu. Puis taper l'inconnue ("C" pour celsius, "K" pour kelvin, "F" pour fahrenheit et "R" pour réaumur. Entrer la valeur de la température connue et vous obtenez la conversion désirée.

## FORMULES DES CONVERSIONS

- Conversion entre celsius (C) et kelvin (K):  
 $C = K - 273$   
 $K = C + 273$

- Conversion entre celsius (C) et fahrenheit (F):  
 $C = (F - 32) \times (10/18)$   
 $F = (C \times 1.8) + 32$

- Conversion entre celsius (C) et réaumur (R):  
 $C = 1.25 \times R$   
 $R = 0.8 \times C$

Conversion entre kelvin (K) et fahrenheit (F):  
 $K = ((F - 32) \times (10/18)) + 273$   
 $F = ((K - 273) \times 1.8) + 32$

- Conversion entre kelvin (K) et réaumur (R):  
 $K = (1.25 \times R) + 273$   
 $R = (K - 273) \times 0.8$

- Conversion entre fahrenheit (F) et réaumur (R):  
 $F = (R \times 2.25) + 32$   
 $R = (F - 32) \times (8/18)$

## EQUIVALENCE DES VARIABLES

"C" valeur en celsius, "F" valeur en fahrenheit, "K" valeur en kelvin, "R" valeur en réaumur, "F\$" inconnue ? (K ou R), "G\$" inconnue ? (F ou R), "I\$" <TEMPERATURE>, "J\$" <TURE EN>, "M\$" inconnue ? (C ou R), "N\$" inconnue ? (K ou F), "P\$" inconnue ? (C ou K), "Q\$" inconnue ? (C ou F), "S\$" <UNE AUTRE (O/N)>, "U\$" <CELSIUS>, "V\$" <KELVIN>, "W\$" <FAHRENH>, "X\$" <EIT>, "Y\$" <REAUMUR>, "Z\$" <>.

Edmond LEMAITRE

```

1 : "LEMAITRE EDMOND"
5 . WAIT 83 . CLEAR . PRINT " CONVERSION DE ". PRINT
" TEMPERATURE ". WAIT
1 0 : INPUT " CELSIUS ", A$
1 5 : INPUT " KELVIN ", B$
2 0 : INPUT " FAHRENHEIT ", C$
2 5 : INPUT " REAUMUR ", D$
3 0 : I$ = " TEMPERA ", J$ = " TURE EN "
3 5 : U$ = " CELSIUS ", V$ = " KELVIN ", W$ = " FAHRENH ", X$ =
" EIT ", Y$ = " REAUMUR ", Z$ = " "
4 0 : IF (A$ = " O ") AND (B$ = " O ") AND (C$ = " ") AND (D$ = " ") GO
TO 100
4 5 : IF (A$ = " O ") AND (B$ = " ") AND (C$ = " O ") AND (D$ = " ") GO
TO 200
5 0 : IF (A$ = " O ") AND (B$ = " ") AND (C$ = " ") AND (D$ = " O ") GO
TO 300
5 5 : IF (A$ = " ") AND (B$ = " O ") AND (C$ = " O ") AND (D$ = " ") GO
TO 400
6 0 : IF (A$ = " ") AND (B$ = " O ") AND (C$ = " ") AND (D$ = " O ") GO
TO 500
6 5 : IF (A$ = " ") AND (B$ = " ") AND (C$ = " O ") AND (D$ = " O ") GO
TO 600
7 0 : GOTO 5
1 0 0 : BEEP 1 . INPUT " INCONNUE ? "; P$
1 1 0 : IF P$ = " C " INPUT " K = "; K . C = K - 273 . BEEP 3 . WAIT 7
2 : PRINT I$ + J$ . WAIT . PRINT U$ + Z$ . C : GOTO 1000
1 2 0 : IF P$ = " K " INPUT " C = "; C . K = C + 273 . BEEP 3 . WAIT 72
: PRINT I$ + J$ . WAIT . PRINT V$ . K : GOTO 1000
1 3 0 : GOTO 5
2 0 0 : BEEP 1 . INPUT " INCONNUE ? "; Q$
2 1 0 : IF Q$ = " C " INPUT " F = "; F . C = (F - 32) * (10/18) . BEEP 3 .
WAIT 72 . PRINT I$ + J$ . WAIT . PRINT U$ + Z$ . C : GOTO
1000
2 2 0 : IF Q$ = " F " INPUT " C = "; C . F = (10/18) * C + 32 . BEEP 3 . WA
IT 72 . PRINT I$ + J$ . WAIT . PRINT W$ + X$ . F : GOTO
1000
2 3 0 : GOTO 5
3 0 0 : BEEP 1 . INPUT " INCONNUE ? "; M$
3 1 0 : IF M$ = " C " INPUT " R = "; R . C = 1.25 * R . BEEP 3 . WAIT 72
: PRINT I$ + J$ . WAIT . PRINT U$ + Z$ . C : GOTO 1000
3 2 0 : IF M$ = " R " INPUT " C = "; C . R = 0.8 * C . BEEP 3 . WAIT 72
: PRINT I$ + J$ . WAIT . PRINT Y$ + Z$ . R : GOTO 1000
3 3 0 : GOTO 5
4 0 0 : BEEP 1 . INPUT " INCONNUE ? "; N$
4 1 0 : IF N$ = " K " INPUT " F = "; F . K = ((F - 32) * (10/18)) + 273
. BEEP 3 . WAIT 72 . PRINT I$ + J$ . WAIT . PRINT V$ . K :
GOTO 1000
4 2 0 : IF N$ = " F " INPUT " K = "; K . F = ((K - 273) * 1.8) + 32 . BEEP
3 . WAIT 72 . PRINT I$ + J$ . WAIT . PRINT W$ + X$ . F : GOTO
1000
4 3 0 : GOTO 5
5 0 0 : BEEP 1 . INPUT " INCONNUE ? "; F$
5 1 0 : IF F$ = " K " INPUT " R = "; R . K = (1.25 * R) + 273 . BEEP 3 .
WAIT 72 . PRINT I$ + J$ . WAIT . PRINT V$ . K : GOTO 1000
5 2 0 : IF F$ = " R " INPUT " K = "; K . R = (K - 273) * 0.8 . BEEP 3 .
WAIT 72 . PRINT I$ + J$ . WAIT . PRINT Y$ + Z$ . R : GOTO
1000
5 3 0 : GOTO 5
6 0 0 : BEEP 1 . INPUT " INCONNUE ? "; G$
6 1 0 : IF G$ = " F " INPUT " R = "; R . F = (R * 2.25) + 32 . BEEP 3 .
WAIT 72 . PRINT I$ + J$ . WAIT . PRINT W$ + X$ . F : GOTO
1000
6 2 0 : IF G$ = " R " INPUT " F = "; F . R = (F - 32) * (8/18) . BEEP 3 .
WAIT 72 . PRINT I$ + J$ . WAIT . PRINT Y$ + Z$ . R :
GOTO 1000
6 3 0 : GOTO 5
1 0 0 0 : INPUT " UNE AUTRE (O/N) "; S$
1 0 1 0 : IF S$ = " O " GOTO 5
1 0 5 0 : END

```

# WATER PANIC

Encore un jeu Japonais qui est à mon sens franchement GÉNIAL (Eh ! les Français, on dort ?). Sur un écran aussi petit, c'est un vrai tour de force de réaliser un jeu d'arcade aussi

intéressant. J'en veux pour preuve le succès qu'il a rencontré pendant certains TD de maths ennuyeux à la Fac.

Que les possesseurs de PC 1401/02 se rassurent, une version leur sera proposée dans un prochain numéro (cela nécessite un peu de travail, alors patience !). Suivant la carte RAM disponible dans votre PC 1450, faites All Reset, puis :

- pour une carte 4 Ko, faire POKE &5F01,0,&55 ENTER

- pour une carte 8 Ko, faire POKE &5F01,0,&45 ENTER
- pour une carte 16 Ko, faire POKE &5F01,0,&25 ENTER puis NEW ENTER.

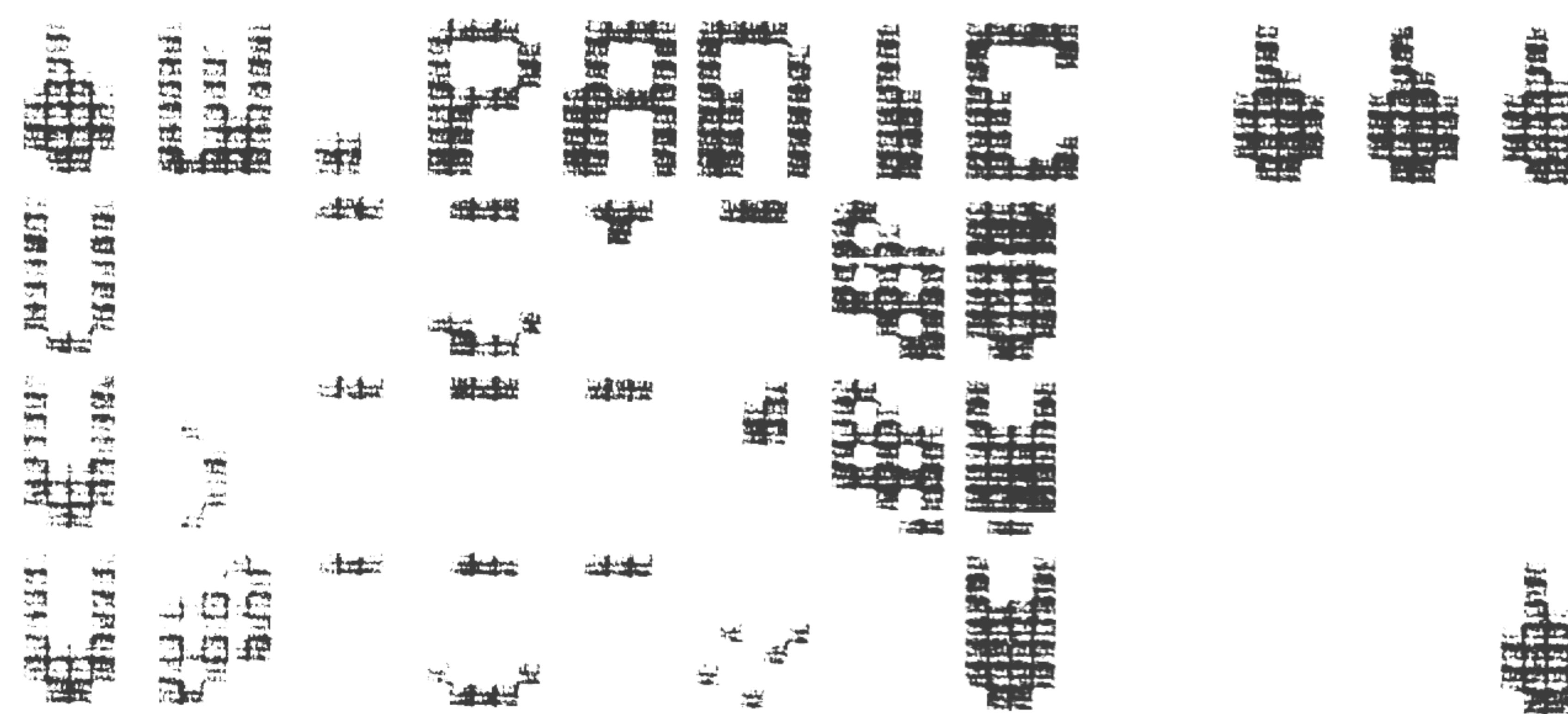
Considérons le cas d'une carte 4 ko de base (il reste 1838 octets). Le jeu WATER PANIC est entièrement en langage machine et va de l'adresse &2040 à &24FF. Par contre, pour le mettre en mémoire, on passe par un chargeur Basic. En fait, à cause du peu de mémoire disponible, on passera par quatre programmes Basic. On les tapera, sauvera et exécutera tour à tour (par RUN). Le LM se construit petit à petit et ne s'efface pas avec un NEW. Une fois le dernier chargeur exécuté, on peut sauver le LM par : CSAVE M «W.PANIC»;&2040,&24FF ENTER.

