

L'Ordinateur de poche

Trimestriel N°4 - 12 FF

ISSN 0291 - 5243

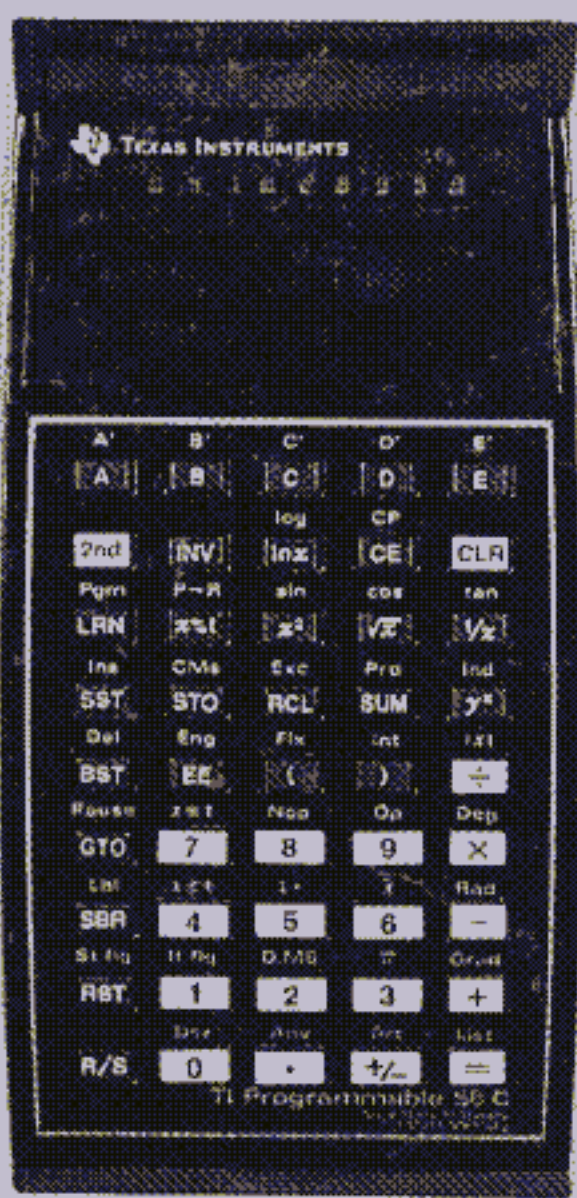
**NOUVEAU
CASIO FX 702 P**



Et toujours
du **BASIC**
des chiffres
des lettres
et des jeux

M 1859-04 - 12 F

Belgique 90 FF - Canada 2,95 \$ - Suisse 5 FF



TEXAS-INSTR TI 58C

- 10 chiffres rouges • Affichage LED • 60 mémoires ou 480 pas de programmes permanents • Module interchangeable préprogrammé de 5 000 pas environ contenant 25 programmes divers • Trigo • Log • Moyenne, Ecart-type • Régression linéaire • Corrélation • En option : 5 modules préprogrammés et une imprimante • Alimentation par batteries rechargeables et secteur • Autonomie 3 heures.

164 x 80 x 37 mm

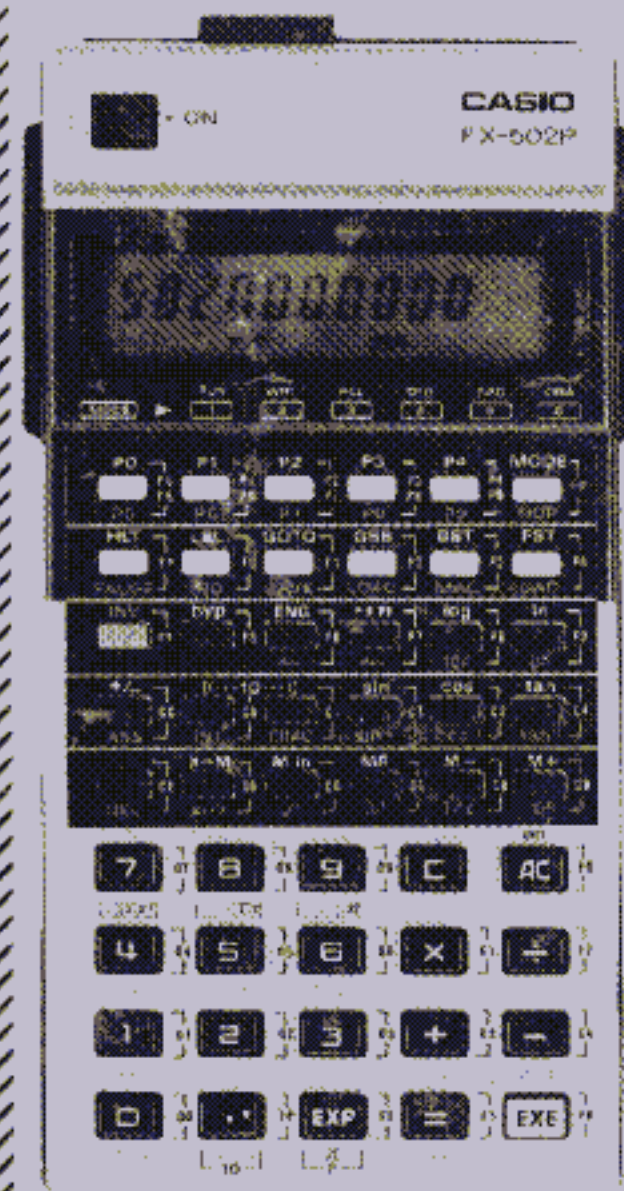
750 F ttc

Performances/Prix Très bonnes

Qualité : Bonne

Adaptable toutes professions par modules et mémoire permanente.

La Polytechnicienne



CASIO FX 502 P

- 10 chiffres noirs sur fond jaune • Affichage LCD • 22 mémoires permanentes • Notation prioritaire • Trigo • Log • n! • Moyenne, Ecart-type • 256 pas combinés de programmes permanents • Option interface pour magnétophone à cassettes • Autonomie jusqu'à 1 300 heures.

144 x 75 x 15 mm

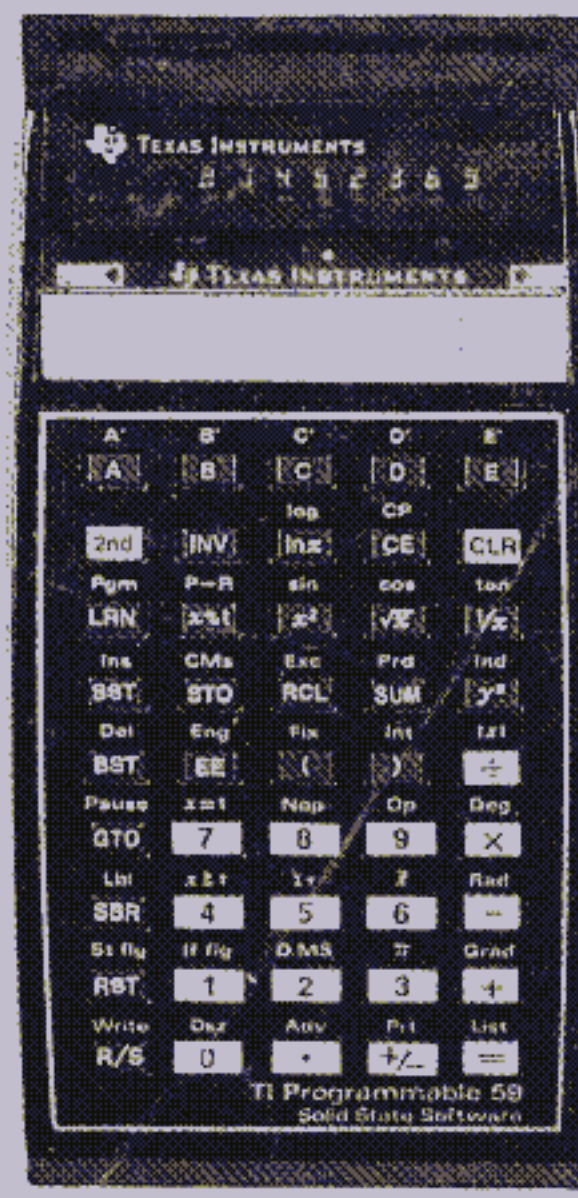
544 F ttc

Interf magnéto : 189 F Performances/Prix Bonnes

Qualité : Bonne

une programmable puissante réellement de poche.

La moins chère magnétophonable



TEXAS-INSTR TI 59

- Mêmes caractéristiques que la TI 58, sauf les suivantes • Capacité de programmation 960 pas ou 100 mémoires et 160 pas • Lecteur enregistreur de cartes magnétiques pour l'enregistrement des mémoires et des programmes • Possibilité d'écriture alphanumérique et de tracé de courbes par points avec le bloc imprimant en option.

164 x 80 x 37 mm

1350 F ttc

Performances/Prix Bonnes Qualité : Bonne

De très loin la moins chère à cartes.

La Super-Mémoire



Casio 702P

- Micro-ordinateur de poche • Langage Basic • Très grande rapidité de calcul • De 1680 pas + 26 mémoires à 80 pas + 226 mémoires • Nombreuses fonctions au clavier, dont Trigo, Log, Stat, régressions, corrélations • Capacité 10 chiffres • Affichage 20 caractères.

Prix : 1320 F ttc.

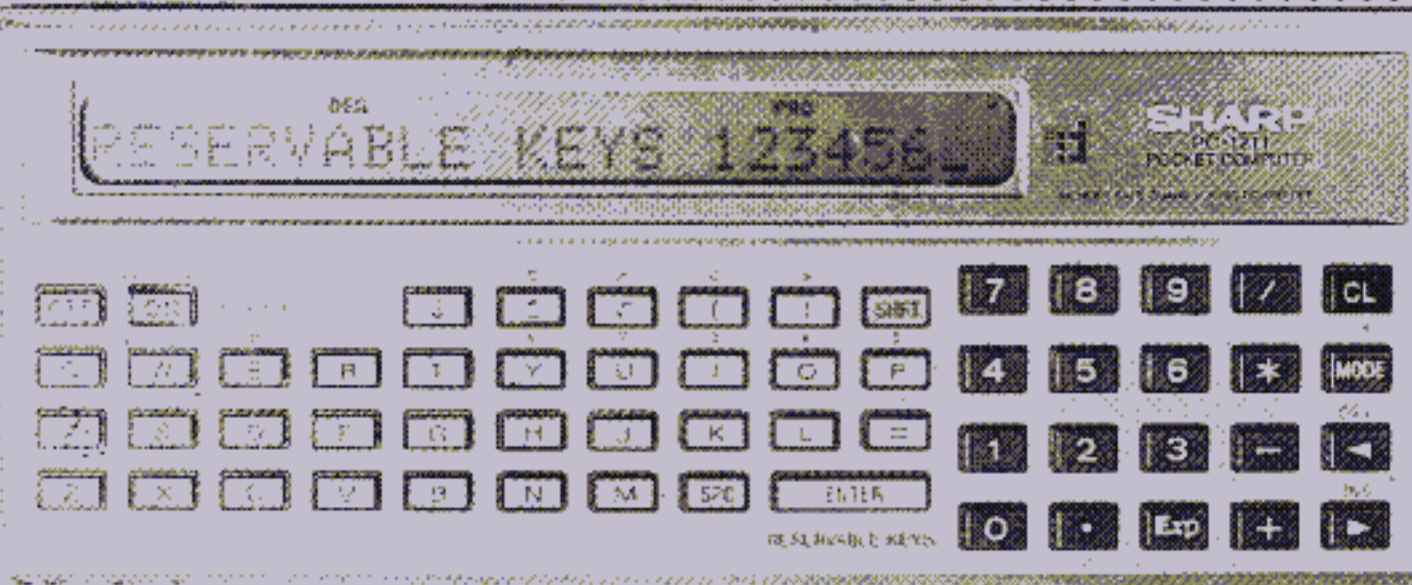
82 x 165 x 17 mm

Performances/Prix :

Très bonnes. Qualité : Bonne Beaucoup de fonctions au clavier avec la programmation en basic.

La Fascinante

- Inter face magnétophone prix Duriez 210 F ttc
- Inter face imprim 550 F ttc
- Option ultérieure : Extension mémoire programmable.



SHARP PC 1211

- Micro-ordinateur de poche • Affichage LCD 24 caractères alphanumériques noirs sur fond jaune • Capacité 10 chiffres • Langage Basic • 1 424 pas de progr. permanents (ou 178 mémoires) + 26 mémoires indépendantes permanentes • Mini clavier machine à écrire • Option interface pour magnétophone • Etui plastique rigide • Autonomie jusqu'à 300 h. • Manuels d'utilisation, de Basic, d'applications (79 programmes divers).

71 x 177 x 17

Promotion Noël

1090 F ttc

Imprimante : 850 F ttc

Performances/Prix

Très bonnes

Qualité : Bonne

idéal pour apprendre le Basic et très performante pour sa taille.

Le Basic en Poche

////////////////////



HEWLETT-PACKARD 34 C

- 10 chiffres rouges • Affichage LED • Notation polonaise inverse • Mémoire permanente • Mémoire dynamique - configuration de base : 70 lignes de programmes et 21 registres de données ; possibilité d'avoir jusqu'à 210 lignes de programme • Deux touches à définir A et B • 6 niveaux de sous-programmes • Adressage indirect et relatif • Fonction SOLVE : calcul des racines de plusieurs types de fonction • Fonction INTEGRATE : calcul de l'aire d'une courbe de fonction f(x) dans un intervalle spécifié, y compris maxima et minima.

142 x 75 x 30 mm

1070 F ttc

Performances/Prix Bonnes

Qualité : Bonne

permet la résolution directe de la plupart des calculs scientifiques.

La Super-Savante

50 Millions de Calculs au Banc d'Essai Duriez



HEWLETT-PACKARD 41C

- Affichage alphanumérique noir sur fond LCD gris • 12 caractères alphabétiques • 130 fonctions préprogrammées • Mémoire à 63 registres permanents de données (1 registre = 7 lignes de programme ou 1 mémoire de données) • 6 niveaux de sous-programmes • Adressage indirect sur tous les registres • Configuration modulaire • Nombreux logiciels et livres d'applications • Autonomie jusqu'à 1 000 heures.

144 x 79 x 33 mm

1780 F ttc

Performances/Prix Bonnes

Qualité : Très bonne

remarquable par ses possibilités d'extensions.

La Surpuissante

Extensions de la HP 41C :
I. Jusqu'à 4 modules de mémoires programmables supplémentaires, comportant chacun 64 registres • Supplément au prix de base 237 F ttc

II. Nombreux modules préprogrammables • Mathématiques • Statistiques • Finances, etc. • 237 F ttc sauf except.

III. Lecteur enregistreur de cartes magnétiques. Les cartes enregistrées pour le modèle HP 67 et 97 sont

IMPARTIAL, Duriez a testé toutes les calculatrices possibles : les marques établies - quelquefois surfaites ou dépassées - et les jeunes qui percent en force.

Duriez n'est pas chargé de soutenir telle ou telle marque. Il vous aide à voir clair et acheter sûr. Il comprime les prix. Si vous n'êtes pas satisfait sous 8 jours, il vous rembourse.

Parmi les 38 modèles du Palmarès Duriez (voir catalogue-banc d'essai, gratuit), en voici 7. Dignes de Duriez.

Duriez, fondé en 1783 (Nombre Premier), 132, Bd St-Germain, 6^e, M^o et RER Odéon - St-Michel - Luxg. 9 h à 19 h du Ma. au Sa.

Prix ttc jusqu'au 31.12.1981

Tous modèles de cette page garantis 1 an.

Toutes bibliothèques et accessoires en stock.

compatibles, ce qui permet d'utiliser les bibliothèques et fascicules de programmes existants pour ce modèle

• Prix : 1450 F ttc

IV. Imprimante thermique alphanumérique permettant le tracé de courbe par points

• Prix : 2500 F ttc



V. Lecteur optique de code-barres pour introduction rapide des programmes • Prix : 940 F ttc

HEWLETT-PACKARD 41CV

Mêmes caractéristiques que la 41C, sauf 319 registres. 2 390 F ttc

commandez / Duriez par poste

Joindre règlement à commande. Expédition sans frais. Garantie 1 an, pièces et main-d'œuvre. Satisfait ss 8 jours ou remboursé. • Duriez 132, Bd St Germain, 6^e. M^o Odéon.

1 COUVERTURE

Avec Annie Goetzinger à qui l'on doit la couverture de ce numéro, l'OP vous souhaite de joyeuses fêtes et une heureuse année 1982.

5 ÉDITORIAL

6 A VOS CLAVIERS

8 MAGAZINE

16 LA TI 57 AURAIT-ELLE LA MÉMOIRE LONGUE ?

Une application utile qui nous vient directement des « dessous de la TI 57 » : comment éteindre et rallumer l'affichage seul.

18 DES ORDINATEURS DE POCHE POUR...

toutes sortes d'utilisations dans le cadre et à l'extérieur d'un C.H.U.

20 UNE INSTRUCTION MÉCONNUE DU PC 1211/TRS 80 P :

Clod 1 expliquée en détails.

22 LE DESTIN DES NOMBRES

Dans un micropoche, les nombres subissent des traitements variés : binaire, hexadécimal, etc.

26 LE BASIC NE MANQUE PAS DE CARACTÈRES

Quatre ordinateurs et les chaînes alphanumériques.

29 UNE MONTRE ÉLECTRONIQUE

avec ce court programme pour HP 41 C qui ne compte qu'une chose : le temps qui passe.

31 LES MICROPOCHES EN CLASSE

Multiplications et divisions : quand les écoliers apprennent les opérations à leur TI 57.

34 NOUVEAU : LE CASIO FX 702 P

Nous avons essayé ce nouveau poquette BASIC : nos premières impressions.

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part que « les copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective », et, d'autre part, que les analyses et les courtes citations dans un but d'exemples et d'illustrations, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants-droit ou ayants-cause est illicite » (alinéa 1^{er} de l'Art. 40). Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contre-façon sanctionnée par les Art. 425 et suivants du Code Pénal.