Les possesseurs de carte 8 ou 16 ko peuvent modifier les chargeurs pour faire une ou deux passes seulement. Les programmes contrôlent par checksum la validité des codes entrés, et affichent 'CS ERROR IN No' si une erreur de frappe est détectée. Le bon chargement est mentionné par l'affichage de 'Completed'. Respectez l'ordre d'introduction des chargeurs numérotés de 1 à 4. Maintenant que le jeu est en mémoire, il s'exécute par CALL &23F0. Et les touches de déplacement gauche/droite sont 4 et 6. Sur la figure publiée quelque part pas loin, vous pouvez voir des exemples d'écran possible. Après un écran de présentation, le jeu débute.

Vous dirigez un récipient qui peut contenir au maximum 3 gouttes. Il faut recueillir ces gouttes qui tombent au hasard parmi 4 positions. Vous avez trois vies, symbolisées par trois grosses gouttes d'eau à droite de l'écran, juste avant le score. Mais là où ça devient difficile, c'est qu'il faut vider le bac avant qu'il ne déborde, à gauche ou à droite. Une goutte en trop (3 maxi) ou une goutte à côté, et vous perdez une vie.

Bien sûr, la machine prend un malin plaisir à déplacer un obstacle empêchant tantôt à gauche, tantôt à droite de vider son bac dans les grands récipients (qui se vident seuls). Tous les 100 points, vous avez une vie en plus par un bonus, mais le jeu s'accélère notablement. Le High Score est à 420, alors à vos PCs. Vous verrez, on comprend rapidement, et c'est très rageant car ça va vite.

Le Club



#### Chargeur N.1

```

100:REM +00 +01 +02 +03
      +04 +05 +06 +07 +08
      +09 +0A +0B +0C +0D
      +0E +0F :0S
110:DATA 000,001,001,001
      ,000,000,000,000,000
      ,001,003,001,000,000
      ,000,000,008
120:DATA 000,003,003,000
      ,000,000,000,000,000
      ,006,007,000,000,000
      ,000,000,013
130:DATA 000,000,001,000
      ,000,000,000,000,000
      ,031,031,000,000,000
      ,000,000,078
140:DATA &3F,&40,&40,&3F
      ,000,000,000,000,03F
      ,060,060,&3F,000,000
      ,000,000,030
150:DATA &3F,&70,&70,&3F
      ,000,000,000,000,03F
      ,078,&78,&3F,000,000
      ,000,000,000
160:DATA &3F,&70,&70,&3F
      ,000,000,000,000,03F
      ,07E,&7E,&3F,000,000
      ,000,000,&F0
170:DATA &3F,&7F,&7F,&3F
      ,000,000,000,000,000
      ,000,000,000,000,000
      ,000,000,&70
180:DATA &70,&54,&3E,&15
      ,&1F,&00,000,000,000
      ,044,&38,&00,000,000
      ,000,000,&BE
190:DATA &20,&40,&40,&40
      ,020,000,000,000,020
      ,060,040,&40,&20,000
      ,000,000,020
200:DATA &20,&60,&60,&40
      ,020,000,000,000,020
      ,060,&60,&60,&20,000
      ,000,000,&A0
210:DATA &20,&08,&40,&10
      ,008,000,000,000,020
      ,068,&70,&60,&28,000
      ,000,000,000
220:DATA &00,&38,&70,&7F
      ,038,000,000,000,020
      ,026,&37,&34,&20,&26
      ,030,036,&E0
230:DATA &84,&03,&20,&01
      ,05A,&5A,&5A,&74,&40
      ,013,002,00A,005,037
      ,000,04D,&E2
240:DATA &78,&21,&00,000
      ,005,024,&26,&41,&29
      ,004,&37,&00,000,000
      ,000,04D,&DA
250:DATA &03,&24,&02,&03F
      ,0E2,&95,&A6,&59,&74
      ,006,&78,&21,&10,&8A4
      ,059,&74,&72

```

```

260:DATA 000,&78,&21,&10
      ,002,&20,&34,&30,&59
      ,078,&21,&10,&5B,&42
      ,067,&24,&66
270:DATA 029,&0B,&A7,&59
      ,074,&0E,&78,&23,&E2
      ,0A9,&59,&74,&06,&79
      ,021,&10,&59
280:DATA &AA,&59,&74,&10
      ,078,&21,&00,&AB,&59
      ,075,&21E,&286,&2DB,&059
      ,05A,&05A,&25
290:DATA &44,&78,&40,&07
      ,000,005,&2B,&24,&0DB
      ,096,&213,&04,&0A,&084
      ,013,&06,&DE
300:DATA 00A,&0B,&24,&47
      ,059,&26,&24,&13,&016
      ,00A,&41,&29,&16,&037
      ,000,000,&ED
370:"A" WAIT 0:PRINT "AT
      TENDEZ":CALL &5A0:
      WAIT
380:RESTORE 110
390:FOR I=&2040 TO &217F
      STEP 16
900:S=0
910:FOR J=0 TO 15
920:READ A:I=A+(S+A) AND 2
      55:POKE I+J,A
930:NEXT J
940:READ A:I IF A<>5 BEEP
      1:PRINT "CS ERROR IN
      "(I-&2040)/1.6+110
      :END
950:NEXT I
960:BEEP 2:PRINT "Comple
      ted."
970:END

```

#### Chargeur N.2

```

100:REM +00 +01 +02 +03
      +04 +05 +06 +07 +08
      +09 +0A +0B +0C +0D
      +0E +0F :0S
110:DATA 002,005,000,010
      ,012,05F,061,010,&0F
      ,034,&43,029,&02,05B
      ,060,001,036
120:DATA &0F,&41,&29,&0D
      ,037,000,000,04D,002
      ,000,&12,05F,&10,&21
      ,043,&61,&82
130:DATA &10,&0F,&4E,&00
      ,060,001,&0F,&42,&052
      ,029,00B,&37,000,000
      ,000,04D,&0C9
140:DATA &02,&0A,&78,&021
      ,082,000,&20,&0C0,&0E8
      ,041,&029,&04,037,000
      ,000,04D,&0D1
150:DATA &03,&078,&002,&07C
      ,0E2,&095,&088,&060,000
      ,090,&78,&21,&0C,E,&91
      ,059,&34,&065

```

```

160:DATA &58,&78,&21,&0D5
      ,05B,&088,&64,&0F,&047
      ,028,&03,&02,&0F,&074
      ,030,&78,&90
170:DATA &00,&4E,&000,005
      ,025,&027,&041,&29,004
      ,037,&05B,&05B,&05B,&05B
      ,037,&040,&04
180:DATA &03,&70,&002,&030
      ,0E2,&095,&0AF,&059,&088
      ,00B,&089,&002,&03,00B
      ,002,&016,&014
190:DATA &0B,&0B,&0B,&0B,&049
      ,02A,&004,&002,&000,&0B
      ,059,&078,&021,&000,000
      ,005,&024,&0D0
200:DATA &27,&041,&029,&004
      ,009,&029,&14,&037,&078
      ,021,&020,&079,&022,&0F6
      ,000,000,&1C
210:DATA &0A,&030,&063,000
      ,028,&0F,070,&02,034
      ,078,&022,&018,&05B,&030
      ,060,000,067
220:DATA &74,&002,&0AA,&071
      ,001,&02A,&012,&060,000
      ,030,&063,000,038,006
      ,071,&002,&072
230:DATA &AA,&070,&001,002
      ,030,&079,&021,&02,030
      ,070,&002,063,008,028
      ,04F,&060,&070
240:DATA &01,&091,&002,&010
      ,000,001,&00C,&03A,&04A
      ,041,&041,&04D,&04D,&04D
      ,04D,&04D,&050
250:DATA &4D,&091,&063,000
      ,028,&035,&0AF,&063,000
      ,038,&00B,&071,&001,002
      ,01B,&078,&0FA
260:DATA &21,&082,&078,&021
      ,0F0,&090,&062,&001,028
      ,015,003,024,&093,&070
      ,041,&059,&020
270:DATA &E2,&09A,&024,&0AD
      ,02B,&059,&060,&00F,&090
      ,063,&050,&069
280:DATA &29,&009,&0AC,&061
      ,001,&002,&010,&078,&021
      ,0B2,&078,&021,&000,002
      ,010,&079,&080
290:DATA &21,&032,&000,003
      ,0E5,&090,&078,&021,&098
      ,0E5,&0A0,&078,&021,&098
      ,041,&029,&0A8
300:DATA &00,&079,&021,&0E0
      ,037,&04D,&005,000,005
      ,025,&026,&041,&029,004
      ,037,&04D,&05D
370:"A" WAIT 0:PRINT "AT
      TENDEZ":CALL &5A0:
      WAIT
380:RESTORE 110
390:FOR I=&2180 TO &223F
      STEP 16