L'Ordinateur de poche

n° 4

12 F - Trimestriel

RÉDACTION-RÉALISATION

Rédacteur en chef : Bernard Savonet
Rédacteurs : Jean Baptiste Comiti, Jean-Pierre Brunerie
Assistante de rédaction : Michelle Aubry
Maquette : Eliane Gueylard

Ont participé à ce numéro : Jean Biron, Damien Bommart, Luc Bureller, Pierre Courbier, Olivier Dabée, Jean-Jacques Dhénin, Marc Ferrant, André Flédric, Antoine Jenet, Claude Lacour, Xavier de La Tullaye, Pascal Leclerc, François Martin, Jean-Michel Muratet, Jean-Pierre Richard, Jean Simoni, Jean Thiberge, Dominique Thiébaud, André Turlure, Antoine Vaussy-Lesbaudy.

Iconographie : Eric Berthier, Serge Fayol, Alain Mangin, Alain Mirial, Fabrice Péray, Nicolas Spinga.

ÉDITION-PUBLICITÉ

Éditeur : Jean-Pierre Nizard
Assistante d'édition : Maryse Marti

Rédaction-vente-publicité : 41, rue de la Grange aux Belles, 75483 PARIS CEDEX 10.
Téléphone : (1) 238 66 10 - Télex : 230 589 EDITEST.

Abonnement voir page 11

L'Ordinateur de poche
est une publication du *groupe tests*
Directeur de la publication : Jean-Luc Verhoye

39 LES CHIFFRES DE GARDE

Un petit programme pour connaître exactement tous les chiffres d'un résultat tel que l'a calculé la TI 57.

40 LES ÉQUATIONS DU 2^e DEGRÉ

Comment en trouver facilement les racines : un sujet d'examen.

42 LES MICROBES S'INSTALLENT DANS LA FX 502 P

Le Jeu de la Vie adapté à ce micropoche.

43 TROUVEREZ-VOUS LE BON MOT ?

Pour le poquette de Sharp et de Tandy, une espèce de mastermind littéraire...

47 VOUS VOULEZ DES PROGRAMMES ? SERVEZ-VOUS

La bibliothèque de base des TI 58/59 contient plus de 5 000 pas de programmes tout faits.

49 UN AUTRE JEU DE MOTS, MAIS SUR HP 41

Découvrez en vous distrayant quelques-unes des possibilités alphanumériques de votre machine.

56 BRICOLAGE ET PLASTIQUE

Construisez vous-même les petits accessoires dont vous avez envie.

58 L'ALPHABET RÉDUIT DE LA TI 57

Une suite inattendue aux « dessous ».

59 LES SAUTS : UNE QUESTION D'ADRESSE

Sur les TI 58 et 59, comment choisir entre les deux modes d'adressage ?

61 UN POT COMMUN POUR TOUTES LES MACHINES

Conversion d'unités biologiques pour HP 41 et PC 1211 ; maximum d'une fonction et Kibur pour TI 58/59.

66 UN ANCÊTRE

La HP 25 aura bientôt 7 ans. Celle que nous avons vue marche encore. Honneur aux anciens !

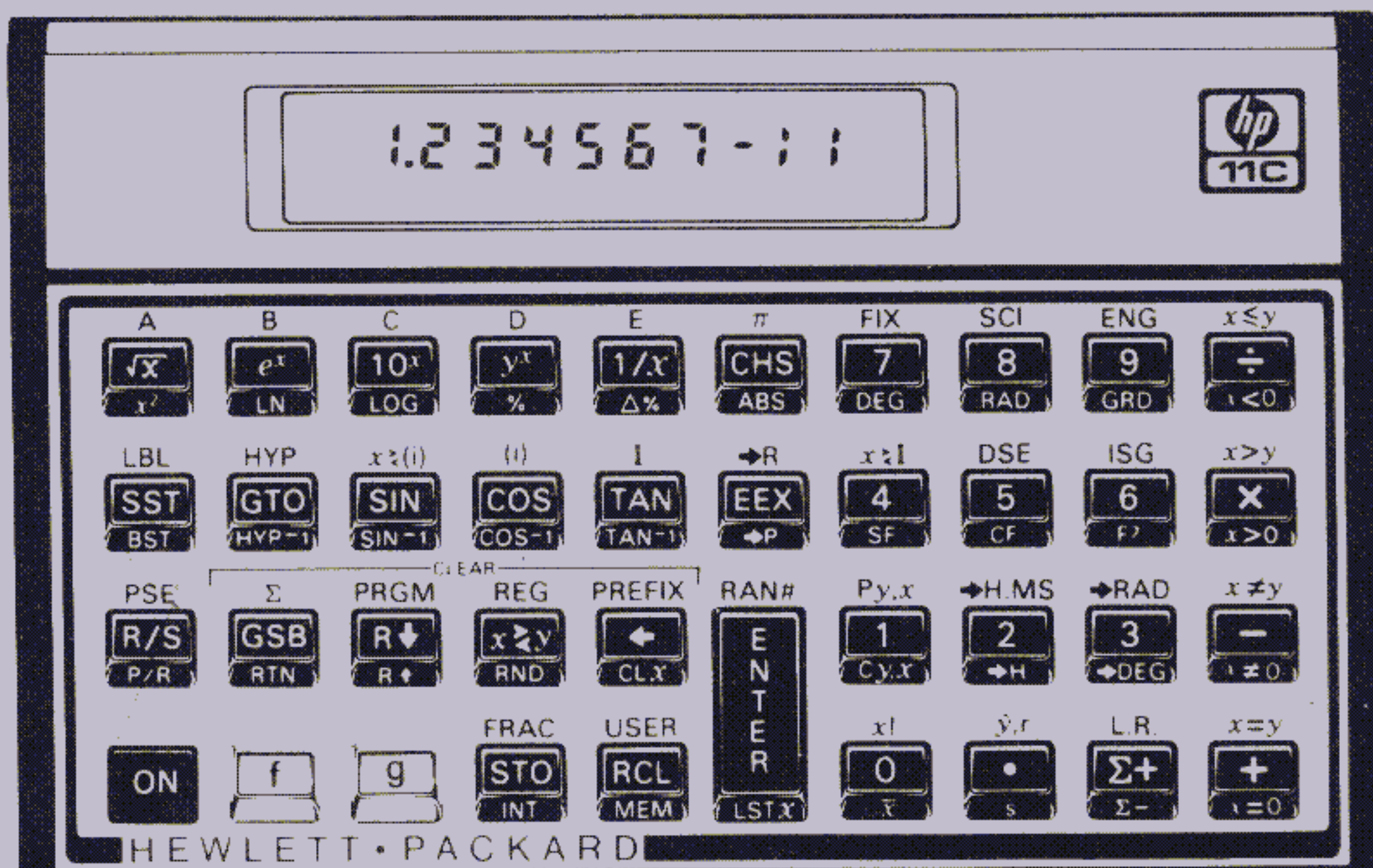


Notre publication contrôle les publicités commerciales avant insertion pour qu'elles soient parfaitement loyales. Elle suit les Recommandations du Bureau de Vérification de la Publicité. Si, malgré ces précautions, vous aviez une remarque à faire, vous nous rendriez service en écrivant au BVP, BP 116, 75722 PARIS CEDEX 15.



HEWLETT
PACKARD

CALCULS ACTUELS distribue les calculateurs HP-11C et HP-12C



Mémoire permanente - Affichage à cristaux liquides - Poids 112 g

HP-11C : calculateur scientifique programmable - 106 fonctions - 15 labels, jusqu'à 203 lignes de programme - 21 registres mémoire.

HP-12C : calculateur financier programmable - 70 fonctions - jusqu'à 99 lignes de programme - 20 registres mémoire.

CALCULS ACTUELS distribue également toute la gamme des calculateurs de poche Hewlett-Packard et le système de calcul personnel, simple et évolutif HP-41C/CV.

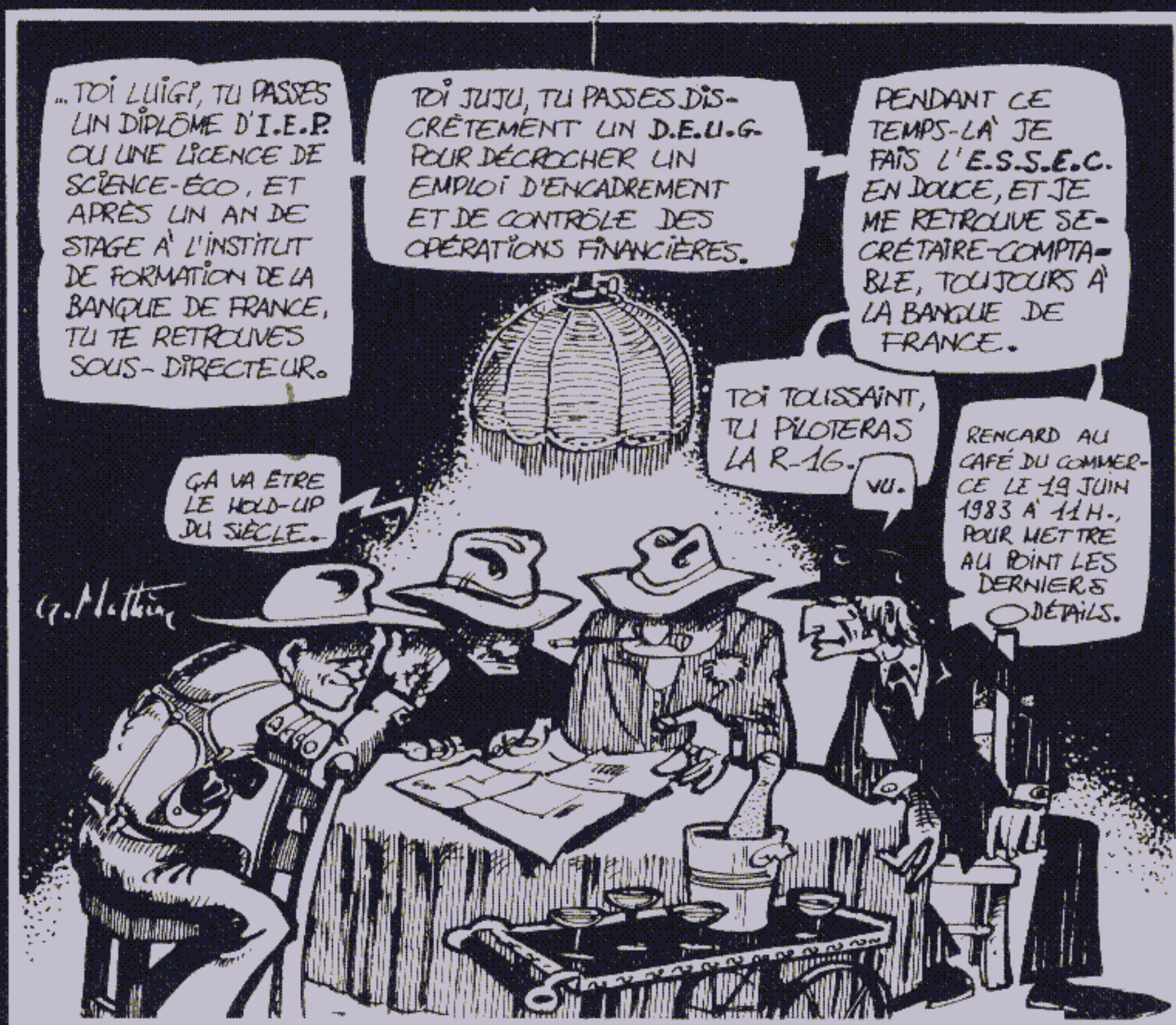
CALCULS ACTUELS

informatique personnelle

49, rue Paradis - 13006 MARSEILLE
Tél. : (91) 33.33.44

L'Etudiant

LES GRANDES ECOLES



300 GRANDES ECOLES

30.000 BONNES PLACES A PRENDRE

TOUS LES TUYAUX

POUR PRENDRE LA VOTRE.

EN VENTE PARTOUT 18 F.

Les micropoches à la mode

Les ordinateurs de poche font beaucoup parler d'eux en ce moment. Sans doute est-ce dû aux récentes déclarations de François Mitterrand, à la suite d'un rapport que lui avait adressé Jean-Jacques Servan-Schreiber : une dizaine de spécialistes du monde entier ont accepté de collaborer à un centre mondial chargé d'étudier ce que la société peut et doit attendre de l'informatique, et notamment des ordinateurs individuels de table et de poche.

Ce centre mondial sera opérationnel dès le début de 1982, et il est question d'y définir un ordinateur ayant " la taille et le poids d'un livre ", bref un nouveau type d'ordinateurs de poche. La réunion de tels spécialistes ayant accepté de travailler ensemble, les sujets de leurs études, les matériels qu'ils vont imaginer, tout cela nous semble revêtir pour *l'Ordinateur de poche* et ses lecteurs une importance capitale. C'est pourquoi nous attendons les résultats de ce projet avec une espérance impatiente : pourvu qu'elle ne soit pas déçue !

Bientôt des micropoches avec écran plat de " grandes " dimensions, avec graphiques, couleurs, et le reste ? Oui, nous pensons depuis longtemps qu'ils finiront par arriver. Avoir " des lettres ", en attendant d'avoir " des dessins ", voilà la caractéristique actuellement la plus marquante des ordinateurs de poche récents. Aussi nous sommes-nous attachés dans ce numéro à donner des lettres à vos micropoches, plus exactement à utiliser les possibilités alphanumériques qu'ils possèdent d'origine et celles que l'on obtient en les utilisant avec un peu d'astuce.

□ Jean-Baptiste Comiti

A vos claviers

Nul n'est censé ignorer la loi

Il y a une question que je me pose et dont la réponse intéressera de nombreux lecteurs. J'ai entendu dire qu'une nouvelle loi était sortie concernant les calculatrices au baccalauréat. Si c'est le cas, quelles restrictions apporte-t-elle ?

Jean-Christophe Allué
Paris 14^e

■ *A notre connaissance, aucun texte n'est venu modifier les règlements qui délimitent l'usage des calculatrices aux examens et aux concours. Nous n'avons pas non plus d'informations selon lesquelles un changement en ce domaine serait sur le point de se produire. La circulaire du Ministère de l'Éducation datée du 20 octobre 1979 (voir l'article d'André Warusfel pages 16 et 17 de l'Op n° 1) est donc toujours en vigueur.*

Une petite pièce de rechange

Sur le TRS 80 pocket, le couvercle qui protège la prise de connexion du micropoche à son interface cassette ou imprimante n'est pas bien grand. On a tôt fait de l'égarer. C'est ce qui m'était arrivé.

Or j'ai eu la surprise de découvrir l'autre jour sur l'imprimante, en bas et à droite du logement réservé au poquette, le frère jumeau de ce petit morceau de plastique. Il est si discret que je ne l'avais pas remarqué. Je l'ai donc récupéré pour remplacer celui que j'avais perdu et j'ai obstrué la petite fenêtre ainsi créée dans l'inter-

face au moyen d'un morceau de ruban adhésif. Je ne pense pas qu'il y ait d'inconvénient à faire ainsi.

Voilà qui pourrait être utile à ceux qui ont, eux aussi, perdu leur couvercle...

Jean Thioulouse
69 Lyon

■ *La même remarque vaut, nous l'avons vérifié, pour le PC 1211 et son imprimante CE 122.*

TI 57 C ?

Dans le numéro 2 de l'Ordinateur de poche, on peut lire à la page 63 dans le chapitre « les dessous de la TI 57 » cette proposition : « on peut envisager un petit bricolage donnant presque à la TI 57 une mémoire continue ! »

C'est pour connaître ce petit bricolage que je vous écris. Comment éteindre l'affichage tout en conservant un programme ?

Philippe Durand
11 Narbonne

■ *Plusieurs lecteurs nous ont indiqué des astuces qui permettent d'obtenir simplement cette extinction de l'affichage. Quand on maintient la touche R/S enfoncée, la machine n'affiche plus rien, mais le contenu des mémoires demeure intact. Pour y parvenir — en gardant les mains libres — la méthode la plus commode consiste sans doute à utiliser un petit cube découpé dans une gomme qui presse cette touche et la maintient enfoncée grâce à un élastique fortement tendu.*

Cela dit, vous trouverez également dans le présent numéro (voir la suite des « Dessous de la TI 57 ») comment obtenir un petit interrupteur purement logiciel de l'affichage.

Ad vitam æternam...

En ce qui concerne les abonnements « à vie » gagnés, s'agit-il de la vie de la revue ou de celle de l'abonné ? (Ne le prenez pas en mauvaise part).

Je vous joins un abonnement (payant) en vous souhaitant personnellement et journalistiquement bonne et longue vie.

Toute ma sympathie.

Daniel Connan
93 Pantin

■ *Il s'agit bien entendu de la vie des abonnés... aussi longtemps que la revue existera. Ces abonnements ne sont donc pas près « d'expirer ». Nous le souhaitons de tout cœur. Bonne et longue vie à tous !*

A la cantonade...

La Casio FX 502 P m'a enthousiasmé

- par son encombrement : je la balade partout sauf en planche à voile !
- par son autonomie : au bout d'un an, je n'ai jamais changé les piles et je m'en sers tous les jours ;
- la frappe des touches est un plaisir ;
- l'utilisation ne comporte pas de complications inutiles et les possibilités de calcul sont très correctes eu égard au prix ;
- enfin il est possible de stocker le programme sur cassette.

A noter cependant qu'il y a un revers à cette belle médaille. La bibliothèque des programmes est en allemand, ce qui est gênant car il y a peu d'informaticiens germanophones. D'autre part, la notice est obscure pour un non-spécialiste.

Enfin et surtout, il se révèle finalement impossible de stocker ses programmes sur cassettes, ou plus exactement de les recharger dans sa machine par l'intermédiaire d'un magnétophone ordinaire (en l'occurrence le mien).

Cela revient à dire que, possédant une chaîne HiFi, un « walkman » enregistreur et un magnétophone Philips mono, il faudrait pour bien faire que j'achète un quatrième magnétophone spécial pour ma calculatrice avec impédance correspondant à l'interface FA 1. Ça me semble un peu fort.

De ce fait, je vous serais reconnaissant, à vous ou à vos lecteurs, de m'indiquer comment je pourrais réaliser ou acheter un correcteur d'impédance à intercaler entre la sortie du magnétophone et l'interface.

Merci d'avance dans cette éventualité et à bientôt, au n° 4 de l'Op.