```

```

900:S=0
910:FOR J=0 TO 15
920:READ A:I=S+(S+A) AND 2
55:POKE I+J,A
930:NEXT J
940:READ A:IF A<>S BEEP
1:PRINT "CS ERROR IN
":(I-&2180)/1.6+110
:END
950:NEXT I
960:BEEP 2:PRINT "Complet."
970:END

```

Chargeur n.3

```

100:REM +00 +01 +02 +03
+04 +05 +06 +07 +08
+09 +0A +0B +0C +0D
+0E +0F :CS
110:DATA &F2,&D1,&9A,&07
,&38,&16,&DB,&63,&2B
,&28,&12,&AC,&63,&00
,&28,&16,&62
120:DATA &AB,&03,&24,&59
,&43,&67,&1F,&7E,&22
,&20,&DB,&37,&63,&29
,&29,&04,&7F
130:DATA &AC,&63,&00,&29
,&14,&AB,&03,&27,&59
,&42,&67,&24,&2D,&16
,&00,&00,&8A
140:DATA &78,&21,&20,&78
,&21,&50,&03,&6F,&02
,&FF,&13,&02,&86,&0A
,&03,&24,&E1
150:DATA &02,&3F,&13,&02
,&84,&0A,&00,&28,&24
,&26,&41,&29,&04,&37
,&00,&00,&FB
160:DATA &95,&71,&01,&2A
,&64,&94,&59,&95,&DB
,&A3,&20,&63,&00,&38
,&51,&70,&11
170:DATA &01,&63,&06,&28
,&4B,&34,&60,&00,&03
,&14,&AB,&C7,&28,&00
,&03,&15,&46
180:DATA &AA,&70,&01,&63
,&04,&28,&32,&60,&00
,&38,&DA,&DB,&78,&22
,&F0,&78,&23
190:DATA &21,&98,&AF,&70
,&01,&78,&21,&F0,&78

```

```

,&21,&B5,&AF,&63,&03
,&28,&1C,&09
200:DATA &02,&03,&88,&DB
,&E5,&9C,&00,&50,&78
,&21,&B7,&E5,&A0,&00
,&50,&78,&D6
210:DATA &21,&B7,&49,&29
,&10,&79,&21,&EB,&78
,&21,&80,&5B,&30,&60
,&00,&43,&26
220:DATA &30,&67,&1F,&29
,&59,&78,&23,&80,&37
,&00,&00,&00,&00,&00
,&00,&4D,&D7
230:DATA &92,&71,&01,&2A
,&12,&02,&08,&4D,&4D
,&4D,&92,&DB,&A4,&70
,&01,&60,&13
240:DATA &01,&A7,&70,&01
,&60,&01,&AE,&71,&01
,&2A,&21,&AD,&59,&AE
,&DB,&D0,&44
250:DATA &99,&59,&98,&C4
,&99,&59,&58,&DB,&98
,&59,&99,&C4,&67,&F0
,&2A,&00,&4E
260:DATA &64,&03,&65,&20
,&30,&63,&00,&28,&03
,&61,&01,&37,&78,&23
,&00,&37,&D5
270:DATA &F2,&D1,&9A,&07
,&38,&0A,&60,&00,&67
,&16,&28,&04,&79,&21
,&EA,&37,&2A
280:DATA &78,&22,&F0,&78
,&22,&C0,&78,&23,&00
,&78,&23,&10,&78,&23
,&C0,&2D,&72
290:DATA &10,&4D,&78,&21
,&00,&05,&05,&00,&05
,&25,&26,&41,&29,&04
,&37,&4D,&42
300:DATA &78,&24,&E0,&4D
,&AF,&60,&00,&78,&24
,&F0,&78,&21,&F0,&99
,&44,&03,&CD
370:"A" WAIT 0:PRINT "AT
TENDEZ":CALL &5A0:
WAIT
880:RESTORE 110
890:FOR I=&2400 TO &24FF
STEP 16
900:S=0
910:FOR J=0 TO 15
920:READ A:I=S+(S+A) AND 2
55:POKE I+J,A
930:NEXT J
940:READ A:IF A<>S BEEP
1:PRINT "CS ERROR IN
":(I-&2400)/1.6+110
:END
950:NEXT I
960:BEEP 2:PRINT "Complet."
970:END

```

```

910:FOR J=0 TO 15
920:READ A:I=S+(S+A) AND 2
55:POKE I+J,A
930:NEXT J
940:READ A:IF A<>S BEEP
1:PRINT "CS ERROR IN
":(I-&2200)/1.6+110
:END
950:NEXT I
960:BEEP 2:PRINT "Complet."
970:END

```

```

,7F,&00,&60,&60,&00
,&00,&7F,&D6
200:DATA &79,&09,&09,&06
,&78,&7F,&09,&09,&7F
,&7F,&79,&01,&01,&7E
,&00,&00,&91
210:DATA &7F,&78,&00,&7F
,&79,&41,&41,&63,&A2
,&73,&B2,&63,&73,&A2
,&A2,&82,&37
220:DATA &C1,&B1,&93,&B1
,&83,&72,&C1,&B2,&B1
,&82,&54,&54,&63,&72
,&44,&53,&65
230:DATA &85,&83,&73,&B1
,&82,&82,&64,&81,&45
,&72,&63,&53,&56,&44
,&82,&81,&1F
240:DATA &73,&91,&53,&846
,&39,&64,&91,&A1,&F0
,&F0,&4D,&4D,&4D,&4D
,&4D,&4D,&1A
250:DATA &E5,&A0,&90,&00
,&07,&02,&00,&1E,&10
,&70,&00,&00,&3B,&1F
,&37,&00,&4D
260:DATA &78,&21,&C0,&03
,&70,&02,&6D,&E2,&95
,&02,&30,&79,&21,&DF
,&00,&00,&5D
370:"A" WAIT 0:PRINT "AT
TENDEZ":CALL &5A0:
WAIT
880:RESTORE 110
890:FOR I=&2400 TO &24FF
STEP 16
900:S=0
910:FOR J=0 TO 15
920:READ A:I=S+(S+A) AND 2
55:POKE I+J,A
930:NEXT J
940:READ A:IF A<>S BEEP
1:PRINT "CS ERROR IN
":(I-&2400)/1.6+110
:END
950:NEXT I
960:BEEP 2:PRINT "Complet."
970:END

```

## CODES ASSEMBLEUR

**Le PC-1600 comporte deux microprocesseurs : le SC-7852 compatible Z-80 et le LH-5803. Dans cet article, on va comparer les codes assembleur du Z-80 et du SC-7852 puis ceux du LH-**

### 5803 et du LH-5801 du PC-1500.

D'abord le SC-7852 peut gérer huit pages mémoires de 0 à 7 et possède donc plusieurs instructions en plus par rapport au Z-80. Les codes du Z-80 se trouvant très facilement dans les livres d'informatique, je ne les donnerai pas dans cet article. Les instructions supplémentaires du SC-7852 correspondent aux fonctions RST du Z-80.  
&C7 - Code du reset du SC-7852 et enclenche la routine d'initialisation.  
&CF - Ne correspond à aucune fonction.  
&D7 - Ne correspond à aucune fonction.

&DF 010h - 01 octet faible de l'adresse de saut  
0h octet fort de l'adresse de saut.

Exécute un saut inconditionnel à l'adresse 010h (JP 010h) en page 6 ou 3. Cette instruction est suivie de deux octets définissant à la fois l'adresse de saut et le numéro de page.

- Si 0h>&80 le saut s'effectue en page 6 à l'adresse 010h
- Si 0h<&80 le saut s'effectue en page 3 à l'adresse 010h.

Je rappelle que la mémoire morte programme de la page 3 va de &4000 à &7FFF, et celle de la page 6 va de &8000 à &BFFF.

&E7 a 010h :

- 01 octet faible de l'adresse de saut
- 0h octet fort de l'adresse de saut
- a numéro de la page où s'effectue le saut.

Exécute un CALL à l'adresse 010h de la page a.

Au premier RET code &C9 le programme continue son exécution dans la page où se situe le CALL.

&EF - ne correspond à aucune fonction.

&F7 - effectue un saut à l'adresse &F0D4 en page 0 (JP &F0D4).

&FF - effectue un saut à l'adresse &F0D7 en page 0 (JP &F0D7).

Maurice CHOCROUN

# NEWS

**Possesseur d'un PC-1500 depuis le mois de mai 1982, je suis tombé sous le charme du 1600 en le voyant à Paris début juillet.**