Eric Larcher
69 Gleizé

■ *Peut-être un lecteur a-t-il la réponse à votre question ? Nous nous ferons un plaisir de vous la transmettre.*

Les bonnes idées

J'aimerais pouvoir lire dans vos colonnes des programmes écrits par des lecteurs. Cela est-il possible ?

Philippe Lebarz
69 Lyon

De nombreux lecteurs ont certainement plein d'idées. Pourquoi ne pas créer une page pour diffuser les idées (annonces, lettres publiées avec les trucs découverts pour toutes les machines) dans chaque numéro ?

Hervé de Saboulin
49 Mûrs Erigné

■ En réalité, ce n'est pas une page de l'Op qui est consacrée à la diffusion de ces idées, mais plusieurs dizaines : une bonne partie du journal émane en effet des lecteurs. Plus que jamais, à vos claviers !

D'un lecteur (de cartes) à l'autre...

Vous avez publié dans votre numéro 3 un jeu pour HP 41 C que je ne possède pas. Par contre, je dispose d'une TI 59 que j'utilise depuis plusieurs années sans aucun ennui de fonctionnement. J'ai donc essayé de transposer votre programme pour le passer sur ma machine, et je crois y être parvenu en suivant l'idée générale de l'organigramme.

Je me permets de vous transmettre ce programme établi assez rapidement ; il peut être encore amélioré, mais il fonctionne correctement. J'avais pensé vous l'adresser sous la forme d'une bande magnétique enregistrée, mais je me suis souvenu que Texas ne garantissait pas qu'une bande enregistrée sur une TI 59 pouvait être lue par une autre TI. Je vous l'envoie donc manuscrit, ne possédant pas d'imprimante.

Jean Simoni
95 La Varenne

■ Merci de nous avoir communiqué votre programme que nous allons essayer après l'avoir « entré » manuellement dans la TI 59 de la rédaction.

Au sujet des cartes magnétiques, et malgré l'avertissement de Texas Instruments, l'expérience montre qu'elles permettent, le plus souvent, de transmettre programmes et données d'une machine à l'autre. Il arrive parfois qu'une TI 59 refuse de lire les cartes qui n'ont pas été enregistrées par elle, mais cela paraît rare. On a donc tout intérêt à envoyer ses programmes non seulement sous forme d'une liste (en utilisant l'imprimante, lorsque cela est possible, on élimine les risques d'erreurs de recopie), mais encore sous forme de carte

magnétique : le gain de temps est important, et les risques d'erreurs sont, là aussi, réduits.

Il en va de même, bien entendu, pour tous les micropoches qui sont équipés d'imprimantes ou de dispositifs d'enregistrement magnétique.

Tarif aérien oblige

Je voudrais me renseigner sur les conditions et les tarifs d'abonnement pour l'Outre-Mer, et spécialement la Guadeloupe.

Moïse Bilonère
97 Guadeloupe

■ Le tarif d'abonnement pour un an (4 numéros) qui s'applique aux lecteurs de Guadeloupe est de 85 FF, somme qui comprend les frais d'envoi par la poste aérienne.

Vers un sésame nouvelle manière ?

A propos du PC-1211, essayez ceci : saturez la mémoire de réservation et enregistrez-la sur une cassette, puis passez en mode PRO et chargez le programme de réservation. Essayez alors de le lister ou d'en modifier des lignes. Vous verrez qu'il se passe de drôles de choses.

Yann Gainche
45 Fleury les Aubrais

Je crois pouvoir vous annoncer qu'il est toujours possible de dévoiler la face cachée du PC-1211, quoique désormais ce soit un peu plus délicat. Il faut charger depuis le magnétophone un programme de réserve en mode PRO. En listant ce programme, on entendra des bruits bizarres. De temps en temps, cela plante tout, mais en appuyant sur ON/Break, on reprend parfois la main, sinon il ne reste plus que la solution "All reset" avant de recommencer à zéro. Le reste du temps, miracle ! on

Alea jacta est : Dix abonnés à perpétuité...

■ Chose promise, chose due. Nous avons procédé au tirage au sort des 10 abonnés à vie. La main innocente a officié pendant le SICOB 81, et voici la liste de ceux et de celles que le sort a désignés :

Daniel Benezra	93 Saint-Denis
Dominique Buteau	92 Asnières
Didier Celse	95 Cergy
Patrick Henot	57 Grostenquin
Hélène Hugot	30 Nîmes
Véronique Levec	29 Brest
Patrick Mellul	94 Créteil
Gérard Pointurier	39 Orchamps
Jean Vilte	77 Provins
François Youssef	95 Taverny

A l'avenir, ces dix gagnants recevront donc gratuitement tous les numéros de l'Ordinateur de poche. Charge à eux seulement de nous informer, lorsque cela se produira, de leurs changements d'adresse.

l'Op

se retrouve en " langage machine ".

Luc Bureller
Paris 17^e

■ Inutile de dire que la procédure que vous décrivez intéressera de nombreux utilisateurs du poquette de Sharp et de Tandy. Si l'on parvient à obtenir à partir de cette idée un sésame qui fonctionne à coup sûr et qui soit suffisamment simple d'emploi, chacun pourra — même sur les machines de fabrication récente — explorer la face cachée de l'ordinateur.

Cette exploration n'a pas pour seul mérite de faire découvrir des bizarreries, elle devrait également permettre de petites applications. N'hésitez pas à nous écrire pour nous communiquer ce que vous aurez découvert.

Sur son 31

Dans le programme du Tiercé-Loto que vous avez publié page 28 de l'Op n° 3, le pas 150 correspond à l'instruction 31. Or celle-ci (touche LRN) ne peut rentrer en mode programmation ni par LRN — qui ferme la programmation —, ni par 31 qui s'inscrit sur deux pas.

Comment faire ?

Martine Vaucelle
35 Lécousse

■ En réalité, le code 31 qui est listé au pas 150 ne représente pas une instruction, mais seulement les deux dernières décimales de l'adresse (pas 131) à laquelle doit se poursuivre le programme tant que la mémoire 7 n'est pas égale à zéro (dsz 7 des pas 147 et 148). Pour obtenir 31 en ligne 150, il suffit de taper 147 : 2nd Dsz, 148 : 7 (et non pas 07), puis 131 (pas 149 et 150).

Jouons avec la partition

Il y a une chose que je ne comprends pas dans les numéros 2 et 3 de votre revue. En effet, il y a des programmes qui contiennent 294 pas et plus et qui sont proposés pour TI 58. Or ma machine, une TI 58, ne contient que 239 pas (240 avec le pas 000). Alors comment dois-je faire pour entrer ces programmes qui débordent ? Il faut dire que je suis très novice en la matière.

Joël Duquesnoy
59 Roubaix

■ Sur les TI 58 et 59, la mémoire est flexible : on peut la répartir comme l'on veut entre registres de données et mémoire du programme. A l'allumage, la TI 58 est toujours en parti-

tion 239.29, ce qui vous donne bien 240 pas (de 000 à 239) et 30 mémoires (de 00 à 29).

A tout moment, vous pouvez vérifier cette partition en effectuant 2nd Op 16 : l'affichage l'inscrit alors en clair, et vous pouvez aussi la modifier en indiquant combien de dizaines de registres de données vous souhaitez.

En demandant 2 2nd Op 17, vous obtiendrez 20 mémoires et 320 pas. En effectuant 0 2nd 17, vous obtiendrez 480 pas... Cela dit, avant de changer la partition, on doit s'assurer que la calculatrice n'est ni en virgule fixe, ni en notation scientifique ou ingénieur.

Et le ZX 81 ?

J'ai 12 ans et j'ai découvert l'Op à son troisième numéro ; cette revue m'a bien plu car j'y ai trouvé beaucoup de programmes et d'articles qui m'ont intéressé.

J'aurais bien aimé m'abonner pour l'année 82, mais ce qui me freine un peu, c'est que j'utilise depuis peu un Sinclair ZX 81 et que je n'y ai pas trouvé grand'chose concernant cet appareil, si ce n'est votre article de la page 13 qui m'a donné plus d'envie encore. Je ne suis certainement pas le seul à avoir le Sinclair et j'en conclus que si vous introduisiez des programmes et des articles de fond pour le ZX 81, vous feriez plus d'un heureux. Je vous en remercie d'avance.

Christian Koehle
Genève, Suisse

Pourquoi ne pas ouvrir vos colonnes au Sinclair ZX 81 dont le prix est inférieur à celui de beaucoup d'autres microproches ? En ce qui me concerne, je trimbale partout mon ZX dans la poche de mon porte-documents (cet ordinateur ne pèse que 350 grammes), et partout où je vais, je n'ai aucun mal à trou-

A vos claviers

Bogue corrigée est à moitié pardonnée

Dans notre précédent numéro, nous avons encore commis quelques faux pas... de programme. C'est ainsi qu'à la page 50, dans le jeu du 13, le pas n° 35 était tout simplement impossible à entrer (un comble : il correspondait précisément à l'instruction qui efface un pas !). La bonne instruction était INV EE, code -42 comme indiqué : le " 2nd " était de trop et gâchait tout.

A la page 25, dans le petit exemple de programme donné par Jean Thiberge, le deuxième pas, +, a disparu. Il convient donc de le remettre à sa place entre le chiffre 2 et le premier ENTER.

Dans la rubrique " A vos claviers ", il est répondu à un lecteur que la séquence INV 2nd C.t., sur TI 57, efface toutes les mémoires sauf la mémoire t. C'est faux. En réalité, elle efface toutes les mémoires sauf celle du programme.

Enfin deux erreurs se sont glissées dans la liste du Laser Oméga pour Casio FX 502 P (à quand une imprimante pour cette machine ?). Dans le P3, il faut insérer l'instruction INV Pause entre LBL 0 et INV x = F et dans le P4, il faut lire 1/x là où nous avons écrit yx. C'est tout. Ouf !

Beaucoup de nos lecteurs nous ont signalé un GSB P5 et l'absence de P5, mais cette absence-là est volontaire : l'appel de P5 déclenche l'affichage d'erreur qui vous signale que vous avez perdu.

ver un téléviseur où le raccorder.

Gérard Gourdain
Paris 20^e

■ C'est vrai, le ZX 81 n'est pas encombrant, il est léger et il est moins cher que les poquettes Sharp, Casio ou HP.

Ce petit ordinateur britannique possède des qualités qui lui sont propres. Et qui plus est, comme on le dit en trois mots (mais en anglais) : *small is beautiful...* (ce qui est petit est beau).

Certes, le ZX dépasse un peu de la poche d'une veste ou d'un pantalon, mais pas de celle d'un porte-documents. On peut ajouter qu'un poste de télévision d'un modèle courant ne tient pas dans un sac à main. Mais on peut prévoir que la taille minimale des récepteurs de télévision en vente d'ici un ou deux ans sera nettement plus réduite.

Prenons un exemple : à l'Op, nous utilisons quotidiennement, afin de sauvegarder certains programmes du poquette Sharp/Tandy et du Casio, un magnétophone

à micro-cassettes guère plus volumineux qu'un paquet de cigarettes... Et ça marche très bien.

Bref, nous nous demandons si nous ne devrions pas ouvrir au ZX 81 les colonnes de l'Ordinateur de poche. Arguments favorables : il possède un BASIC relativement complet ; ses dimensions ne sont pas énormes ; son prix le met à portée de la bourse de ceux de nos lecteurs qui veulent un « BASIC pas cher », au même titre que le Sharp/Tandy ou le Casio.

Arguments défavorables : ses dimensions dépassent un peu celles d'un véritable ordinateur de poche, tout au moins quand ce dernier ne possède pas beaucoup d'extensions ; son alimentation sur secteur lui « met un fil à la patte » un peu gênant pour sa portabilité.

Qu'en pensez-vous ? Vous avez tous votre mot à dire : estimez-vous que ce système doit acquérir droit de cité dans les colonnes de votre (notre) journal, au même titre que les systèmes qui y figurent déjà ?

Quelques touches introuvables

Utilisateur d'une TI 58 C, je trouve parfois dans des revues (et l'Op n'échappe pas à la règle) des programmes pour ma machine dont certains pas ne figurent sur aucune touche de mon micropoche. Ces instructions sont-elles obtenues grâce à la pression de plusieurs touches, ou s'agit-il d'erreurs de frappe ? Les voici : GE, LST, RTN, EQ, EE, STF, IFF, LNX, PAU, SM*, ST*, RC*, PD*, ADV, PRT. J'attends votre réponse avec impatience.

Vincent Gaillon
37 St Pierre des Corps

■ Vous n'êtes pas le seul à être déconcerté par ces codes qui sont en fait ceux de l'imprimante PC-100. Il est souvent difficile de deviner à quelles touches ils correspondent. C'est la raison pour laquelle vous trouverez dans le présent numéro, à la page 63, un petit tableau d'équivalence entre ces codes abrégés d'une part, et les séquences de touches auxquelles ils correspondent d'autre part.

Avis de recherche

Depuis un certain temps, je suis à la recherche d'un programme me permettant de fournir plusieurs dizaines, voire plusieurs centaines de chiffres décimaux à la racine carrée d'un nombre. Pourriez-vous me le fournir ?

C. Aoudi
61 Flers

■ Votre avis de recherche est maintenant diffusé auprès de tous les lecteurs de l'Op. C'est bien le diable s'il ne se trouve pas quelqu'un pour nous adresser un développement sur ce sujet (avec, ce serait très bien joué, l'organigramme). □

Magazine

Bientôt du nouveau

pour le poquette

Sharp/Tandy

■ Nous avons eu quelques informations un peu trop belles pour paraître vraies, mais elles semblent maintenant se confirmer : le PC 1211 devrait connaître courant 82 quelques développements intéressants. Il semblerait même que le nouvel ordinateur de poche de Sharp soit bien avancé. Ce matériel s'appellerait Sharp PC-500. Il utiliserait un vrai microprocesseur 8 bits et serait doté de 3,5 Ko de mémoire vive utilisateur extensible à 24 Ko, le tout dans un boîtier d'un volume équivalent à celui de l'actuel PC-1211. Les

autres caractéristiques du PC-1211 auraient été conservées, sauf peut-être le BASIC. Le clavier de la nouvelle machine comporterait entre autres des touches de fonction préprogrammables. Il est également question d'une imprimante améliorée qui permettrait l'impression en plusieurs couleurs (?). Une telle imprimante ne semble en fait pouvoir être utilisée valablement qu'avec un système beaucoup plus puissant que le PC-1211 actuel, ce qui confirme bien la montée en puissance que nous prévoyons. □

Des clubs

■ Vous êtes nombreux à nous demander dans votre courrier l'adresse d'un club près de chez vous auquel vous pourriez adhérer. A notre connaissance, la seule liste un peu importante qui ait été publiée pour les clubs d'informatique individuelle est celle de *Guide 81-82*, numéro hors série de nos amis de *L'Ordinateur*

Individuel ; mais nombre des clubs de cette liste ne s'intéressent pas aux poquettes. Bref, la seule source d'information qui vous reste, c'est ce bon vieux (?) *l'Op...* mais il connaît peu de clubs. Aussi, si vous appartenez à un club où les "micropochistes" sont bienvenus, dépêchez-vous de nous

envoyer avant le 5 février 1982 toutes les informations utiles : adresse, personne à contacter, types d'activité, lieux et dates des réunions du deuxième trimestre 82, etc. D'ici là, nous vous donnons ci-dessous quelques adresses de clubs dont nous avons eu connaissance :

Club informatique du Lycée Florent Schmitt
112 bd de la République
92210 SAINT-CLOUD

Club Micro-informatique de Merlebach
c/o René Freyberger
3 place de la Gare
57800 MERLEBACH

Association Informatique du Pays Haut
c/o Bernard Caron
5 rue de Paris
54440 HERSERANGE
Tél. : (8) 224 80 11

Club Informatique du Lycée Ampère
c/o Angel Martin
120 rue E. Locard
69005 LYON

De la mémoire

pour les HP-41

■ Si vous trouvez que la 41 est un peu chiche en mémoire morte MEM (comme les modules de bibliothèque), vous pouvez envisager d'y connecter un module d'une taille fort respectable (4, 8 ou 16 K octets) composé de circuits de mémoire morte programmable MEMP (EPROM) grâce à un nouveau produit américain. Cette extension, le HHP 4/8/16 de la société F. M. Weaver Associates, est un boîtier d'une taille proche de celle de la HP-41, qui peut se connecter comme un module ou un périphérique standard. Il est livré sans les circuits de mémoire MEMP, que l'utilisateur doit installer dans les supports à 28 broches prévus à cet effet, bien entendu après y avoir inscrit les pro-

grammes adéquats. Le logiciel nécessaire pour "brûler" les MEMP est également fourni.

Prix à l'unité : 241 \$ (1400 FF environ), ce qui ne le met pas à la portée de toutes les bourses, puisqu'il faut ajouter à ce prix celui des circuits MEMP et de leur programmation. Signalons toutefois que les adhérents du club PPC (voir *l'Op* numéro 3) peuvent acheter ce matériel avec une réduction (peut-être pas très importante), de même que les acheteurs par quantités.

Coordonnées du fabricant :
F.M. Weaver Associates
Hand Held Products
Div.
6201 Fair Valley Drive
Charlotte, NC 28211
USA

Magazine

Le Sinclair ZX

in the pocket ?

■ C'est apparemment ce à quoi s'emploie la firme britannique avec sa future TV de poche, qui devrait bien sûr être principalement destinée à des utilisations plus conventionnelles, comme de regarder en catimini un match de cricket (restons anglais) ou de football en classe ou au bureau.

Le matériel n'en est encore qu'au stade du prototype, mais les premiers

remment ce qui s'est produit au dernier trimestre de 81 pour le ZX-81, dont les délais de livraison semblent (d'après les courriers que nous recevons) plus proches de 8 à 10 semaines que des 4 annoncées dans les publicités : c'est dur le succès, et les Britanniques semblent actuellement être un peu débordés par la demande du marché français (plus de 6000 machines par mois !).



exemplaires de série devraient être disponibles en Grande-Bretagne au printemps 82, à un prix inférieur à 50 £ (550 FF environ). Le tube cathodique utilisé se caractérise principalement par sa faible épaisseur (2 cm), ce qui devrait permettre ultérieurement des postes de grandes dimensions "à pendre au mur".

Ce matériel ne sera sans doute pas disponible en France avant fin 82, d'une part à cause de la différence de standard TV (oui, même en noir et blanc !), d'autre part parce que le marché français ne sera probablement pas prioritaire par rapport aux marchés britannique et américain. C'est d'ailleurs appa-

Signalons de plus que la firme japonaise Panasonic annonce depuis peu aux Etats-Unis une autre TV de poche, le *Travelvision TR-1000P*. Sa conception semble plus classique (tube cathodique traditionnel), et les dimensions de l'écran sont plus petites pour un encombrement du poste assez voisin. Et pour revenir aux micropoches, mentionnons que dans notre prochain numéro nous ferons le point sur tous les systèmes d'affichage adaptés à une utilisation de poche ; notamment les écrans à cristaux liquides d'Epson (4 lignes de 20 caractères), Toshiba (8 lignes de 40 caractères, soit le tiers d'un écran d'Apple). □

■ UN LIVRE

Pico-informatique et gestion d'entreprise : modèles et programmes pour calculatrices.

Gary Baumgartner, Jean-Marie Petitgand.

Les Editions d'Organisation, Paris 1981.

228 pages, broché.

Prix : 94 FF ttc.

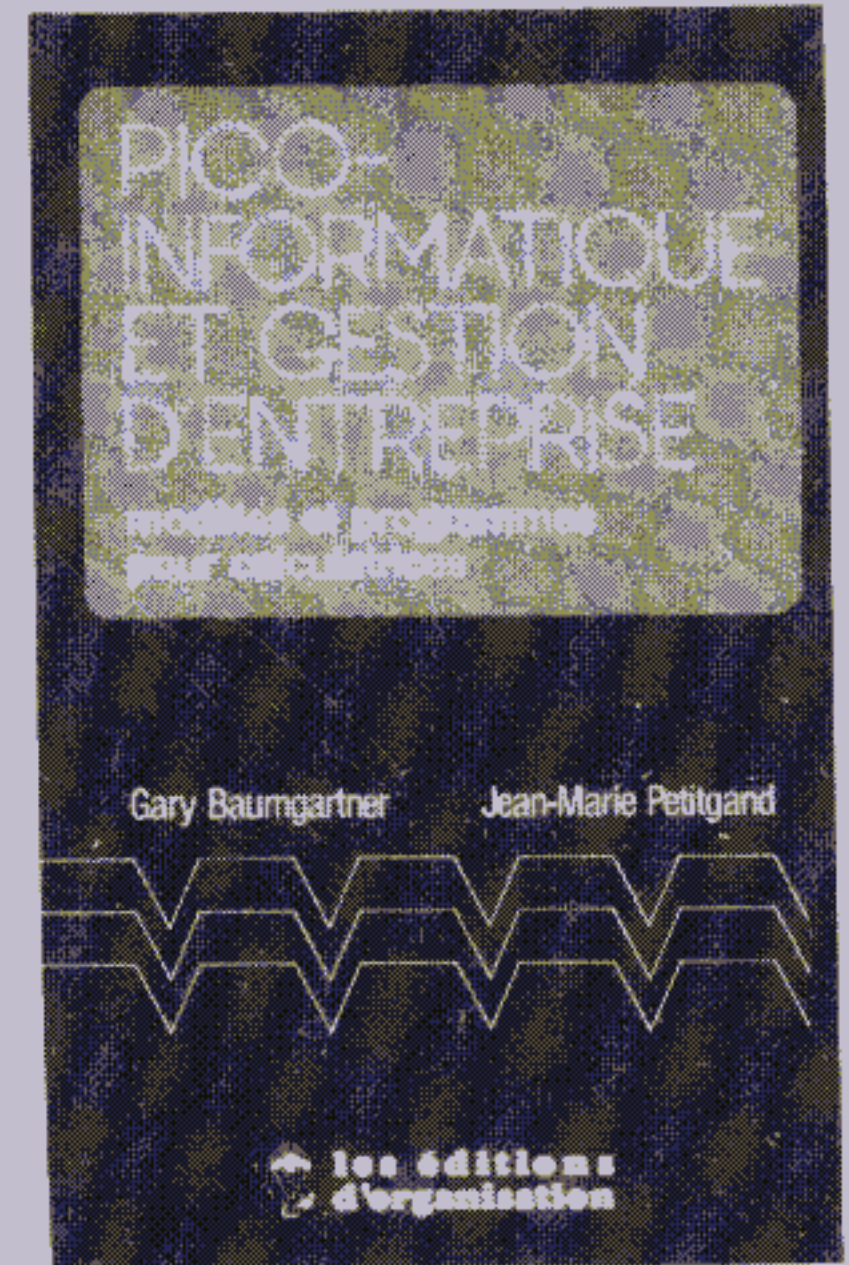
■ Cet ouvrage comporte 8 chapitres, chacun destiné à un problème particulier de la gestion d'une entreprise. Les grands thèmes en sont : la gestion des stocks et les réapprovisionnements, la gestion de la production (courbe d'apprentissage, rebuts de fabrication), la gestion financière (escompte ou sélection d'investissements et de financements), la prévision à court terme et la détermination d'échantillons pour des études de marché.

Comme le laisse entendre le titre, chaque chapitre se compose de l'énoncé du problème, accompagné du modèle proposé comme solution, du programme correspondant (sur HP-41 ou TI-59 suivant les cas) et de son organigramme, ainsi que de quelques exercices ou plutôt suggestions d'applications à des problèmes proches.

Les modèles présentés, comme les programmes donnés par leur texte ou leur organigramme, sont relativement simples à mettre en œuvre. Ils donneront sans doute de nombreuses idées aux utilisateurs d'ordinateurs individuels "de table" ou de "poche", tout au moins à ceux que la manipulation de chiffres et de formules n'effraie pas trop.

J'avoue avoir bien aimé ce livre et l'approche assez formelle des problèmes qui y sont traités, mais tout le monde n'aura sans doute pas la même réaction : certains passages sont un peu "universitaires" par leur formalisme. Ce formalisme risque de heurter ceux des gestionnaires qui sont habi-

tués à s'appuyer principalement sur leur expérience plus que sur des modèles mathématiques, même si ces modèles ont des bases empiriques éprouvées. Bref, les gestionnaires "scientifiques" ou rigoureux aimeront bien, les autres sans doute moins : dommage.



En plus des gestionnaires, cet ouvrage s'adresse également aux étudiants et enseignants de gestion : il sera effectivement utile aux uns et aux autres (sauf si le prof' et l'élève ont tous les deux ce livre !), notamment grâce aux nombreux exemples, exercices et suggestions de variantes qu'il comporte.

Enfin, les auteurs destinent aussi leur livre « aux informaticiens, ingénieurs, membres des clubs passionnés d'informatique auxquels il offre la possibilité d'appliquer, d'approfondir et de compléter leurs connaissances ». C'est exact, cet ouvrage semble bien fait pour ceux qui sont avides de connaissances en programmation et en gestion. Cependant, il leur faudra parfois ignorer certains développements peut-être un peu trop mathématiques, et sauter directement aux applications.

Plus généralement, tous ceux des lecteurs de *l'Op* qui sont à la recherche d'utilisations de leur micropoche (ou d'un système plus gros) pour la gestion de leur entreprise, trouveront dans ce livre une mine de programmes, tout faits ou faciles à réaliser, qui devraient les satisfaire.

□ BS

ABONNEZ-VOUS A

L'Ordinateur de poche

C'est la seule revue française exclusivement dédiée à l'informatique de poche.

Si vous possédez déjà un ordinateur de poche ou une calculatrice programmable, vous trouverez dans L'ORDINATEUR DE POCHE des tas d'astuces qui vous permettront de tirer un meilleur parti de votre machine.

Si vous envisagez d'en acheter une, L'ORDINATEUR DE POCHE sera pour vous un guide de choix irremplaçable.

N'hésitez pas à investir 45 Francs^(*) pour une meilleure information.

Retournez aujourd'hui même le bulletin d'abonnement ci-dessous.

Bulletin à retourner à
L'ORDINATEUR DE POCHE Service Abonnements
41 rue de la Grange aux Belles 75483 Paris Cedex 10

Nom _____ Prénom _____
Adresse _____
Pays _____ Code postal _____ Ville _____

Veuillez m'abonner pour un an (4 numéros) à L'ORDINATEUR DE POCHE,
ci-joint mon règlement de 45 FF.

(*Etranger : 60 FF, Belgique : 320 FB, Suisse : 18 FS).

(Tarif par avion : Afrique, Moyen-Orient : 110 FF; autres pays : 85 FF)

Enfin en France LE SINCLAIR

VOTRE MICRO-ORDINATEUR INDIVIDUEL POUR SEULEMENT 764^F TTC complet en kit

Quelques heures bien utilisées pour une bonne compréhension du micro-ordinateur.

C'est en 1980 qu'a été fait un pas en avant décisif : l'apparition du Sinclair ZX80, le premier micro-ordinateur individuel vendu pour 1.250 F. Pour 1.250 F, le ZX80 présentait des caractéristiques et des fonctions inconnues dans sa gamme de prix.

Plus de 50.000 ZX80 ont été vendus en Europe et cet ordinateur a reçu les louanges unanimes des professionnels de l'informatique. Aujourd'hui, l'avance de Sinclair augmente. Pour 985 F, le nouveau Sinclair ZX81 vous permet de bénéficier de fonctions encore plus évoluées à un prix encore plus bas. Et en kit, au prix de 764 F, le ZX81 est encore plus économique.

Prix plus bas : capacités plus grandes

Il est toujours aussi simple d'apprendre à utiliser vous-même votre ordinateur, mais le ZX81 vous apporte des possibilités plus larges que le ZX80. Le microprocesseur est le même, mais le ZX81 contient une ROM BASIC 8K nouvelle et plus puissante, qui constitue "l'intelligence domestiquée" de l'ordinateur. Ce dispositif travaille en système décimal, traite les logarithmes et les fonctions trigonométriques, vous permet de tracer des graphiques et construit des présentations animées.

Le ZX81 vous permet de bénéficier d'autres avantages – possibilité d'enregistrer et de conserver sur cassette des programmes donnés par exemple, de sélectionner par le clavier un programme sur une cassette.

Si vous avez un ZX80...

La nouvelle mémoire ROM BASIC 8K du ZX81 peut être utilisée avec un ZX80 comme circuit de remplacement (elle est complète, avec un nouveau clavier et un nouveau manuel d'exploitation).

A l'exception des fonctions graphiques animées, toutes les fonctions plus évoluées du ZX81 peuvent être intégrées à votre ZX80, y compris la possibilité de commander l'imprimante Sinclair ZX.

L'imprimante ZX pour 690 F TTC

Conçue exclusivement pour le ZX81 (et pour le ZX80 avec la ROM BASIC 8K), cette

imprimante écrit tous les caractères alphanumériques sur 32 colonnes et trace des graphiques très sophistiqués. Parmi les fonctions spéciales, COPY imprime exactement ce qui se trouve sur tout l'écran du téléviseur, sans demander d'autres instructions. L'imprimante ZX sera disponible à partir de septembre, au prix de 690 F TTC. Commandez-la!



Mémoire RAM 16K-octets : une augmentation de mémoire massive.

Conçue comme un module complet adaptable à votre Sinclair ZX80 ou ZX81, la mémoire RAM s'enfiche simplement dans le canal d'expansion existant à l'arrière de l'ordinateur : elle multiplie par 16 la capacité de votre mémoire des données/programmes!

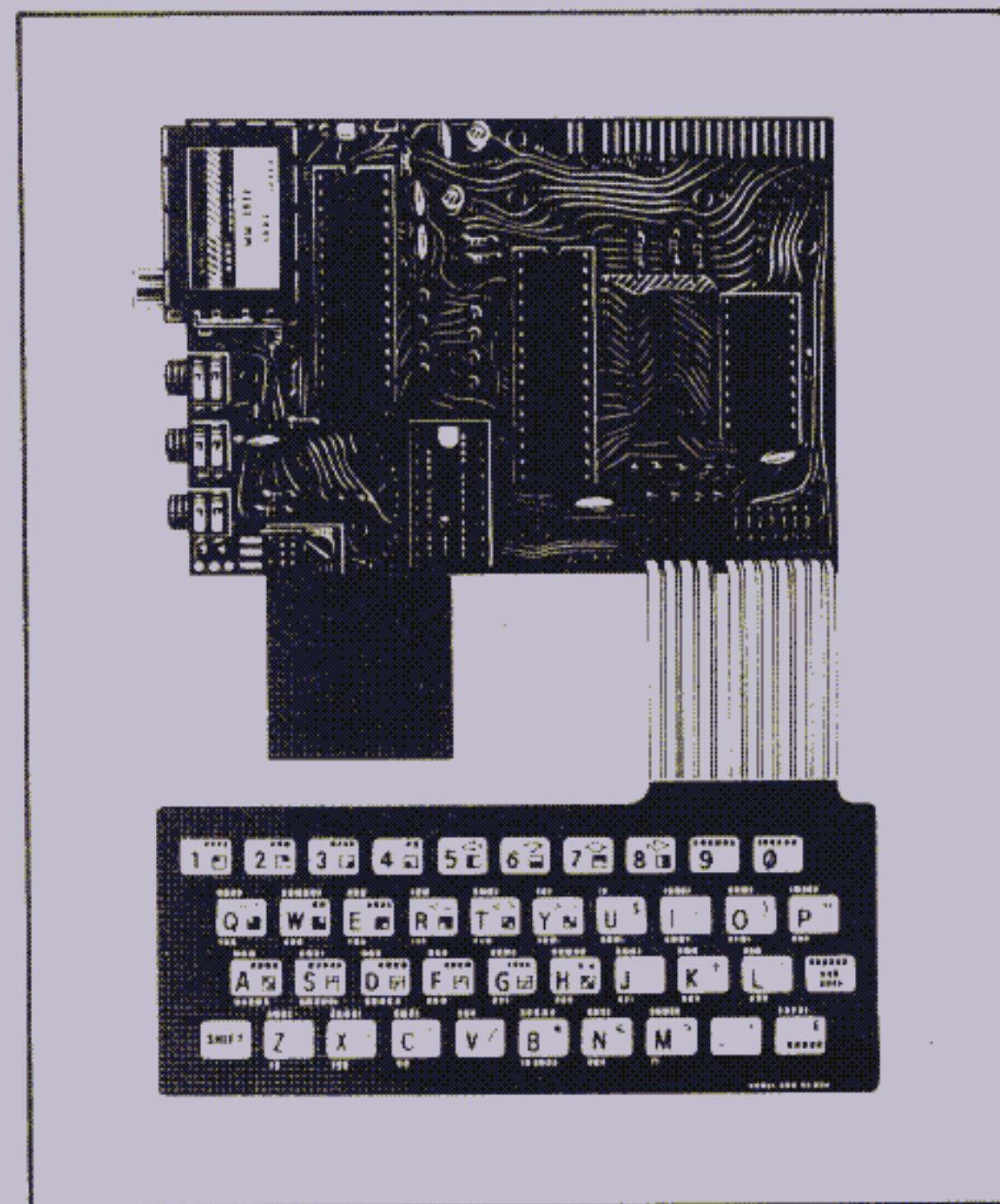
Vous pouvez l'utiliser pour les programmes longs et complexes, ou comme base de données personnelles. Et pourtant, elle ne coûte que la moitié du prix des modules de mémoire complémentaires de la concurrence.



Comment peut-on baisser le prix en augmentant les spécifications ?

Très simple, tout se fait au niveau de la conception. Dans le ZX80, les circuits actifs de l'ordinateur sont passés de 40 environ à 21. Dans le ZX81, les 21 sont devenus quatre ! Le secret : un circuit totalement nouveau. Conçu par Sinclair et fabriqué spécialement en Grande-Bretagne, ce circuit nouveau remplace 18 puces du ZX80.

En kit ou monté, à vous de choisir !



La photo illustre la facilité de montage du kit ZX81.

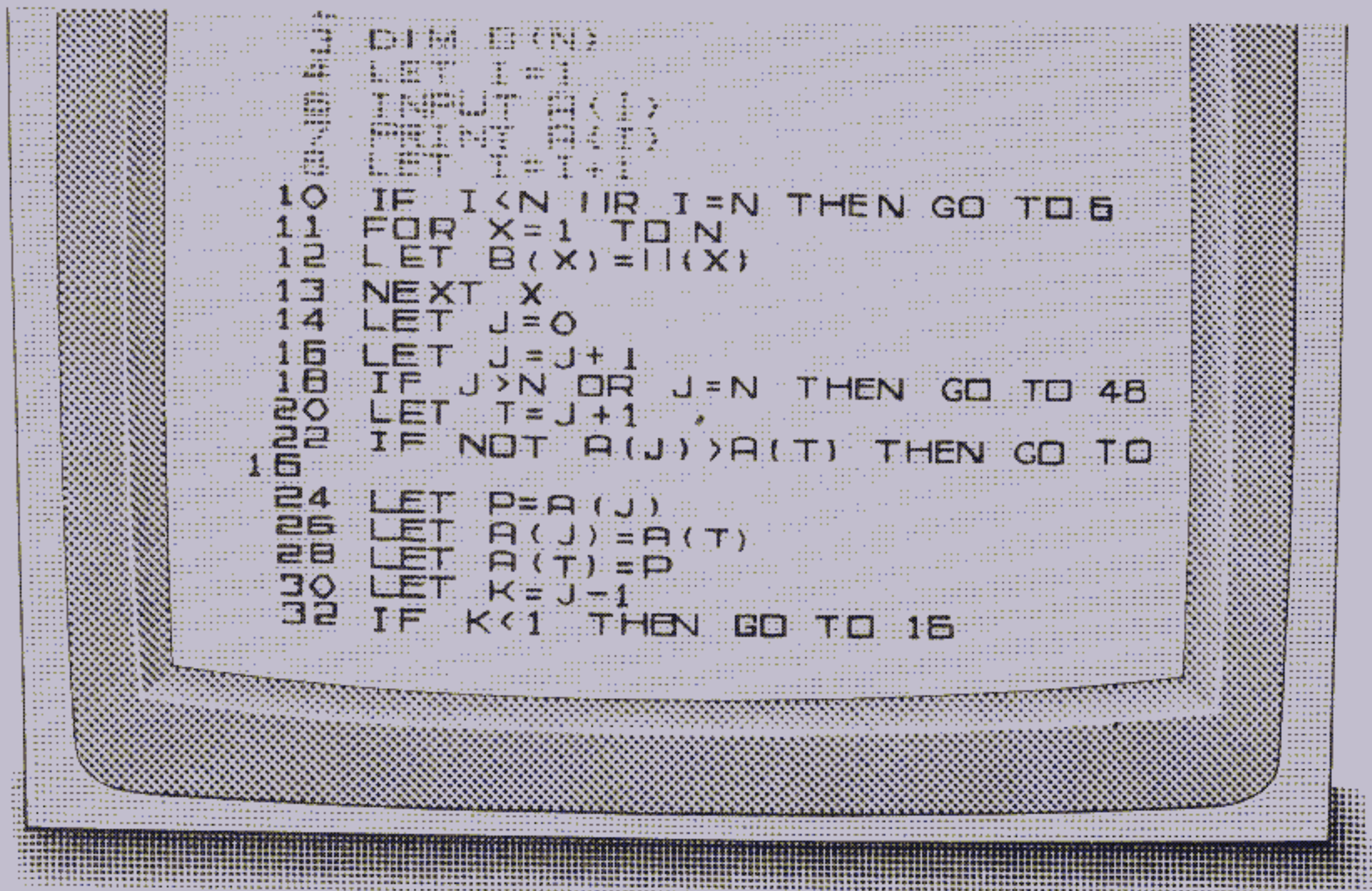
Quatre circuits à monter (avec, bien entendu, les autres composants), quelques heures de travail avec un fer à souder à panne fine.