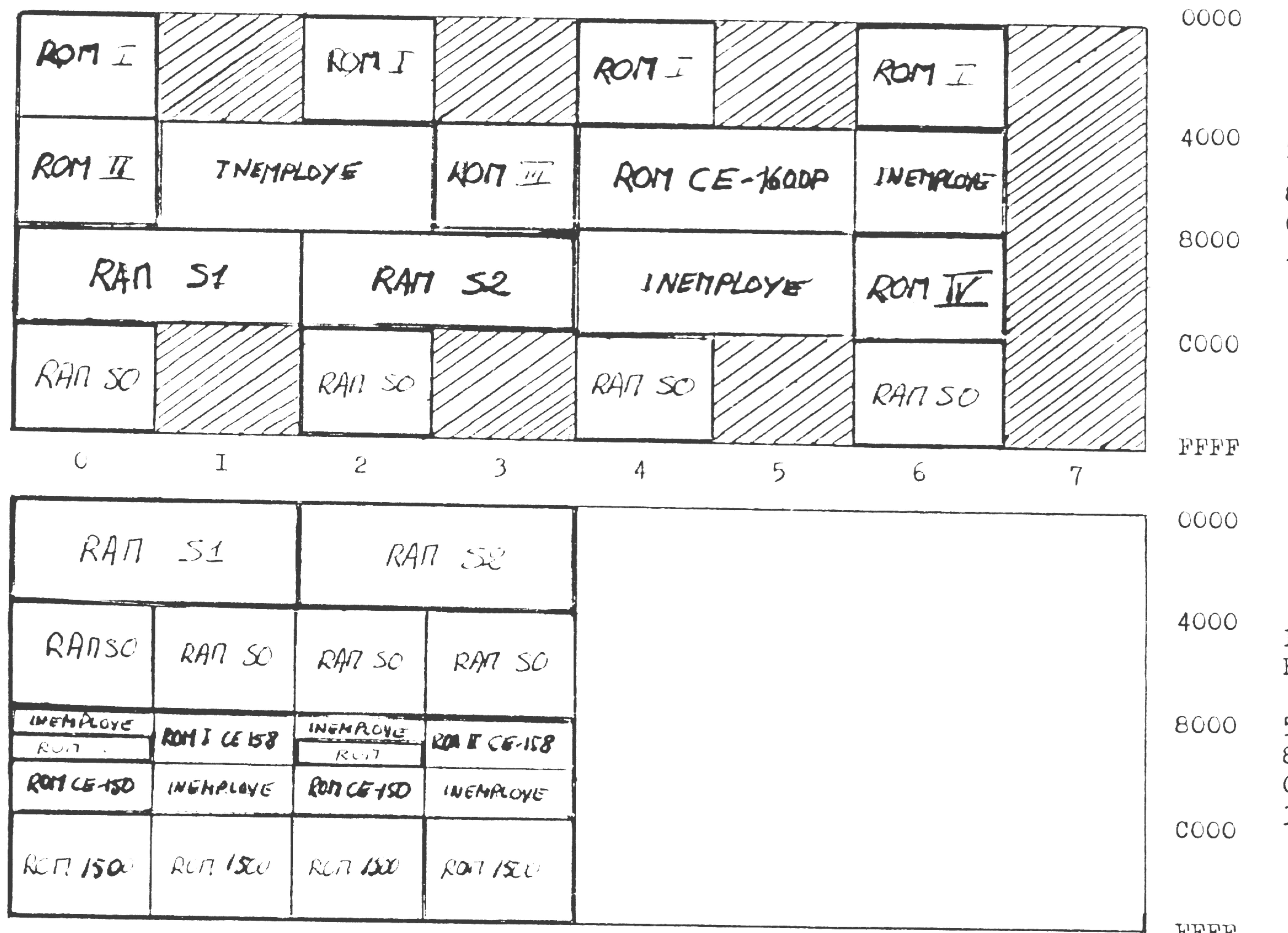
**Après quatre années de bons et loyaux services, le 1500 se voyait donc remplacé par le petit nouveau. Et c'est pour vous parler de lui que je vous écris.**

Je dirais tout d'abord que c'est une machine fantastique qui est à la fois plus puissante que le 1500, plus rapide et plus souple. Que demander de mieux ? Ma seule déception concerne la compatibilité que je trouve relative avec le 1500, mais j'y reviendrai plus tard.

Je voudrais tout d'abord vous donner quelques astuces que j'ai découvertes durant un mois d'utilisation.

Si le manuel nous donne une carte de la mémoire du 1600, celle-ci est assez incomplète comme vous avez dû le constater vous-même.

Voici la carte telle que j'ai pu l'établir :



Et c'est là que j'ai un problème que vous allez sans doute pouvoir résoudre : on peut compter 4 zones de ROM de 16 ko chacunes dans l'aire du Z-80A, or cela ne fait que 64 ko et non 96 comme indiqué par SHARP. Comme il n'y a pas de raisons de douter de ce que dit SHARP, la question se pose : où sont les 32 ko manquants ? Ce ne sont pas ceux de la ROM de la CE-1600P puisqu'ils sont dans le périphérique et non dans l'unité centrale.

Il faut alors peut-être compter la ROM adressée par le LH 5803. On constate sur la carte que celle-ci est constituée des 16 ko qui constituent la ROM du 1500 et de 4 ko supplémentaires. Cela ne nous fait que 20 ko. On arrive donc à trouver 80 ko de ROM dans le 1600. Il m'en manque 16. Les avez-vous trouvés ?

Je vous avouerai que lorsque j'ai découvert que la ROM adressée par le LH était celle du 1500 à quelques octets près, j'ai trouvé ça génial : réussir à mettre un 1600 et un 1500 avec son processeur et sa ROM sous un volume identique au vieux 1500, cela donne une idée de la différence de

technologie entre les deux machines. Je me suis alors dit que les logiciels que j'avais écrits et qui utilisaient des possibilités avancées du 1500 allaient pouvoir tourner sur le 1600 puisqu'en mode 1 c'était la ROM du 1500 qui était adressée par le LH-5803. Fort de cette certitude, je lance un programme utilisant la possibilité de détournement du driver clavier et... rien !! Histoire de vérifier je tente la manœuvre décrite dans le « Sharpentier » n° 10, et toujours rien. Me suis-je mal débrouillé où est-il impossible d'utiliser cette astuce ? Le désassemblage de la ROM à l'adresse E243 est pourtant identique à celui du 1500. Je veux bien que le clavier soit scruté au travers de la ROM spécifique au 1600, mais alors pourquoi avoir pris la peine de remettre cette routine dans la ROM ? Déçu, je charge mon logiciel de jeu préféré, écrit en collaboration avec un copain, et qui utilise pour l'affichage des messages de caractères redéfinis. C'est une horreur !!! J'ai droit aux symboles graphiques de l'IBM-PC mais pas à mes caractères. Là encore, même avec un autre logiciel utilisant cette intéressante possibilité du 1500, échec ! Ce

qui met à mal la compatibilité annoncée par SHARP ! Je ne vous surprendrai pas en vous disant que l'astuce permettant la réassignation du clavier avec ht ne fonctionne pas, etc. Je n'ai pas encore eu le temps d'essayer un logiciel basé sur la routine d'interruption publié dans le « Sharpentier » n° 6, mais j'ai peur ! Je vais chercher à faire fonctionner tout ça malgré ces premiers échecs mais je voudrais savoir si vous-même n'avez pas déjà trouvé comment faire. Il serait dommage que les logiciels les plus avancés, donc souvent les plus intéressants, ne fonctionnent pas sur le 1600.

Concernant le LM j'ai eu une autre surprise. Je charge le logiciel « PLANET FIGHTER » publié dans un « Sharpentier » et je tape : XCALL &41D0. Au vu de la rapidité de déplacement de l'avion de combat, j'ai cru avoir à faire à un logiciel BASIC sur PC-1211 !!! L'intérêt de ce logiciel sur le 1500 étant sa rapidité liée au LM c'est un comble !

En fait, tous les programmes LM LH-5803 me semblent tourner nettement moins vite sur le 1600 que sur le 1500 (en particulier Planet). Cela est

étonnant car si l'on en croit la notice, le LH-5803 tourne à 1.3 Mhz comme le LH-5801. Avez-vous une explication à me donner ?

Pour terminer cette partie LM, j'ajouterais que je souhaite que SHARP publie rapidement un TRM pour le 1600.

Jérôme GAUDIN

**UN PROGRAMME À DÉVELOPPER ?  
CONTACTEZ NOUS !**

# LE PORT D'ENTRÉE ANALOGIQUE

**Le PC 1600, quelle détonnante machine !!!**

**Ce coup-ci, on a vraiment fait un effort pour avoir un micro puissant dans « la poche » chez SHARP.**

Olivier PROTT

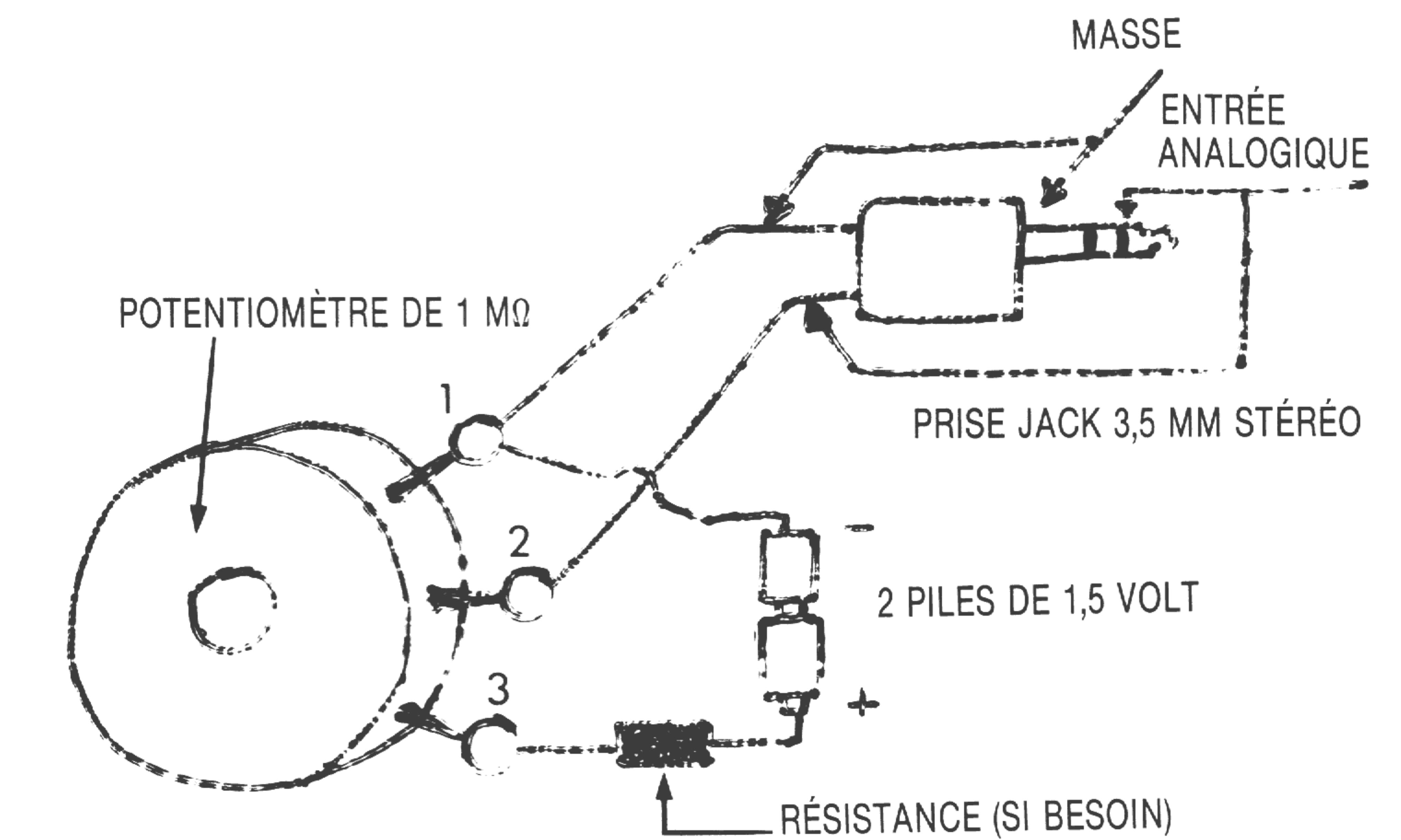
En effet, tout est prévu (d'après la documentation L.M) pour mettre dans le logement S2, 1 Mo de ROM ou 512 Ko de RAM sous forme de disque mémoire.