Les versions montées et en kit sont complètes, c'est-à-dire qu'elles contiennent tous les conducteurs requis pour connecter le ZX81 à votre téléviseur (couleur ou noir) et à votre enregistreur à cassette.

Un microprocesseur ayant fait ses preuves, une nouvelle mémoire morte BASIC 8K, une mémoire à accès sélectif et un nouveau circuit maître unique.

ance! ZX81

985^F
TTC
monté



Une nouvelle
spécification
améliorée

- Le micro-processeur ZX81 - une nouvelle version plus rapide du fameux ZX80, reconnu à l'unanimité comme le meilleur de sa catégorie.
- Fonction exclusive d'entrée de "mots-clés" par une touche : le ZX81 supprime une grande partie des opérations fastidieuses de dactylographie. Les mots-clés comme RUN, LIST, PRINT, etc. sont entrés par une seule touche spécialisée.

- Codes uniques de présentation et de contrôle de syntaxe identifiant immédiatement les erreurs de programmation.
- Gamme complète de fonctions mathématiques et scientifiques avec une précision de 8 positions décimales.
- Fonctions de traçage de graphiques et d'affichages animés.
- Tableaux numériques et chaînes multi-dimensionnelles.
- Jusqu'à 26 boucles FOR/NEXT.
- Fonction RANDOMISE, utile pour les jeux comme pour les applications sérieuses.
- Enregistrement (LOAD) et conservation (SAVE) sur cassette de programmes donnés.
- Mémoire vive 1K-octets pouvant être portée à 16K octets grâce au module RAM Sinclair.
- Possibilité de commander la nouvelle imprimante Sinclair.
- Conception évoluée à quatre circuits : micro-processeur, mémoire morte, mémoire vive et circuit principal - circuit unique fabriqué spécialement pour remplacer 18 puces du ZX80.

Pour toute information : 359.72.50 (4 l. groupées).

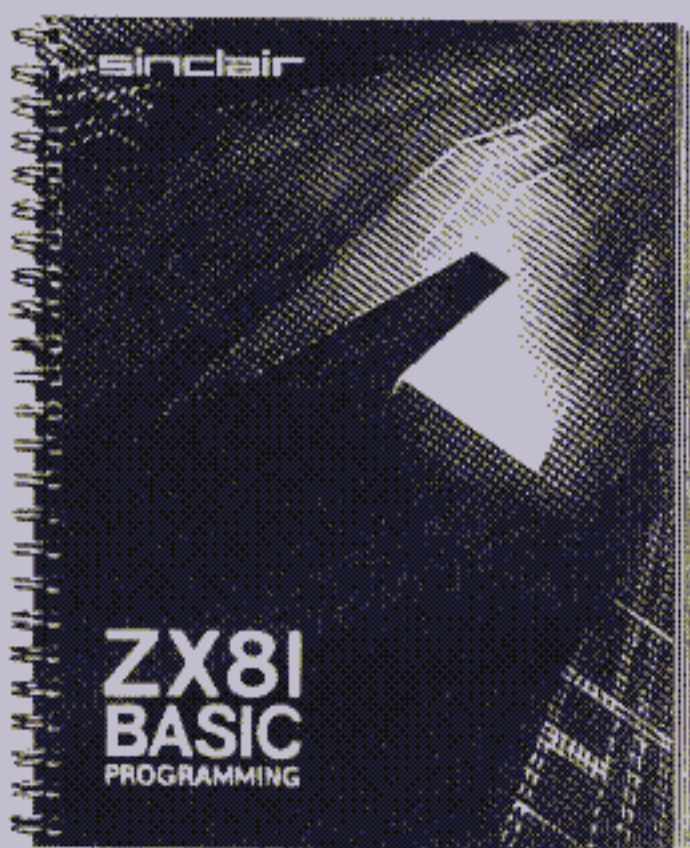
Pour commander votre ZX81

Par coupon-réponse, en utilisant l'imprimante ci-dessous. Vous pouvez payer par chèque ou par mandat-postal.

Quel que soit le cas, vous recevrez sous 4 semaines votre micro-ordinateur Sinclair. Votre imprimante vous sera expédiée sous un délai de 10 semaines. Et, bien entendu, vous disposez de 14 jours pendant lesquels vous pouvez demander le remboursement. Nous voulons que vous soyez satisfait, sans doute possible, et nous sommes convaincus que vous le serez.

sinclair ZX81

Nouveau
manuel BASIC.



Chaque ZX81 est accompagné d'un manuel de programmation et langage BASIC;

ce manuel est complet, il est rédigé spécialement et traduit en français pour permettre au lecteur d'étudier d'abord les premiers principes puis de poursuivre jusqu'aux programmes complexes.

Découpez ce bon et envoyez-le à : DIRECO INTERNATIONAL, 30, avenue de Messine, 75008 Paris OP4

Je désire recevoir sous 4 semaines (ou 10 semaines pour l'imprimante), par paquet-poste recommandé :

le micro-ordinateur Sinclair ZX 81 en kit avec son adaptateur secteur et le manuel BASIC pour le prix de 764 F T.T.C.

le micro-ordinateur Sinclair ZX 81 monté avec son adaptateur secteur et le manuel BASIC pour le prix de 985 F T.T.C.

l'extension de mémoire RAM (16 K-octets) pour le prix de 650 F T.T.C.

l'imprimante pour le prix de 690 F T.T.C. (paiement séparé).

Je choisis de payer :

par C.C.P. ou chèque bancaire établi à l'ordre de Direco International, joint au présent bon de commande.

directement au facteur, moyennant une taxe de contre-remboursement de 14 F.

par carte bleue, n° de carte joint.

Nom _____

Prénom _____ N° _____

Rue ou Lieu-dit _____

Commune _____ Code postal _____

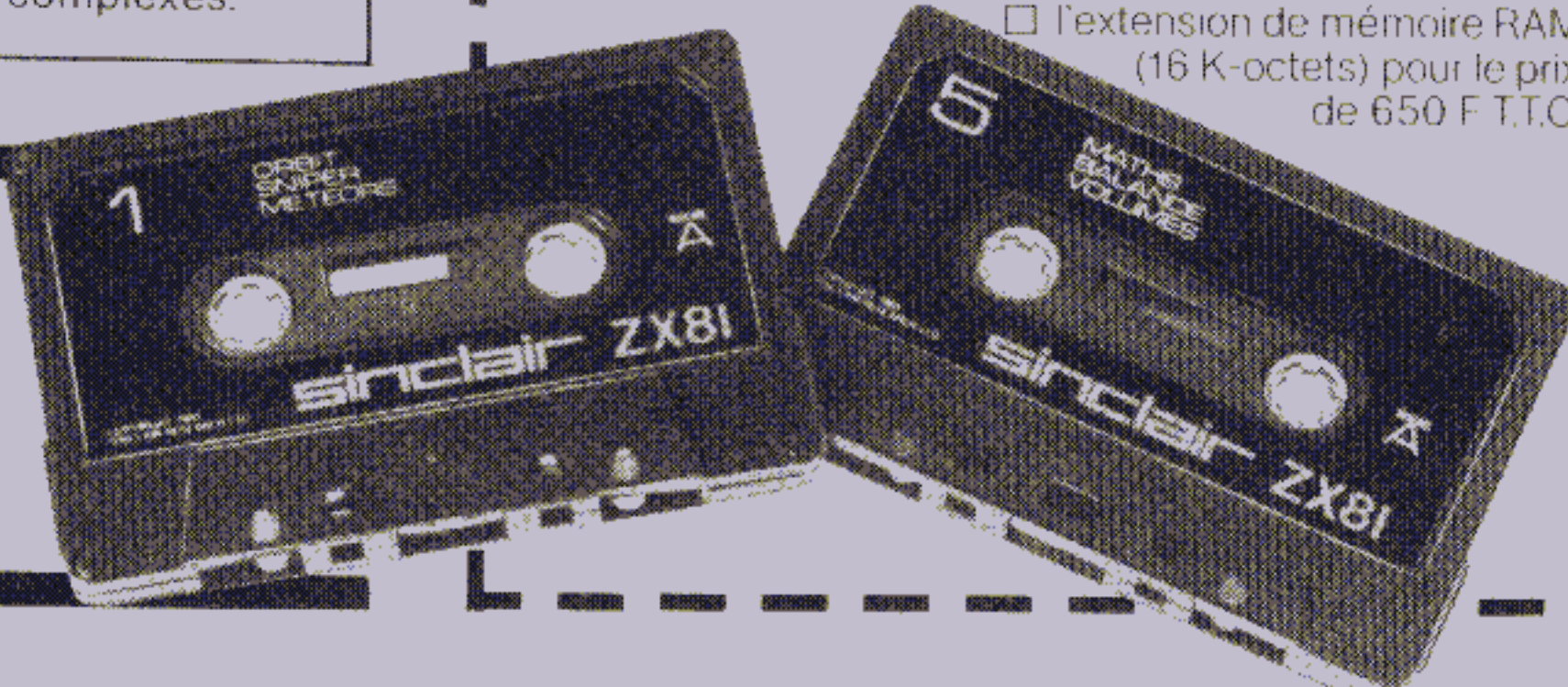
Localité du bureau de poste _____

Signature _____

(Pour les moins de 18 ans, signature de l'un des parents).

GRATUIT

Pour toute commande passée avant le 10 décembre 1981, 2 cassettes préprogrammées offertes en cadeau.



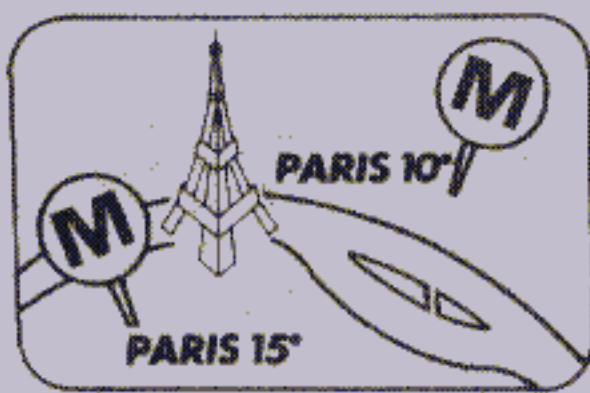
Démonstration
chez Direco-International.

ILLEL

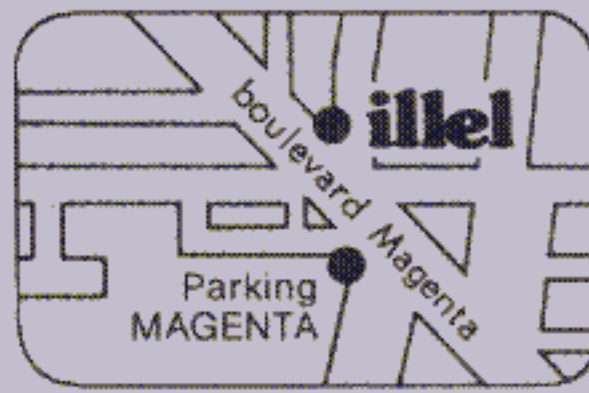
ESPACE ELECTRONIQUE

ILLEL : LES SERVICES, LES PRIX, DEUX ESPACES INFORMATIQUES A PARIS.

Deux grands espaces de présentation et de démonstration, où vous serez accueilli et conseillé par un personnel sérieux et hautement qualifié. Initiés ou non, vous trouverez des interlocuteurs courtois capables de vous guider aussi bien pour le choix d'un micro-ordinateur que celui d'une calculatrice, d'une imprimante, d'un moniteur ou d'un logiciel approprié à votre configuration. Micro-informatique de gestion ou de loisirs, ILLEL présente les plus grandes marques actuelles du marché aux prix les plus compétitifs. N'hésitez pas, par curiosité, pour un conseil, un achat, ou simplement pour nous dire bonjour, si la micro-informatique vous intéresse, elle est aussi notre passion, alors, venez nous rendre visite, nous nous entendrons sûrement.



ILLEL CENTER PARIS 15° :
143, av. Félix-Faure, 75015 Paris
Tél. 554.97.48
Métro : Balard.



ILLEL CENTER PARIS 10° :
86, bd Magenta, 75010 Paris
Tél. 201.94.68
Métro : Gare de l'Est
Parking : Magenta.

Ouvert de 9 h 30 à 12 h 30 et de 14 h à 19 h - Lundi à partir de 15 h.

FACILITÉS DE PAIEMENT, CRÉDIT DE 4 À 36 MOIS, LEASING SUR 3 OU 4 ANS - VENTE PAR CORRESPONDANCE.



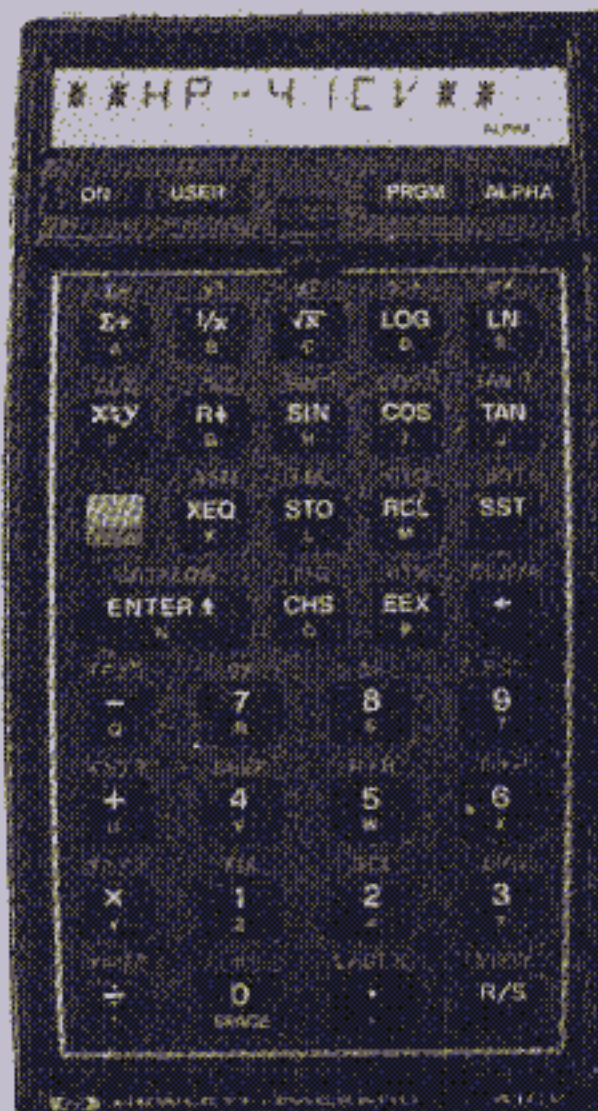
Sharp PC 1211

Un ordinateur de poche maniable utilisant le langage BASIC.

De nombreuses fonctions à performances élevées sont incorporées dans un corps plat et compact. Le PC 1211 est conçu comme un ordinateur du type "à action réciproque" pour satisfaire les besoins personnels de l'utilisateur qui emploie le langage BASIC facile à comprendre.

Une petite machine performante originale jusqu'au bout des touches.
1 424 pas de programme. 26 mémoires.

PROMOTION
1 050 F T.T.C.

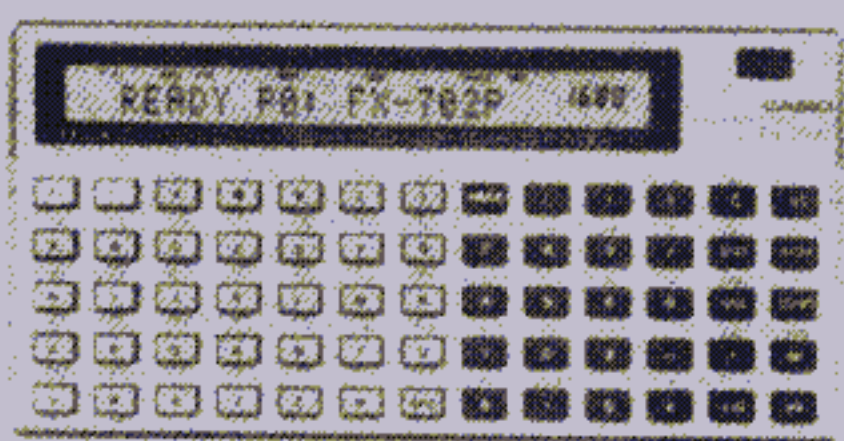


H.P. 41C

Calculateur programmable.

Affichage alphanumérique. Mémoire à contrôle dynamique. Modules-mémoire enfichables. Modules d'application enfichables. Mémoire permanente.

PROMOTION
H.P. 41C
+ bibliothèque
maths ou jeux
1 790 F T.T.C.



CASIO FX-702 P

BASIC
alphanumérique

PRIX T.T.C. 1 250 F
prix au comptant

Caractéristiques de programmation

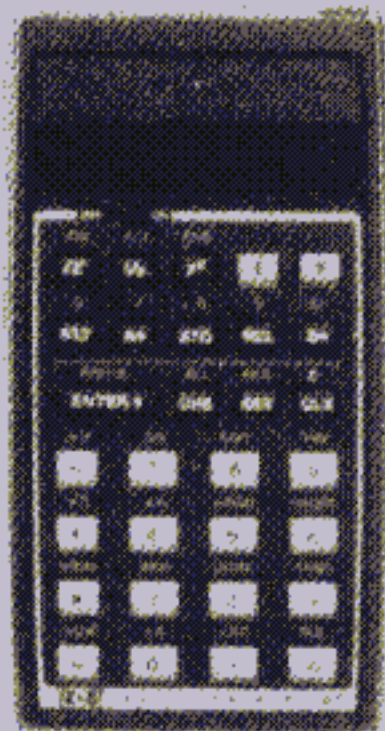
- Capacité de mémoire (protégée) : variable suivant affectation : de 1 680 pas de programmes avec 26 mémoires, jusqu'à 80 pas de programmes avec 226 mémoires.

- Langage de programmation : BASIC (pouvant utiliser des touches préprogrammées).
- Nombre de programmes : possibilité de définir 10 programmes (P0 à P9). Numérotation de ligne possible jusqu'à 9 999.
- Possibilité de boucles : appel à des sous-programmes (10 niveaux); utilisation de boucles (8 niveaux).
- Fonction de contrôle et d'édition : correction, effacement, insertion et ajout par déplacement de curseur.
- Possibilité d'entrée par KEYIN.
- Possibilité d'utiliser mémoire externe avec adaptateur magnétophone (Interface FA-21) :
 - Extension mémoire (date ultérieure) par adjonction de RAM enfichables.
 - Connexion à une mini-imprimante (date ultérieure).
- Fonctions diverses :
 - trigonométriques et inverses,

- hyperboliques et inverses,
- logarithmiques et exponentielles,
- racine carrée, puissance, factorielles,
- conversation sexagésimal → décimal,
- conversion polaire → rectangulaire, valeurs absolues, arrondis, partie entière, fraction.

Caractéristiques de calculs

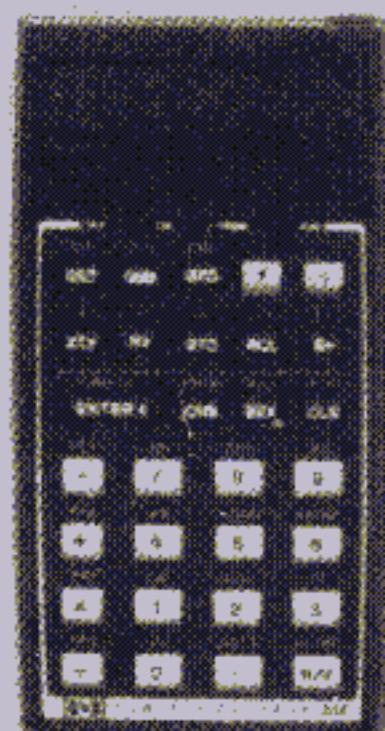
- Capacité : 10 chiffres + 2.
- Fonctions de base : 4 opérations, nombres négatifs, parenthèses, logique algébrique.
- Fonctions statistiques : résultats automatiques par touche ASTAT; nombre de données, Σx , Σy , Σx^2 , Σy^2 , Σxy .
Déviation standard : moyenne \bar{x} , \bar{y} , σ_n , σ_{n-1} .
Régression linéaire : coefficients de régression, coefficients de corrélation, valeurs approchées x , y .
- Fonctions sur chaînes de caractères : LEN, MID.



430 F

HP-32E

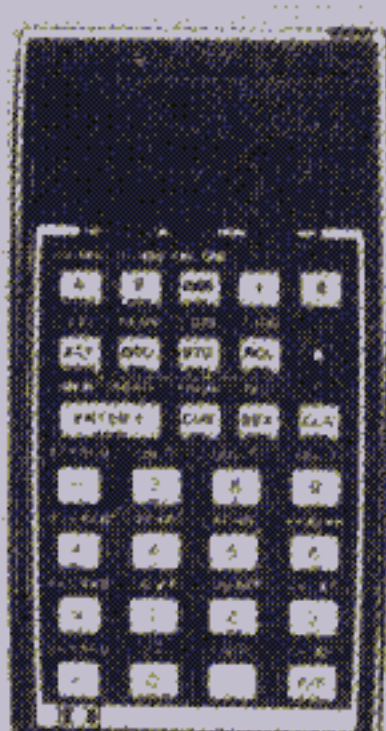
Calculateur scientifique avec fonctions hyperboliques et leurs inverses. Fonction factorielle. 15 mémoires adressables R₀ à R₈ et R.₀ à R.₈.



740 F

HP-33C

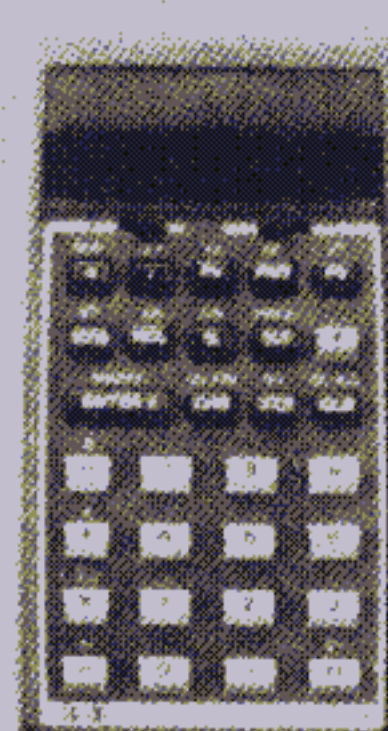
Calculateur scientifique programmable. 49 lignes de programme. 3 niveaux de sous-programme. 8 tests de comparaison. 8 mémoires adressables R₀ à R₈.



1.290 F

HP-34C

Calculateur scientifique programmable. 6 niveaux de sous-programme. 4 indicateurs binaires. Deux nouvelles fonctions SOLVE et INTEGRATE.

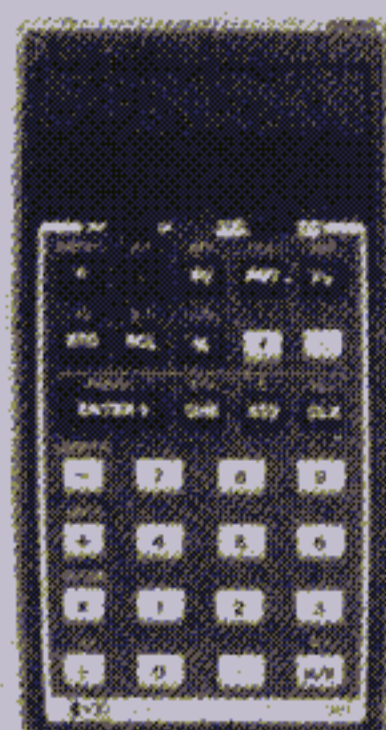


590 F

HP-37E

Calculateur financier. Fonctions statistiques. Fonctions mathématiques : $1/x$, \sqrt{x} , LN, e^x , y^x , n!
7 mémoires adressables R₀ à R₆.

2 NOUVELLES
CALCULATRICES
PRÉSENTÉES
DANS LES MAGASINS
HP 11 C et HP 12 C
995 F et 1190 F



1.290 F

HP-38C

Calculateur financier programmable - 5 registres financiers. 7 à 20 mémoires adressables R₀ à R₉ et R.₀ à R.₉. Fonctions statistiques.



2.390 F

HP-41 CV

Calculateur programmable affichage alphanumérique. Mémoire à contrôle dynamique. Modules mémoire enfichables. Modules d'applications enfichables. Mémoire permanente.



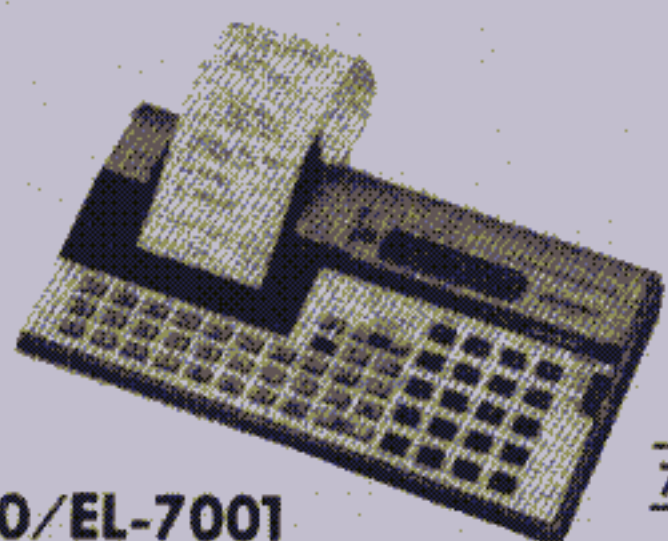
2.900 F

HP-67/HP-97

Calculateurs programmables 4 reg. opérat. + 1 reg. Last X. 26 mémoires. Fonctions scientifiques. Fonctions statistiques. 224 lignes de programme. 3 niveaux de sous-programme.



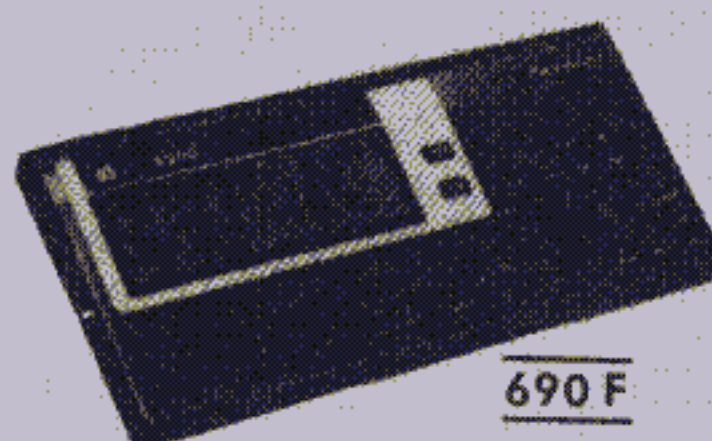
5.590 F



770 F

EL-7000/EL-7001

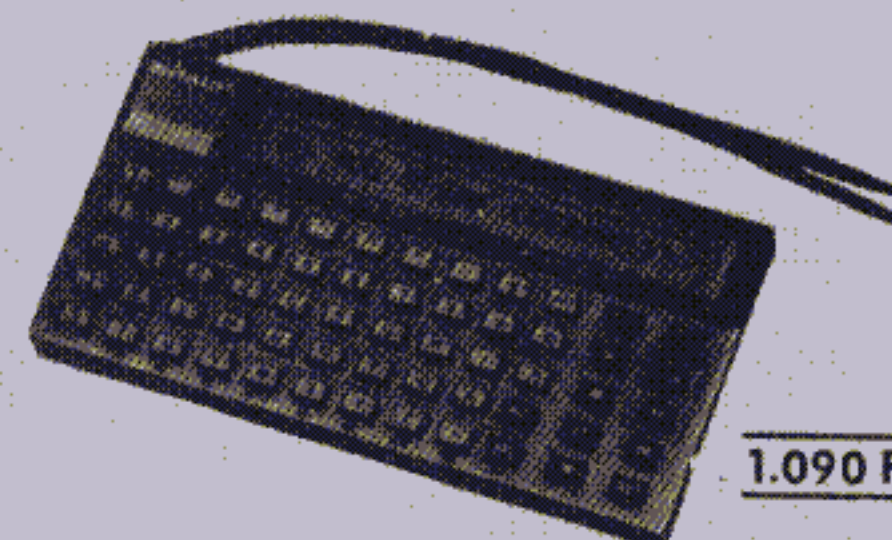
Calculatrice imprimante papier standard. Mémoire : 8 mots (EL-7000) 40 mots (EL-7001). 120 chiffres (EL-7000) 600 chiffres (EL-7001).



690 F

EL-6200

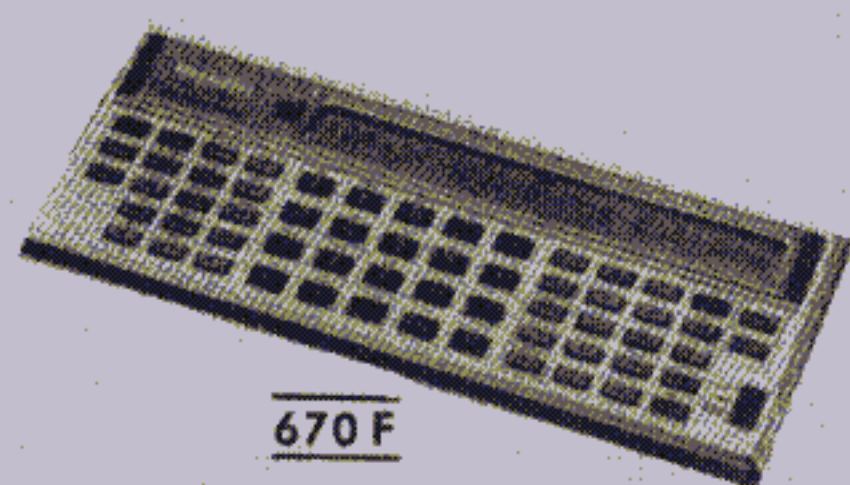
Calculatrice à double affichage 12 + 12 chiffres. Affichage de symboles identifiant appels téléphone, rendez-vous, etc.



1.090 F

IQ-3100

Calculatrice (traduction instant. 3 langues). Modules de langues (option) : IQ-31G Allem. IQ-31F Franç. IQ-31JE/J Jap. IQ-31I Ital. IQ-31S Esp.



670 F

EL-5100

Calculatrice affichage alphanumérique. Entrée de formules. Fonction de reproduction; 61 fonctions scientifiques.

à découper, à remplir et à retourner à : ILLEL CENTER INFORMATIQUE | service vente par correspondance
143, avenue Félix-Faure, 75015 Paris.

Je commande ferme et désire recevoir en urgence le matériel suivant : _____

au prix HT de F _____ + TVA 17,60 % _____ = TOTAL TTC _____ F

Mode de règlement :

Comptant Crédit* Leasing** Je verse au comptant la somme de (20 % minimum pour le crédit) _____

Ci-joint : Chèque bancaire CCP Mandat-carte

NOM _____ PRÉNOM _____ N° téléphone _____

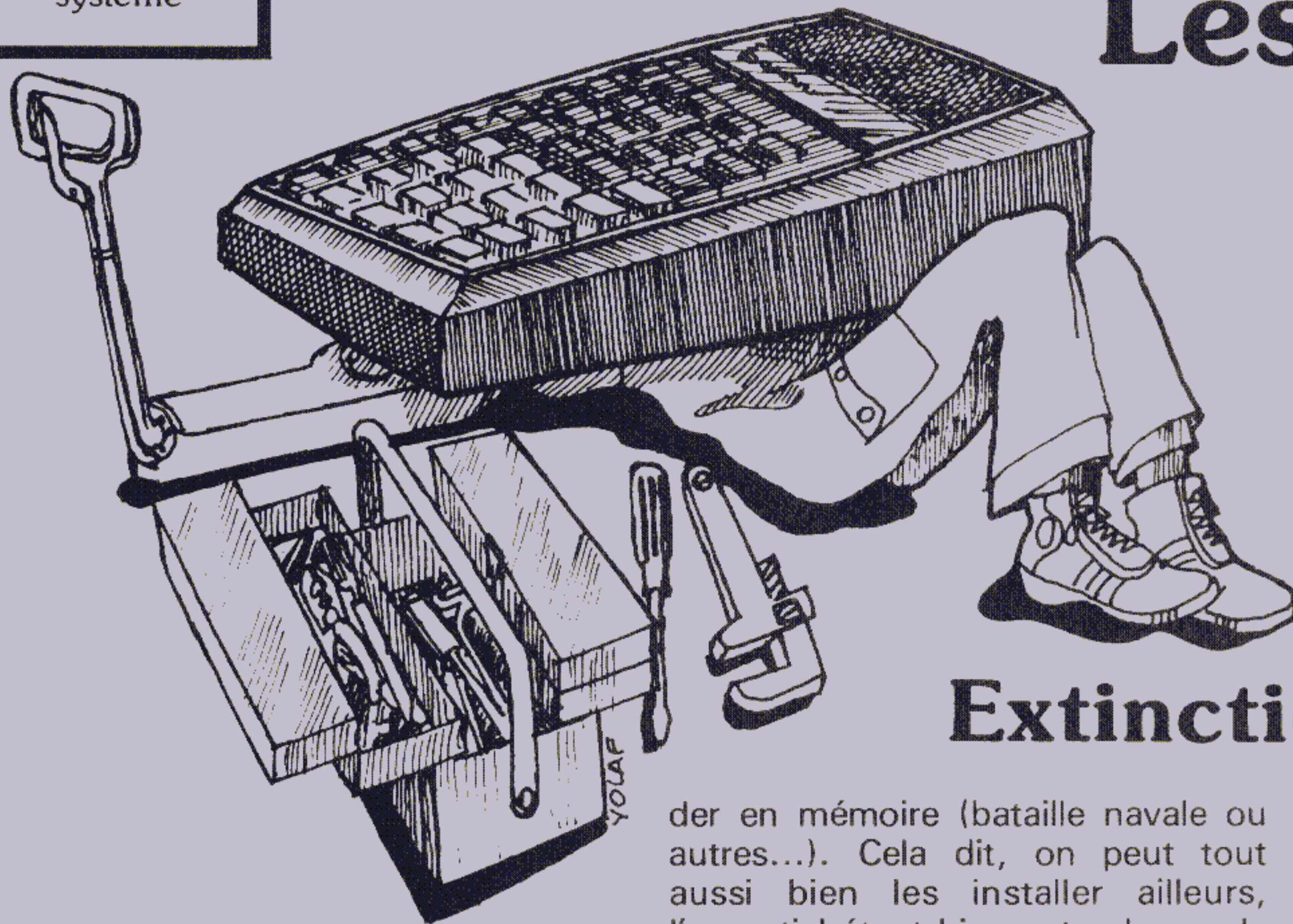
ADRESSE _____ CODE POSTAL _____

Ajouter 30 F de port et d'emballage pour toute commande inférieure à 2 000 F TTC (pas d'envoi contre-remboursement)

* Conditions de crédit : CREG ● être salarié,
● 20 % minimum au comptant, solde arrondi à la centaine supérieure.