A quand les mémoires de 512 Ko de RAM dans un boîtier d'extension !!! Le PC 1600 possède aussi un port d'entrée analogique. La documentation est très mal faite à ce sujet. Voici un petit schéma pour s'amuser avec.

Pour vérifier le montage, s'assurer d'abord à l'aide d'un voltmètre que entre 1 et 3, il n'y a pas plus de 3 volts, sinon rajouter une résistance comme indiqué puis écrire :

10 A = AIN:P.A.:WAIT10:  
CLS:G.10

Amusez-vous bien !



tous les PC

## INITIATION AU LM ESR-H (7<sup>e</sup>)

**Pour notre septième rencontre, nous allons revenir sur un sujet déjà abordé et que vous attendez tous : les Moves et EXchanges. Ces instructions sont très puissantes et nous allons le voir, très intéressantes pour déplacer des blocs de mémoires pointés par les registres P, Q ou DP.**

En effet, nous n'avions étudié jusqu'ici que des instructions affectant ou échangeant un seul registre ou une seule adresse mémoire. Par exemple, MVMD : (DP) → (P). Imaginons maintenant une fonction permettant de renouveler cette opération n fois pour des adresses et des registres contigus. Il faudrait pour cela répéter MVMD n fois en incrémentant de 1 DP et P. Nous avons deux instructions pour faire cela : MVWD et MVBD. Le W signifie que l'on prend I pour compteur d'incrémentation et B indique J. Avant donc d'utiliser MVWD ou MVBD, il faut charger dans I ou J la valeur n-1 correspondant au nombre de Move à faire ; n-1 fois, car l'instruction est exécutée au moins une fois. Pour ce faire, la valeur dans I ou J est automatiquement copiée dans un registre intermédiaire (non utilisable directement) appelé d.

MVWD : code &18 I → d  
On répète (DP) → (P), DP + 1, P + 1, d - 1  
Jusqu'à ce que d = &FF

Si I ou J est nul (d = 0) un seul Move est

effectué. De même :

MVBD : code &1A J → d  
On répète (DP) → (P), DP + 1, P + 1, d - 1  
jusqu'à ce que d = &FF

Il est évident que P et DP sont modifiés car incrémentés. Les flags ne sont pas touchés et les registres I et J ne sont pas affectés. Notons que par exemple si P = 8, DP = &6900 et I = 1, après MVWD P = &0A, DP = &6902 et I = 1.

Comme J est par définition égal à 1, cela permet des Moves (et nous le verrons des échanges) sur 16 bits, pour des pointeurs d'adresse par exemple.

Concrètement on peut expliciter ces mnémoniques en écrivant d'une autre manière la fonction accomplie :

MVWD : (DP) → (P), (DP + 1) → (P + 1),...  
(DP + I - 1) → (P + I - 1)

Prenons DP = &6F01, P = 4, alors MVBD mettra le contenu de &6F01 et &6F02 respectivement dans XI et Xh. Soit l'adresse de début de la zone Basic (pour le 1350 :START BASIC:&6F01/02) dans X. L'opération inverse ne

peut se faire que par un échange avec EXWD ou EXBD.

Ce qui oblige pour éviter de perdre le contenu de ce qui est pointé par P (soit X) de refaire un Move. Exemple :

LP &04 - XI pointé par P  
LIDP &6F01 - START BASIC pointé par DP

EXBD - mettra dans START BASIC l'adresse dans X, mais  
LP &04

LIDP &6F01  
MVBD - placera dans X la valeur initiale sans modifier START BASIC.  
Notons que LIJ &01 n'est pas utilisé car J vaut déjà 1.

Voyons ce que donne EXWD et EXBD :

EXWD : code &19 I → d  
On répète (DP) ↔ (P), DP + 1, P + 1, d - 1  
jusqu'à ce que d = &FF

EXBD : code &1B J → d  
On répète (DP) ↔ (P), DP + 1, P + 1, d - 1  
jusqu'à ce que d = &FF

Là aussi les flags et les registres I et J ne sont pas modifiés. Mais DP et P sont incrémentés.

Nous pouvons aussi copier une partie de la RAM interne dans une autre partie de la RAM interne. Pour ceux qui seraient distraits, je signale que nous avons vu plus haut comment copier (ou échanger) une partie de la RAM externe (donc la RAM programme) vers la RAM interne. Nous utilisions pour cela les registres d'index DP et P. Comme nous avons deux registres d'index pour la RAM interne, P et Q (ils sont rusés ces Japonais, n'est-ce-pas ?), nous devi-

nons tout de suite ce que font MVW et MVB :

MVW : code &08 I → d  
On répète (Q) → (P), Q + 1, P + 1, d - 1  
jusqu'à ce que d = &FF

Idem pour MVB, avec J → d de code &0A

Et on a aussi EXW et EXB :

EXW : code &09 J → d  
On répète (P) ↔ (Q), P + 1, Q + 1, d - 1  
jusqu'à ce que d = &FF

Idem pour EXB, avec J → d de code &0B

P et Q sont incrémentés. I, J et les flags ne sont pas touchés.

Exemple : pour copier Xreg dans Yreg  
LP &18 - P pointe sur premier octet de Yreg

LIQ &10 - Q pointe sur premier octet de Xreg

LII &07 - on répètera le move 8 fois  
MVW - notons bien que l'on va de Q vers P.

Xreg et Yreg sont donnés pour tous les PC sauf 1245/51/55 (voir numéro 17 page 30).

Bon, je fatigue, alors j'arrête là. La prochaine fois nous terminerons ce chapitre en parlant des instructions FILM et FILD, et nous expliquerons comment copier un bloc d'octets de la RAM externe vers la RAM externe. Grâce à X et Y, au travers des mnémoniques IXL, DXL et IYS, DYS. Pendant que l'on y sera, on en profitera pour faire un sort à IX, IY, DX et DY.

Ouf ! ! FINI.

Jean-François VIGNAUD





# GESMAIL

**«Gesmail» est un logiciel de gestion d'un fichier d'adresses pour ordinateur SHARP MZ 700 muni d'une imprimante et d'un lecteur de disquettes KERSTEN & PARTNER.**

**Dès la mise en route du programme, il vous sera posé la question « Travail sur fichier existant » qui vous permettra de mettre en mémoire un fichier se trouvant sur disquette. Puis il vous sera proposé le menu principal sur deux**

**lignes en bas de l'écran. Pour choisir une option, déplacez-vous à l'aide de la barre d'espace, puis validez l'option retenue en tapant « CR ».**

Dix options sont possibles :

**LIRE, FEUILLETER, CRÉER, MODIFIER, SUPPRIMER, TRI, MAILING, SAUVEGARDER, ÉDITER, ARRÊT.**

**LIRE** : Cette option vous permettra de rechercher une fiche précise. Il vous sera demandé d'entrer les deux personnes données de la fiche. Celle-ci vous sera alors affichée en cas de recherche positive.

**FEUILLETER** : Cette option vous permettra de vous promener dans le fichier. Un nouveau menu vous sera alors présenté :

↓ pour avancer dans le fichier

↑ pour revenir en arrière  
← pour retourner au menu principal.

**CRÉER** : Il vous sera alors demandé d'entrer chacune des 8 données possibles constituant une fiche.

**MODIFIER** : Pour corriger une ou plusieurs zones de la fiche. Il vous faut d'abord entrer les deux premières données. La fiche à corriger sera alors visible à l'écran. Tapez alors le numéro de la rubrique à modifier puis entrer le nouveau contenu de cette zone.

**SUPPRIMER** : Entrer les deux premières données de la fiche à supprimer. Cette suppression se fera automatiquement après que vous en ayez donné la confirmation.

**TRI** : Ce tri s'effectue à partir de la première zone de la fiche et dans l'ordre croissant.

**MAILING** : Pour effectuer des étiquettes d'adresses, trois possibilités d'impression :

1) Mailing d'une fiche - entrer alors les deux premières données de cette fiche.

2) Mailing du fichier.  
3) Mailing par critère - le Mailing se fera pour les fiches répondant à la condition que vous aurez indiquée. Pour cela, tapez d'abord le numéro de la rubrique concernée puis entrez la nature du critère. L'impression pourra alors commencer.

**SAUVEGARDER** : Pour stocker votre fichier sur disquette, insérez votre disquette de sauvegarde et à la question « Disquette prête O/N », tapez O.

**EDITER** : Toutes les zones de chacun des enregistrements du fichier seront imprimées.

**ARRÊT** : Pour sortir du programme. Le retour au menu principal vous sera signalé par l'affichage de l'inscription « MENU » au-dessus des lignes d'options.

Maurice GODEAUX

```

5 REM*****
10 REM***GESMAIL 4.1-0 2/1986***
20 REM*****GODEAUX Maurice *****
30 REM*****
90 CONSOLE:COLOR,,7,0:CLS
100 DIM AD$(10),AD$(8,100),T$(8),FR$(8),CD(10,2)
120 FORW=1TO8
130 READFR$(W)
140 NEXTW
150 FORT=1TO10
160 READAD$(Y)
170 NEXTY
180 DATA 1.Nom ,2.Prenom ,3.Adresse ,4.Code postal ,5.Ville ,6.Pays ,7.Téléphone
,8.Indications
190 DATA Lire,Feuilleter,Créer,Corriger,Supprimer,Trier,Mailing,Editer,Sauvegarder,Arrêt
200 FOR Y=1 TO 10:READ CD(Y,1),CD(Y,2):NEXT Y
210 DATA -1,23,4,23,15,23,21,23,30,23,0,24,6,24,14,24,21,24,33,24
220 F$="AD.PER.":CF=0
230 CURSOR2,0:PRINT"Travail sur fichier existant ?"
240 GETG$:
250 IFG$="N"THEN450
260 IFG$="O"THEN320
265 GOTO 240
269 REM***CHARGEMENT FICHIER***:
320 PRINT[2,2]"Disquette prête [O/N] ?"
325 GET G$:IF G$<>"O" THEN 325
330 ROPEN#2,F$:
340 PRINTTAB(10)"File chargé le fichier:":F
345 PRINT[0,2]TAB(10)F$
350 INPUT#2,I
360 FORZ=1TO1
370 FORT=1TO8
380 INPUT#2,AD$(Y,2)
390 NEXTZ:NEXTZ
400 CLOSE#2:CF=1
452 CLS:GOSUB 20000
499 REM*****MENU*****
500 CONSOLE:USR(62)
501 H=1:GOSUB 22000
510 COLOR,,0,2:CURSOR 0,0:PRINT USING "####";SIZE:PRINT" Set disp."
515 CURSOR 0,22:PRINT[0,5]"----- MENU -----";
520 COLOR,,7,0:CONSOLE 1,22,0,42
530 GET G$:
540 IF G$=CHR$(13)THEN 600
550 IF G$<>" " THEN 530
560 H=H+1:IF H>10 THEN H=1
570 GOSUB 22000
580 GOTO 530
600 ON H GOTO 2000,15000,1000,3000,4000,5000,6000,7000,8000,21000
630 GOTO530
999 REM***CRÉATION D'UNE FICHE**:
1000 USR(62):CLS:I=I+
1010 FORT=1TO8
1030 PRINT[0,4]FR$(Y):INPUT" AD$(Y,1)
1040 NEXTY
1045 PRINT[0,2]"Création d'une autre fiche [O/N]?"
1050 GET G$:
1060 IFG$="O"THEN1000
1070 IFG$="N"THEN500
1080 GOTO1050
1999 REM***LECTURE D'UNE FICHE**:
2000 USR(62):CLS:PRINT[6,1]FR$(1):INPUTR1$:
2010 PRINT[6,1]FR$(2):INPUTR2$:

```

```

2020 GOSUB10000
2030 GOSUB11000
2130 GOTO500
2999 REM***CORRECTION***:
3000 USR(62):CLS:PRINT[2,2]"Fiche à corriger"
3010 PRINTER$(1):INPUTR1$:
3020 PRINTER$(2):INPUTR2$:
3030 CLS
3040 GOSUB10000
3050 GOSUB11000
3100 CURSOR 0,20:PRINT[2,2]"Taper le No de la rubrique à corriger"
3102 GET C$:IF C$=VAL(C$)
3104 IF (C<1)+(C>8)THEN 3102
3110 CURSOR 0,20:PRINT[2,2]"Indiquer la correction à effectuer :":ON ERROR GOT
03100:INPUTAD$(C,8)
3140 GOSUB11000
3150 CURSOR 0,20:PRINTSPC(39)
3160 CURSOR 0,20:PRINT[2,2]"Autres corrections [O/N]?"
3170 GETG$:
3180 IFG$="O":GOT03102
3190 IFG$="N":GOT500
3200 GOT03120
3999 REM***SUPPRESSION***:
4000 USR(62):CLS
4010 PRINT[2,2]"Fiche à supprimer"
4020 PRINTER$(1):INPUTR1$:
4030 PRINTER$(2):INPUTR2$:
4040 GOSUB10000
4050 GOSUB11000
4054 PRINT[2,2]"Est ce bien la fiche à supprimer?"
4055 GET G$:
4056 IF G$="O" THEN 4060
4057 IF G$="N" THEN 500
4058 GOTC 4055
4060 FORB=2TO1-1
4070 FORBB=1TO8
4080 AD$(BB,B)=AD$(BB,B+1)
4090 NEXTBB
4100 NEXTB
4110 I=I-1
4120 PRINT[2,2]"Fiche supprimée"
4130 PRINT[2,2]"Autre fiche à supprimer [O/N]?"
4140 GETG$:
4150 IFG$="O":GOT4000
4160 IFG$="N":GOT500
4170 GOTC4140
4399 REM***TRI***:
5000 USR(62):CLS:CURSCR1,1:PRINT[2,2]"JE TRI...":
5005 IU=0
5010 FORZ=1TO1-1
5020 IFAD$(1,Z)<AD$(1,Z+1)THEN5080
5030 FORTR=1TO8
5040 T$=TR0=AD$(TR,2)
5050 AD$(TR,2)=AD$(TR,2+1)
5060 AD$(TR,2+1)=T$(TR)
5070 NEXTTR:IU=1
5080 NEXTZ
5090 IFIU=1THEN5005
5100 CLS:GOT500
5999 REM***MAILING***:
6000 USR(62):CLS:CURSCR5:PRINT"MAILING D'UNE FICHE"
6010 PRINTTAB(5)"MAILING DU FICHIER"
6020 PRINTTAB(5)"MAILING PAR CRITÈRE"
6030 PRINT"Votre choix:"
6040 INPUTC:CLS:ONCGOTO6100,6200,6300

```

# ASTUCES

```

6100 PRINT#(1):INPUT#1$  

6110 PRINT#(1,2):INPUT#2$  

6120 GOSUB10000  

6130 GOSUB9000  

6140 GOT0500  

6200 FOR#=1TO1  

6210 GOSUB9000  

6220 NEXT#  

6230 GOT0500  

6300 FORT=1TO8  

6310 PRINT#(Y)  

6320 NEXT#  

6330 CONSOLE#20,5,0,40  

6340 INPUT"Choix du critere-->":CR  

6350 INPUT"Nature du critere-->":CR$  

6360 WW=0  

6370 FOR#=1TO1  

6380 IFAD$(CR,$)=CR$ THEN WW=1:GOSUB9000  

6390 NEXT#  

6500 IFWW=0 THEN PRINT#(2,7)"Aucun enregistrement ne répond à ces conditions":GOT  

0 500  

6520 GETG$  

6999 REM***EDITION FICHIER***  

7000 PCOLOR 1  

7010 FOR#=1TO1  

7020 PRINT/PAD$(1,8),AD$(2,8)  

7030 PRINT/PAD$(3,8)  

7040 PRINT/PAD$(4,8),AD$(5,8)  

7050 PRINT/PAD$(6,8)  

7060 PRINT/PAD$(7,8),AD$(8,8)  

7080 SKIP1:NEXT#  

7090 GOT0500  

7999 REM**SAUVEGARDE FICHIER**  

8000 CLS  

8220 PRINT#(2,7)"Disquette prête (O/N)?"  

8230 GETG$  

8240 IFG$<>"O":THEN8230  

8250 IF CF=1 THEN DELETE F$  

8265 WOPEN#2,F$  

8270 PRINT#2,I:FOR#=1TO I  

8280 FORT=1TO8  

8290 PRINT#2,AD$(Y,8)  

8300 NEXT#  

8310 CLOSE#2:PRINT#(2,2)"Le fichier ""F$"" est sauvegardé"  

8320 GOT0500  

8999 REM***ETIQUETTES***  

9000 PCOLOR1  

9110 PRINT/PTAB(2)AD$(1,8);TAB(22)AD$(2,8)  

9120 PRINT/PTAB(2)AD$(3,8)  

9130 PRINT/PTAB(2)AD$(4,8);TAB(22)AD$(5,8)  

9140 IFAD$(6,8)<>"FRANCE":THENPRINT/PTAB(2)AD$(6,8)  

9150 SKIP1:RETURN  

9999 REM**RECHERCHE FICHE**  

10000 FOR#=1TO1  

10010 IF (AD$(1,8)=R1$)*(AD$(2,8)=R2$) THEN RETURN  

10030 NEXT#  

10040 PRINT#(2,7)"Enregistrement inconnu du fichier"  

10080 GOT0500  

10399 REM***AFFICHAGE FICHE**  

11000 USR(62):CLS:COLOR,,2,2  

11100 CURSOR 0,:PRINT#(1):TAB(20)FR$(1)  

11110 CURSOR 0,4:PRINT#(3)  

11120 CURSOR 0,8:PRINT#(4):TAB(20)FR$(5)  

11130 CURSOR 0,11:PRINT#(6)  

11140 CURSOR 0,14:PRINT#(7):TAB(20)FR$(8)  

11160 COLOR,,2,0  

11170 CURSOR 0,2:PRINTAD$(1,8);TAB(20)AD$(2,8)  

11180 CURSOR 0,5:PRINTAD$(3,8)  

11190 CURSOR0,9:PRINTAD$(4,8);TAB(20)AD$(5,8)  

11200 CURSOR 0,12:PRINTAD$(6,8)  

11210 CURSOR 0,15:PRINTAD$(7,8);TAB(20)AD$(8,8)  

11220 RETURN  

14999 REM ***FEUILLETER*****  

15000 CURSOR0,22:PRINT#(0,4)" ↑ Avance ↑ Reviens ← Menu principal ";  

15010 CONSOLE 1,21,0,40:CLS:#1:GOSUB 11000  

15020 PRINT CHR$(23)  

15030 GET G$  

15040 IF G$="W" THEN Z=Z+1:GOTO15060  

15050 IF G$="B" THEN Z=Z-1:GOTO15060  

15060 IF G$="E" THEN PRINTCHR$(25):GOTC500  

15070 GOTC15020  

15080 IF Z>1 THEN Z=1:GOTO15020  

15090 IF Z<1 THEN Z=1:GOTO15020  

15100 GOSUB 11000:GOT015020  

19993 REM***INITIALISATION ECRAN**  

20000 CONSOLE23,2,0,40:COLOR,,2,3:CLS  

20005 CONSOLE  

20010 CURSOR0,23:FOR#=1TO10:PRINT#(X);  

20020 IF X<10 PRINT" ";  

20030 NEXT  

20050 RETURN  

20999 REM ***FIN DE PROGRAMME***  

21000 END  

21999 REM ***CHOIX OPTIONS***  

22000 FOR TU=1 TO LEN(O$(U))  

22010 COLOR COC(U,1)+TU,COC(U,2),0,2  

22020 NEXT TU  

22030 FOR TU=1 TO LEN(O$(H))  

22040 COLOR COC(H,1)+TU,COC(H,2),0,2  

22050 NEXT TU  

22060 U=H:RETURN

```

## PROTECTIONS POUR PROGRAMMES EN S-BASIC

Protéger un programme Basic a toujours été un problème. Pourtant, en supprimant l'effet de certaines instructions, on empêche le PIRATAGE ! Commandes invalidées : LIST, SAVE, PEEK, BYE.

Un message d'erreur apparaîtra lors de l'utilisation d'une de ces quatre commandes.