** Conditions de leasing : SOVACREG ● être salarié,
● pas de versement comptant, loyer réparti sur 48 ou 36 mois.

Date et signature :



Les dessous de la TI-57 (suite)

Extinction des feux

On savait déjà qu'il est possible d'éteindre l'affichage de la TI 57 sans toucher à l'interrupteur. Mais on ne savait pas comment le rallumer. Voici une façon de le faire.

■ Depuis bientôt six mois, je n'ai rien trouvé dans *l'Ordinateur de poche* concernant "les dessous de la TI 57". C'est pourquoi je vous propose une astuce qui permet, sans modification de la machine, d'éteindre et de rallumer à volonté l'afficheur en conservant intact le contenu des mémoires (tant programme que données) et cela ne coûte que deux pas de programme. Je pense que ce petit sacrifice vaut largement l'accroissement d'autonomie qui en découle.

Pour aboutir à ce résultat, il suffit d'utiliser deux instructions l'une à la suite de l'autre : l'instruction codée 38 qui s'obtient en frappant 2nd Exc suivi de SST et une étiquette quelconque (2nd Lbl 1 par exemple). Il ne nous en faut pas plus pour obtenir un petit interrupteur logiciel de l'affichage.

On peut installer ces deux instructions aux pas 48 et 49, les 48 premiers pas étant réservés au programme *sérieux* que l'on désire gar-

der en mémoire (bataille navale ou autres...). Cela dit, on peut tout aussi bien les installer ailleurs, l'essentiel étant bien entendu que le programme principal "tourne" indépendamment de cet interrupteur.

La procédure à suivre est donc de frapper au clavier GOTO 2nd 48 (si l'on a décidé de placer ce petit programme en fin de liste), puis LRN, 2nd Exc, SST et 2nd Lbl 1.

Pour éteindre la calculatrice, il suffit dès lors de taper GOTO 2nd 48 (positionnement du pointeur sur le début de la séquence d'extinction) puis d'appuyer sur R/S : l'affichage devient clignotant, on tape alors

INV STO 3 et un petit tiret — toujours clignotant — apparaît à la droite de l'affichage.

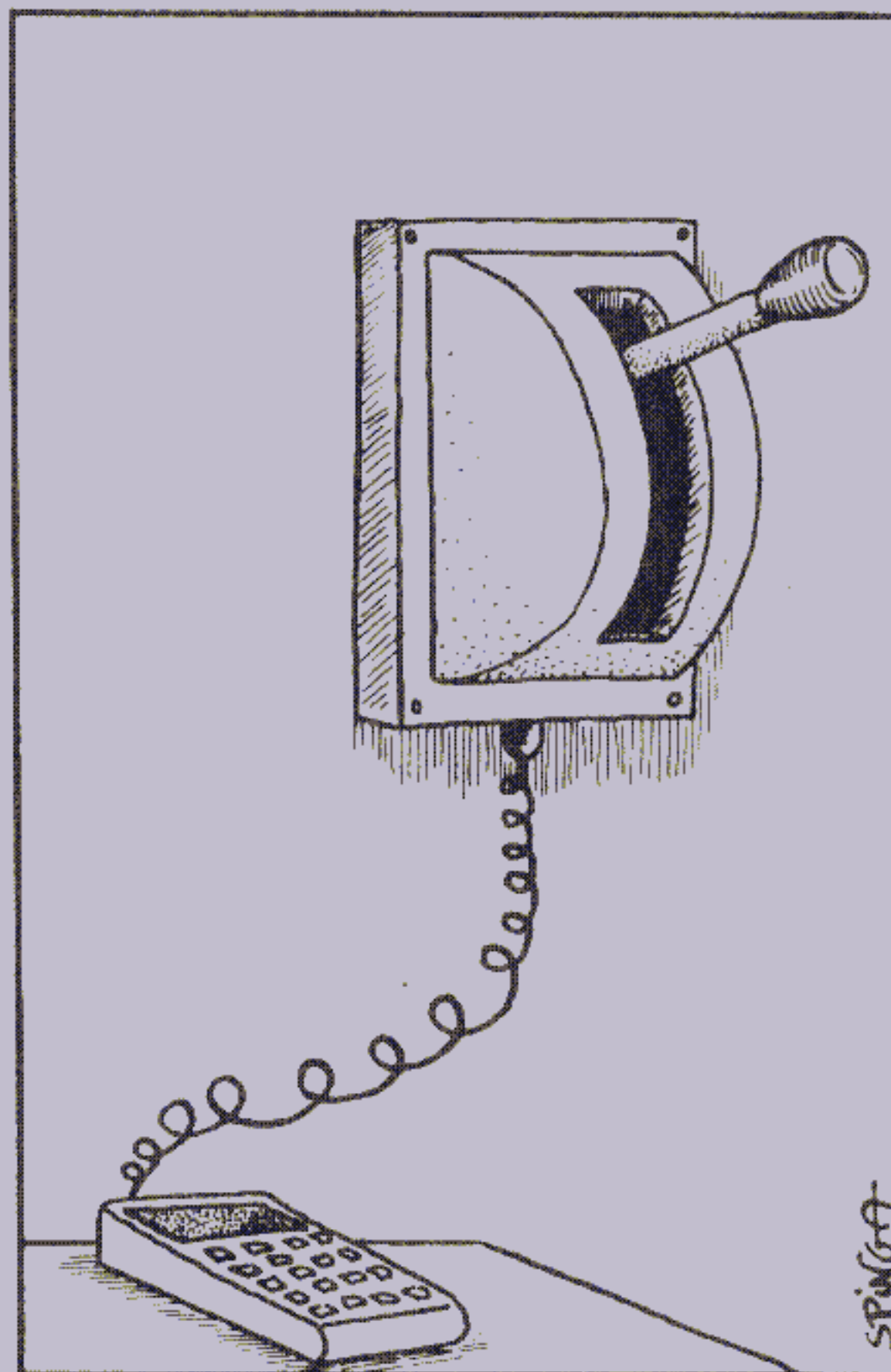
Une première pression sur +/- le déplace d'une case vers la droite et la seconde le chasse : l'affichage est éteint, mais les circuits de la machine sont toujours alimentés.

Pour rallumer l'affichage, il suffira de demander INV 2nd Fix et CLR. Sans CLR, l'affichage continuerait à clignoter. On peut d'ailleurs très facilement éliminer ce petit inconvénient si l'on veut bien consacrer un pas de plus à cet utilitaire : il faut ajouter à la séquence déjà décrite le R/S réglementaire qui évitera le clignotement. Au pas 47, on inscrira 2nd Exc, SST ; au pas 48, 2nd Lbl 1, et R/S au pas 49. L'extinction des diodes s'obtiendra alors par GOTO 2nd 47, R/S, INV STO 3, +/-, +/-, et le réallumage par INV 2nd Fix.

Avant d'appuyer deux fois sur +/- pour rendre le micropoche aveugle, lorsque seul un petit trait est affiché, on peut s'amuser à essayer des opérations ou à faire exécuter des programmes. On vérifiera à cette occasion que la machine est alors en Fix - 1. Il y a sûrement beaucoup à "piocher" de ce côté-là. Peut-être même une suite pour "les dessous de la TI 57", qui sait ? Faites-nous part de ce que vous aurez trouvé.

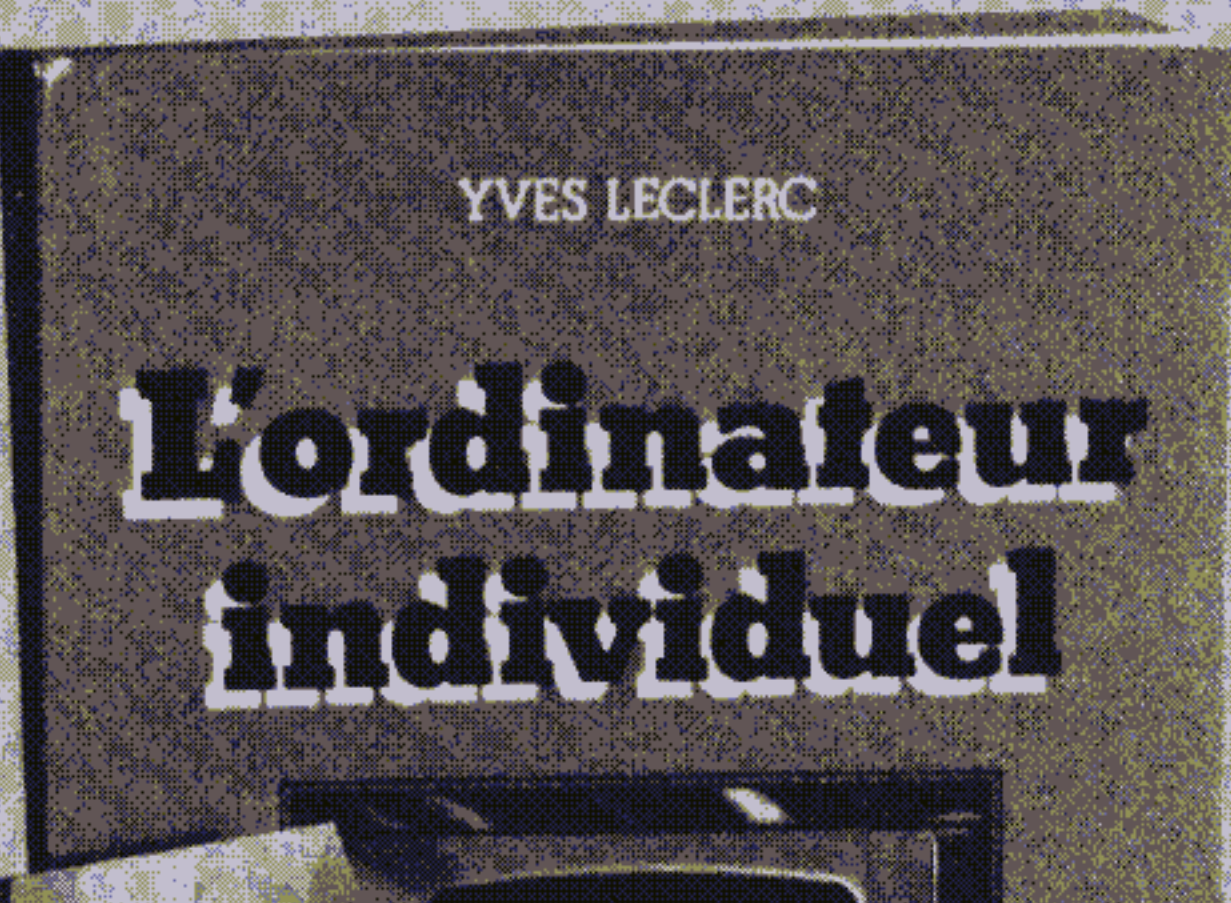
Pour faire réapparaître le petit tiret, lorsqu'on est ainsi parti à la découverte du Fix - 1 de ce micropoche, il suffit d'afficher zéro (ou un nombre positif inférieur à l'unité) et d'appuyer sur = ; deux pressions sur +/- le feront disparaître.

□ Dominique Thiébaud



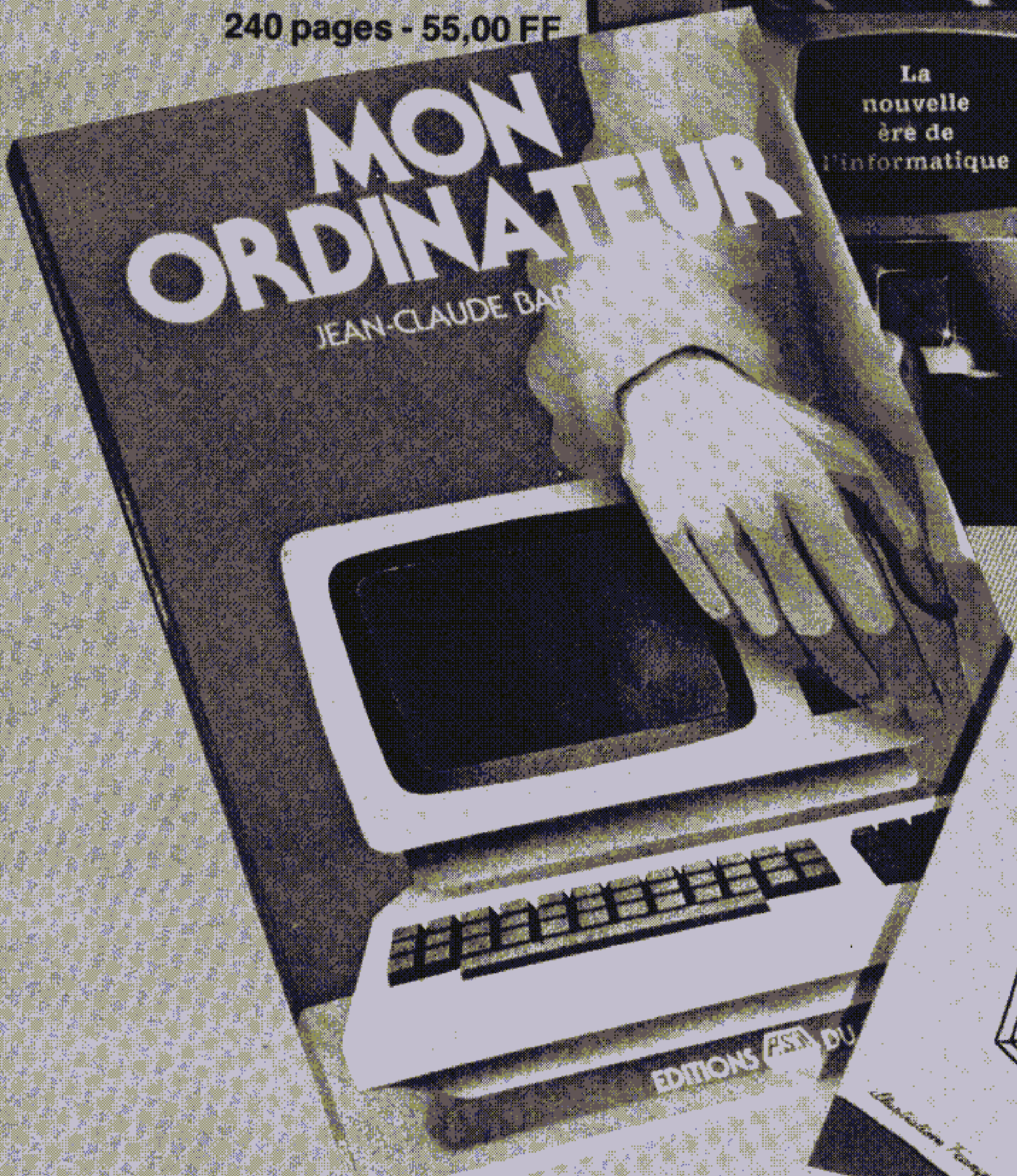
Comment éteindre et rallumer à volonté l'affichage, et lui seul, de votre micropoche ?

L'ordinateur individuel
 La nouvelle ère de l'informatique
 Par Yves Leclerc
 Ecrit par un journaliste canadien,
 L'ordinateur individuel est,
 à la fois, le roman de l'informatique
 de Pascal à nos jours,
 une réflexion sur les implications
 sociales de l'informatique et
 une présentation de
 l'informatique individuelle, de ses
 applications présentes et futures.
240 pages - 55,00 FF



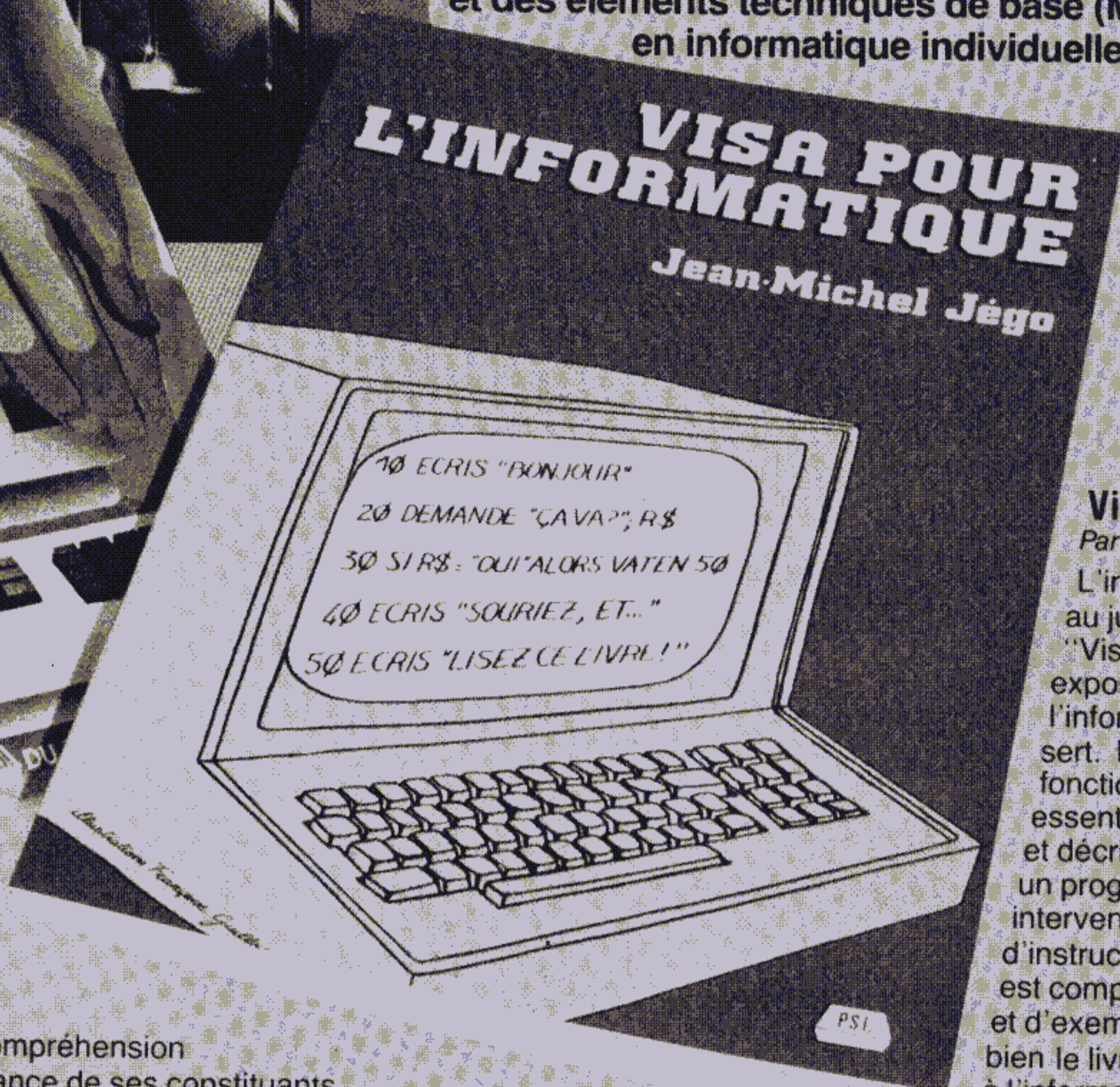
VOUS AVEZ DIT INFORMATIQUE?