### Annulation de LIST par :

POKE 12629,254,32 ou POKE 16642,201

### Annulation de SAVE par :

POKE 12721,254,32 ou POKE 17060,201

### Annulation de PEEK par :

POKE 12919,254,32 ou POKE 26037,201

### Annulation de BYE par :

POKE 12751,254,32 ou POKE 5058,201.

### Réactivation de LIST par :

POKE 12629,2,65 ou POKE 16642,175

### Réactivation de SAVE par :

POKE 12721,164,66 ou POKE 17060,205

### Réactivation de PEEK par :

POKE 12919,181,101 ou POKE 26037,254

### Réactivation de BYE par :

POKE 12751,194,19 ou POKE 5058,229.

Ces POKEs sont à mettre au début du programme, et sont tirés de la revue « Fischel ».

N.B. - Ayez pitié du Club, ne nous envoyez pas de programmes protégés.

**NOUVEAU  
TÉLÉPHONE :  
42.74.07.68**

# GRAFITIK

**Adieu les longues soirées passées sur le papier millimétré à mitonner de superbes présentations graphiques pour vos programmes ! Avec GRAFITIK, vous dessinez librement sur l'écran et le MZ 800 se charge de vous générer le programme correspondant.**

La création d'un dessin avec cet utilitaire se fait en deux parties : on se sert d'abord de GRAFITIK.DES pour créer le graphisme à l'écran puis, une fois qu'il est sauvégarde, on exécute GRAFITIK.CONV pour le transformer en programme BASIC.

### 1) GRAFITIK.DES

Pour dessiner, on déplace le curseur (une petite croix) avec les touches fléchées, la barre d'espace servant d'accélérateur.

Lorsque l'on est placé sur la position désirée, on sélectionne le point grâce à la touche « Joker » située au-dessus de CR.

On dispose des fonctions suivantes :

- F1 annule la dernière sélection.
- F2 trace un segment entre la position courante et la dernière sélection.
- F3 trace un cercle passant par la position courante et ayant pour centre la dernière sélection.
- F4 trace un rectangle passant par la position courante et la dernière sélection.

- F5 colorie la zone autour de la dernière sélection (on doit être positionné sur la frontière de la zone à colorier).

- F6 appelle le menu des fonctions spéciales (lecture, écriture, correction, fin du programme).

- F9 et F10 permettent de choisir la couleur du tracé qui apparaît autour de l'écran.

Les adeptes de la religion polonaise inversée auront reconnu l'emploi d'une pile (on peut sélectionner jusqu'à 35 points successivement) et d'opérateurs (graphiques !) sur le sommet de cette pile.

```

99 *****INITIALISATIONS
100 INIT CRT-M2: COLOR 15,0
110 LIMIT#FB00:POKE#FB00,$3E,15,1,$0E,6,
$ED,$29,$C9:COLOR 15:USR#FB00
120 DEF FNS#(K):CHR$#K:128#D1+CHR#K MOD
128#D0
130 DEF FNN#(X):ASC#MID#CMD#,X,100-D1#
28+ASC#M1#CMD#,X#1,100-D1#
140 DTM PC255#M#C255,101
160 KEP#160:TF#100:CF#2:CF#15
170 FOR T#0 TO 9:DEFKEY#(1+T#1) IN #ESTR#(1
+NEXT#)

```

## 3D LABY

**Ce programme est un contre-exemple des tendances de l'informatique actuelle : il est mal écrit, pas structuré, pas commenté et utilise des ruses mesquines pour gagner quelques instructions. Mais le résultat est un jeu facile à taper et assez impressionnant.**

Le principe est simple : vous devez sortir d'un labyrinthe. Mais (à part une brève vision en plan au début du jeu) vous voyez ce labyrinthe de l'intérieur avec des vues en perspective.

Pour vous déplacer :

- la barre d'espace vous fait avancer
- vous fait tourner à droite
- ← vous fait tourner à gauche.

La sortie est une porte dérobée située à l'autre bout du terrain. Par exemple sur un labyrinthe de taille 14, vous partez de la case (1,1) et la sortie est en (13,13).

Si vous préférez voir se tracer les dessins en perspective au lieu d'avoir une apparition soudaine, il faut supprimer les instructions PAL de la ligne 38.

Bon courage...

```

35 IF ERU<28 AND ERU>22 PRINT ERU,ERU
R,ERU
36 S=S+1,NOISE C01,RESUME 32
37 IF D THEN P=R,D=3,P 2,F,SF,P=R,D=2,F=3
38 R=R22,CLS,PAI2,0,PAI3,0,PR0,1410,T0,0
STEP,-1
39 R=EXP(PI)-10*,480,RR,RXR
40 IF NOT CM, P,0 THEN BOX(T0,160-RR,100-R,
160+RR,100-R),P,I=2,GOTO45
41 IF NOT T0G,P,0 THEN LINE(T0,160-RR,100-
R,160-R),P,0,160-R,100-R,160+R,100+R,160-R
R,100+R,160-RR,100-R,PAINT(T0,160-R,13-R)
R,2,100,C
42 IF T0G,P,0,0 AND NOT T0G,P,0,1 THEN
BOX(T0,160-RR,100-R),160-R,100-R,1,0
43 IF NOT T0D,P,0 THEN LINE(T0,160+RR,100+
R,160+R),P,0,160+R,100+R,160-R,100+R,160+R
R,100+R,160+R,100+R,PAINT(T0,160+R,13+R)
R,2,100,C
44 IF T0D,P,0,0 AND NOT T0D,P,0,1 THEN
BOX(T0,160+RR,100-R),160+R,100+R,1,0
45 P=P+ST,2,PAI1,PAI2,2,PAI3,4,PR0,0
N

```

## ASTUCES

### COLORIEZ...

Voici un sous-programme qui, je pense, intéressera plus d'un Sharpentier. La routine Basic ci-dessous pour MZ-800 colorie selon la valeur de A le contour inutilisé de l'écran. Mettre dans A le code de la couleur désirée. Ce petit programme peut être incorporé dans un programme Basic ou LM. On pourra ainsi colorier tout l'écran sans toujours avoir ce triste contour noir.

Jérôme GESLIN

```

1 ON ERROR GOTO 35
2 GOSUB20
3 R=320/200,X=1,T=1
4 S=S MOD 4,ONS=1,GOSUB 12,13,14,15
5 GOSUB37,CURSOR17,0,PRINT,K,Y
6 IF X=A-1 AND Y=A-1 THEN 16
7 GET S$:IF S$=1 THEN 7
8 IF S$=0 THEN S$=1,GOTO 9
9 IF S$=2 THEN S$=3,GOTO 10
10 IF G$>1 THEN 2,ELSE IF NOT T0X=X,A,THEN
A,0,0 THEN NOISE T0,15$66M30$R2F4,1-GOTO
2,FLSF,X=X+A,1+1-A,GOTO 5
11 GOTO5
12 YA=-1,XA=0,M-XA,XA=-1,DAX=-1,STA=1,RE
TURN
13 YA=0,XA=1,M+XA,XA=1,DAX=1,STA=1,RE
TURN
14 YA=1,XA=0,M-XA,XA=1,DAX=1,DAX=1,STA=1,RE
TURN
15 YA=0,XA=1,M-XA,XA=1,DAX=1,STA=1,RE
TURN
16 SOUND 11,11,SOUND=0,20,SOUND=5,0
17 FOR I=0 TO 120
18 BOX(T0,160-I*RR,100-I*RR,160+I*RR,100+I*RR
SOUND=0,10-SOUND=12,100-I*RR,100+I*RR
00-10*3)=NEXTI
19 SOUND 19,14,PAI3,5,SYMBOL(30,0,0, ENF1
N",5,6,SYMBOL(50,100, LIBRE 2,5,6,CURSO
R0,20,END
20 INPUT @TAILEQ,TA,BDIM,TATA,TA,10,INIT
"CRT,M"
21 X=A-1,Y=A-1,S=3,GOSUB18-BOX210,0,TAX4+3,
TAX4+3,2=GOSUB25,X=A-1,Y=A-1,GOSUB25,Y=A-1,
X=A-1,GOSUB25,Y=A-1,X=A-1,S=2=GOSUB12
22 GOSUB25
23 PAINT(10,4,4,2,IF POINT (TATA-1)*4,TATA
-1)*4)K>1,PRINT JE FAIS GREVE!END
24 RETURN
25 FL=GOSUB34,MUSIC A0,4,PORT=1,TOTA*3
26 IF T0X=X,A,TATA,00 THEN 29
27 P=XX0-T0D-10,IF T0G,P,0,0 OR T0D,P
-P,0,0 THEN 31
28 IF T0M,P+*ST,0,0 THEN 31,ELSE GOTO 31
29 X=IX+XA,TAY+YA,GOSUB34
30 IF RND12K,6,THEN 26
31 S=S+1,IF PND0,15,S,THEN S=S+2
32 ON S MOD 4,*1,GOSUB 12,13,14,15,NEXT I
33 RETURN
34 BOX (XX4,TX4,YY4+3,TX4+3,3,TX,Y,01+1
TCY,X,1,D=1,RET,JRN

```

# 19 Micros-poches Duriez qui décrochent les exam's à prix charter

**A**PRES TESTS, Duriez vous les recommande.

**1** TI 74 de Texas Instruments : 1190 F. **Un tour de force !** Micro-poche basic de pointe, synthèse de calculatrice scientifique et de micro-ordinateur aux performances pro (math, stat, langage Pascal).

Outil puissant pour ingénieurs hommes de stats, de finances, étudiants.

**2** Les Grandes Classiques, très connues, très répandues, polyvalentes : math. stat.

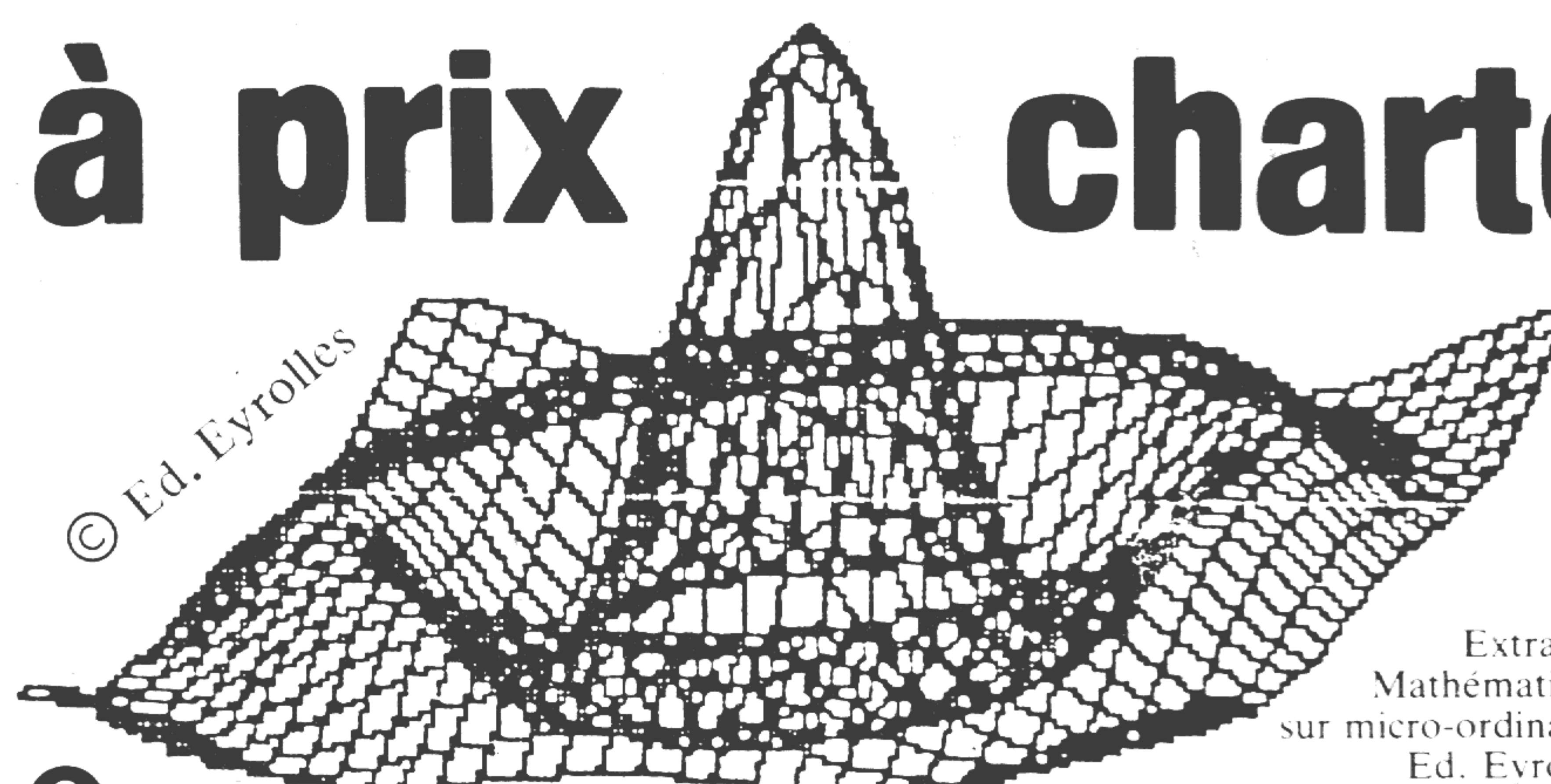
■ 2 programmables langage élémentaire Hewlett-Packard • HP 11C : 471 F • HP 15C : 915 F.

■ 2 calculatrices scientifiques programmables basic

• Sharp PC 1401 : 790 F •

Sharp PC 1402 : 990 F.

**Chez Duriez :**  
**500 Prix charter**



Extrait de  
Mathématiques  
sur micro-ordinateur  
Ed. Eyrolles.

### 3 5 financières programmables.

Chaque fois que vous placez ou retirez de l'argent celui-ci vaut plus ou moins selon les taux d'intérêt et d'inflation. Donc, votre comptabilité est fausse ?

Pas du tout ! Ces micros calculent votre rentabilité interne réelle avec valeurs présente et future annuités, cash flow, nombre de périodes, etc...

• HP 12C de Hewlett-Packard : 915 F.

• Sharp 1421 : 1175 F • Sharp 533 : 440 F • TI BA54 : 390 F.

• Et la nouvelle Star financière HP 18C : 1800 F ttc.

### 4 En première mondiale :

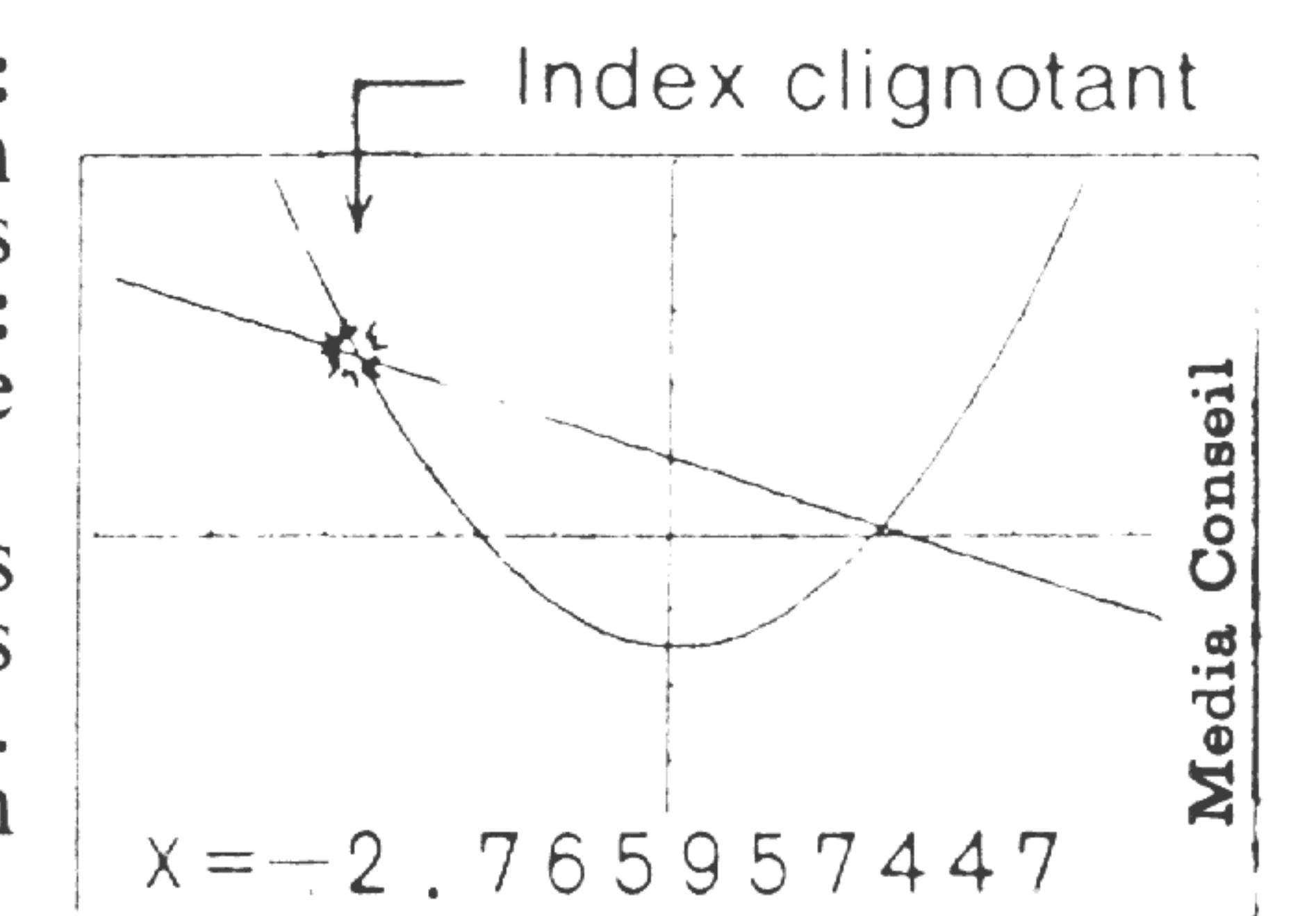
Des micros scientifique à écran graphique avec 422 et 486 pas de programmes fx-6.000G : 690 F - fx-7.000G : 869 F de Casio.

Entrez les équations : les courbes se dessinent sous vos yeux, superposées ou non. Erreurs faciles à réparer. Zoom à volonté ici ou là.

• Demandez notice Duriez : Micro-News Sicob 1986 contre 2 timbres à 2,20 F.

Duriez 132, Bd St-Germain, Paris 6<sup>e</sup>, M<sup>°</sup> Odéon. Ouvert Mardi au Samedi de 9 h 45 à 19 heures.

**5 La mémoire d'éléphant •**  
Sharp PC 1600 : 3 390 F ttc : 40 pages d'anti-sèches intégrables (chut !) 80 Ko de mémoire vive, le record.  
• Branchable disquettes et imprimante.



• Catalogue général (mach. écr., répondeurs, téléph., dictateurs et tout le Bureau) 3 timbres 2,20 F • Gratuits en magasin.

M \_\_\_\_\_  
ADRESSE \_\_\_\_\_

vous prie de bien vouloir lui faire parvenir  
 Recueil(s) des n° 1 à 5 des bulletins SHARPENTIER  
 au prix de 100 F\*  
 Bulletin(s) SHARPENTIER n° 6  7  8  9  10   
 au prix de 20 F chacun\*  
 Bulletin(s) SHARPENTIER n° 11  12  13   
 au prix de 25 F chacun\*  
 Bulletin(s) SHARPENTIER n° 14  15  16  17 épuisé, 18  19   
 20 au prix de 30 F chacun\*  
 Manuel Langage Machine, PC 1251   
 au prix de 180 F chacun\*  
 Manuel de Référence MZ 800 au prix de 240 F l'un\*

\* Port forfaitaire : **20 F**  
 Ci-joint un chèque de ..... F à l'ordre du  
**« CLUB DES SHARPENTIERS »**  
 Date \_\_\_\_\_ Signature \_\_\_\_\_

## SOMMAIRE

Édito	2
Initiation LH-5801 III	3
Ajustement de courbe	3
Astuces 1 et 2	5
Compte bancaire	6
Start Mogura	7
Edit LM	7
ON	8
Graphe ligne	9
Des PC sur le terrain	10
Format scientifique	11
Karaté K	12
Vocable	13
Conversion de température	14
Water panic	14
Codes assembleurs	16
News	17
Le port d'entrée analogique	18
Initiation au LM ESR-H (7 <sup>e</sup> )	18
Musique et acoustique	19
Des erreurs...	20
Gesmail	21
Astuces	22
Graphitik	22
3D LABY	23
Astuces	23