Ma voiture, ma chaîne, ma perceuse... mon ordinateur. L'informatique est toujours plus présente et pourtant nombreux sont ceux qui ne savent pas réellement ce qu'est un ordinateur individuel, comment il fonctionne, et quelles sont ses applications. Ces ouvrages jalonnent les trois étapes du chemin qui les fera passer de "l'ignorance" à la connaissance du contexte économique et social (L'ordinateur individuel) des notions fondamentales (Visa pour l'informatique) et des éléments techniques de base (Mon ordinateur) en informatique individuelle.



Mon ordinateur
 Par Jean-Claude Barbance

Comme tout objet technique, l'ordinateur, même individuel, ne peut être abordé sans une compréhension minimale de son fonctionnement et une connaissance de ses constituants. "Mon ordinateur" s'adresse aux non initiés pour leur apporter cette compréhension et ces connaissances, et les aider à choisir un équipement. Très pratique, ce livre est complété par un mini-dictionnaire des 140 termes et abréviations les plus utilisés en informatique, et une étude des prix à l'automne 1981.
128 pages - 60,00 FF



Visa pour l'informatique
 Par Jean-Michel Jégo

L'informatique c'est quoi au juste? "Visa pour l'informatique" expose clairement ce qu'est l'informatique et ce à quoi elle sert. L'auteur définit les fonctions et organes essentiels d'un ordinateur et décrit ce qu'est un programme, en ne faisant intervenir qu'un nombre restreint d'instructions. L'ouvrage est complété d'exercices et d'exemples. "Visa" est donc bien le livre d'initiation à l'informatique tant attendu.
96 pages - 45,00 FF

BON DE COMMANDE

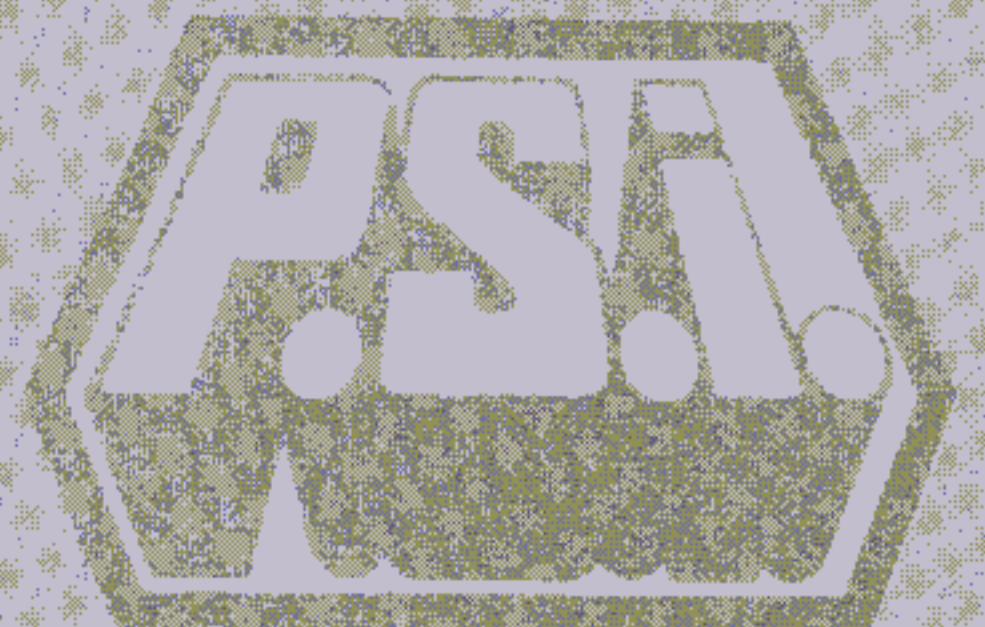
OP 4

Envoyer ce bon accompagné de votre règlement à
EDITIONS DU P.S.I.
 41-51, rue Jacquard BP 86
 77400 Lagny-s/Marne
 Tél. (6) 007.59.31.

DESIGNATION	NOMBRE	RIX
	TOTAL	

Les prix sont : taxes, emballage et port compris.
 (par avion : ajouter 5 FF par livre)

NOM _____ PRENOM _____
 rue _____ N° _____
 Code post. [] [] [] [] [] [] Ville _____



Editions du P.S.I.
 41-51, rue Jacquard
 BP 86
 77400 Lagny-s/Marne
 Téléphone (6) 007.59.31

au Canada : SCE Inc.
 3449 rue Saint-Denis
 Montréal Québec H2X3L1
 Tél. : (514) 843.76.63



Jacques W. est biologiste et dirige le laboratoire d'un hôpital.

Il est médecin et il y a plus de dix ans que les mots *informatique* et *ordinateur* ne lui sont pas étrangers. Il nous a raconté l'itinéraire qui l'a conduit à utiliser des ordinateurs de plus en plus gros, puis à découvrir l'informatique de poche ; il ne se déplace jamais sans son micropoche.

Des ordinateurs de poche pour...

L'Ordinateur de poche : Comment avez-vous fait connaissance avec l'informatique ?

Jacques W. : En réalité, c'est par curiosité. Un jour, il y a une bonne dizaine d'années, mon père m'a montré un livre sur l'informatique, livre que j'ai dévoré. C'était en juin 1970. J'ai tout de suite eu envie de suivre des cours et j'ai, entre autres, appris le Cobol par correspondance. Ensuite j'ai voulu poursuivre et j'ai obtenu un CAP d'informatique en 1972. Inutile de dire qu'à l'époque tout cet enseignement était très livresque.

Utilisez-vous déjà des ordinateurs dans votre travail ?

■ Oui et non. Dans un hôpital, et surtout dans un laboratoire de biologie, les besoins en informations statistiques sont très grands. La précision requise pour les calculs n'est pas vraiment énorme, mais il faut pouvoir traiter à tout moment de la journée les données de l'instant. J'ai donc utilisé divers ordinateurs (le premier nous était obligeamment prêté par un ami banquier) jusqu'à ce que nous nous équipions avec un système Data General que l'on peut qualifier de mini-ordinateur. A côté des applications de contrôle de laboratoire, j'avais des besoins statistiques et en particulier de statistiques non paramétriques : j'ai écrit moi-même les programmes en Fortran.

C'est à cette époque, il y a 4 ans, que j'ai acheté ma première calculatrice programmable, une TI-58. Je l'ai très vite changée pour une TI-59 car j'avais besoin de pouvoir enregistrer les programmes que j'utilisais très fréquemment. A partir de cet instant, elle ne m'a plus quitté — ou plutôt j'ai toujours eu un micropoche dans ma serviette.

En effet, il y a un an et demi, en découvrant le Sharp PC-1211, j'ai eu le coup de foudre à cause du BASIC. Je me suis dit tout de suite : " Avec celui-là, je peux programmer tout à mon aise ; c'est bien plus simple ".

Utilisez-vous toujours votre TI-59 ?

■ Non, ma fille qui est en première me l'a prise et elle s'en sert pour ses cours. Quant à moi, j'ai réécrit mes programmes sur le PC-1211 qui ne me quitte plus.

Travaillant habituellement avec des ordinateurs d'une taille plus imposante, n'êtes-vous pas déçu par votre micropoche ?

■ Certainement pas. Il a ses défauts bien sûr, mais il a surtout beaucoup d'avantages. Quand je me déplace ou quand je rentre chez moi avec du travail, je l'ai toujours à portée de la main. Naturellement, je regrette sa lenteur et surtout son système de variables. Mais la solution est simple : il suffit d'être organisé et de structurer toujours ses programmes de la même manière. Ainsi les variables de fonctionnement vont toujours de A et Z et lorsque j'ai besoin de variables indicées, elles ne sont jamais inférieures à la vingt-septième : A(27).

Quelles sont les principales applications qui vous sont utiles ?

■ Dans le cadre de mon travail ce sont essentiellement des programmes statistiques (Chi-2, Fisher

**Il y a finalement
une dernière application
que j'ai découverte récemment :
on peut jouer avec un ordinateur.**

**Pour les statisticiens
Chi-2 (9 x 9) sur PC-1211**

On garde les mémoires A à Z comme indices divers et on met le tableau et les totaux à partir de A(27). On réserve une fois pour toutes 81 mémoires pour les données et 2x9 mémoires pour les totaux, même si toutes ne sont pas utilisées.

	I										TL
J	27	28	29	30	31	32	33	34	35	108	
	36	37	38	39	40	41	42	43	44		
	45	46	47								
	54										
										105	106
										107	116
Tc	117									125	T

C est le nombre de colonnes, L le nombre de lignes et K la valeur calculée. Le Chi-2 se trouve en fin de programme dans W.

```

10:REM CHI-2*9X: 170:NEXT I
    9            180:NEXT J
20:PAUSE "CHI-2: 190:REM CALCUL-I
    9*9"        U-CHI2
30:CLEAR        200:FOR J=1TO L
40:Y=107:Z=116  210:FOR I=1TO C
50:INPUT " NBR.  220:F=27+((J-1)+
    LIGNES ? ";L  9)+(I-1)
60:INPUT " NB.C: 230:G=Y+J
    OL.? ";C      240:H=Z+I
65:REM GARNISSA 250:K=A(G)*A(H)/
    GE           T:REM VALEUR
70:FOR J=1TO L   -CALCULEE
80:PAUSE USING   255:IF K<5PAUSE
    "##"#"LIGNE  "L";J;"C";I;
    NO ";J        "<5"
90:FOR I=1TO C   260:X=ABS (A(F)-
100:F=27+((J-1)*  K)-.5
    9)+(I-1)      270:X=X*X/K
110:INPUT " VALE: 280:W=W+X:REM CH
    UR ? ";A(F)   I-2
120:G=Y+J        290:NEXT I
130:H=Z+I        300:NEXT J
140:A(G)=A(G)+A( 310:PRINT "***CH
    F)           I-2 :";USING
150:A(H)=A(H)+A( "###".###";W
    F)           320:GOTO 30
160:T=T+A(F)     330:END

```

etc.). Pour mes besoins de gestion domestique, j'ai d'abord un ensemble de programmes pour la gestion de mon portefeuille d'actions (chargement et affichage de la bourse, mise à jour des cours d'achat, suivi du portefeuille...). D'autre part je peux suivre le budget familial en ventilant les dépenses par postes (nourriture, maison, voitures, habillement, loisirs, etc.). Et il y a finalement une dernière application que j'ai découverte tout récemment : on peut jouer avec un ordinateur et j'ai ainsi écrit divers programmes de jeux. C'est par ce biais d'ailleurs que j'ai découvert les ordinateurs individuels et je me suis acheté il y a deux mois un TRS-80 modèle III sur lequel je travaille à la maison avec ma fille - mes deux autres fils se contentant d'utiliser des programmes déjà préparés. Ils se bloquent un peu, comme beaucoup de gens, parce que les mots des langages de programmation sont en anglais.

Vous utilisez visiblement de nombreuses formes différentes d'informatique, et en matériel, et en applications. Etes-vous le seul dans l'hôpital ?

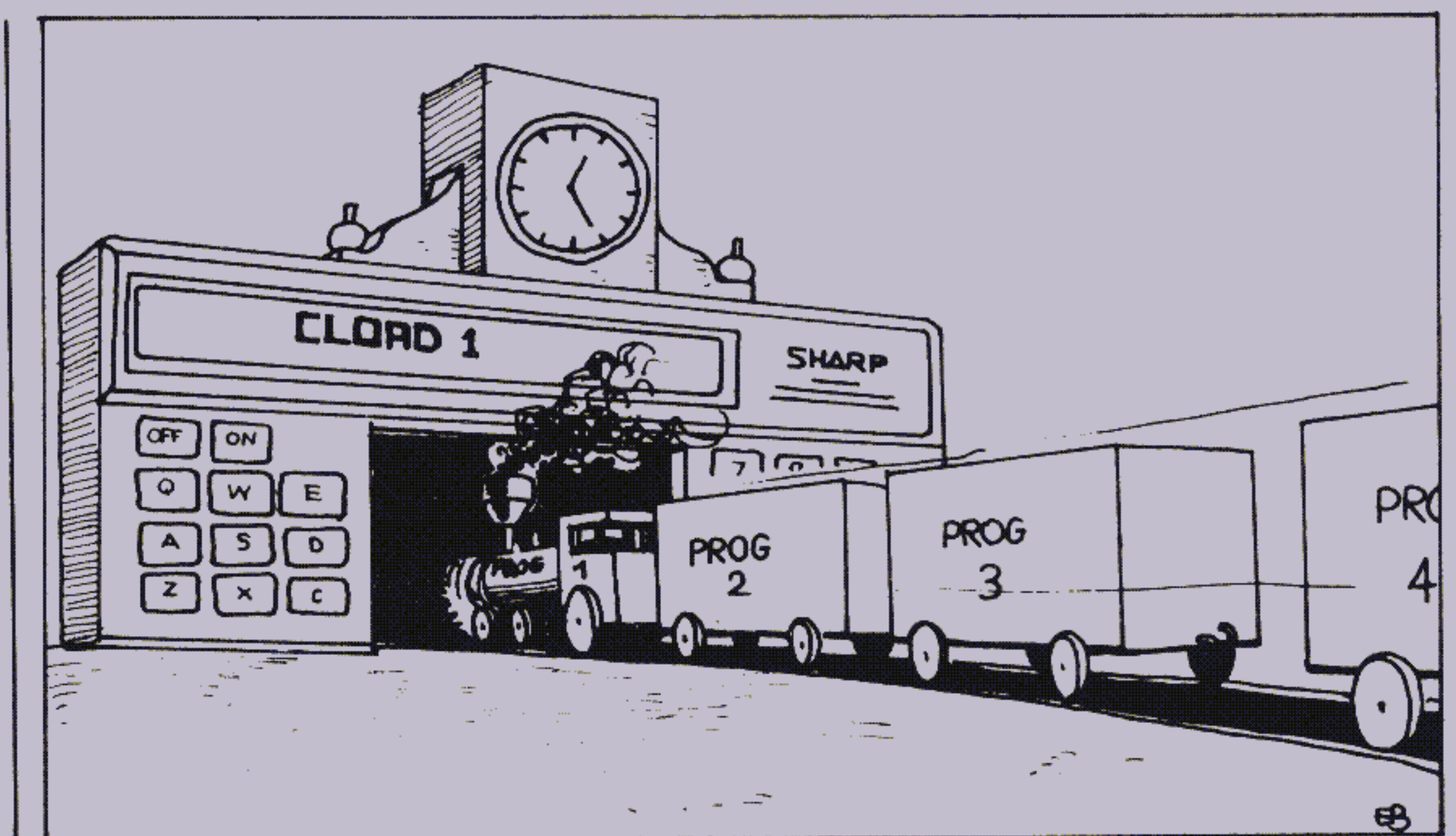
■ Non, de plus en plus de gens possèdent qui une HP, qui une TI, qui une Sharp. Plusieurs ordinateurs sont par ailleurs installés. Certaines personnes utilisent des Apple, et nous utilisons un CBM pour les analyses automatiques. D'ici 3, ou (soyons prudent) 5 ans, tout le monde utilisera un ordinateur dans l'hôpital.

Pensez-vous que cela se fera facilement ?

■ Oui, car chacun voit les nombreux services que cette machine nous rend dès aujourd'hui. Il faut apprendre bien sûr, mais il ne m'est par exemple jamais arrivé d'avoir un problème et de ne pas pouvoir le résoudre.

Je ne pense pas être un jour sans micropoche dans ma serviette. Pour moi c'est vraiment mon "ordinateur de serviette".

Propos recueillis par Jean-Pierre Brunerie



CLOAD 1 : des programmes à la queue leu leu

Nous avons vérifié sur un des premiers TRS 80 pocket vendus en France et sur un PC-1211 très récent que la commande CLOAD 1 fonctionne bel et bien. Profitons-en pour faire le tour de la question.

■ Le manuel de Sharp est curieusement muet sur la commande CLOAD 1. Elle est absente également du memento cartonné qui est livré avec le TRS de poche, mais elle est décrite aux pages 95 à 97 du manuel de Tandy (versions anglaise et française).

Cette commande permet de charger un programme depuis une bande magnétique et de l'inscrire dans la mémoire du micropoche à la suite de celui (ou de ceux) qui s'y trouve déjà. Grâce à elle, il est donc possible de faire cohabiter dans la mémoire du PC 1211 ou du TRS de poche plusieurs programmes différents les uns après les autres, même

s'ils ont en tout ou partie les mêmes numéros de ligne ! Bien utilisée, elle permet donc d'accroître considérablement les possibilités de la machine.

Selon toute apparence, il est impossible de faire exécuter CLOAD 1 par programme (cela conduit régulièrement à l'affichage d'une erreur n° 1). Il faut donc la demander manuellement et elle peut d'ailleurs, comme la plupart des instructions du BASIC Sharp, être abrégée sous la forme CLO. 1 ou CLOA. 1. A la différence de l'instruction CHAIN (qui peut être exécutée par programme), CLOAD 1 est donc une commande au même titre que CLOAD, CLOAD ? et CSAVE. Elle doit être, elle aussi, obligatoirement suivie du nom (sept caractères au plus) qui identifie le programme sur la cassette, faute de quoi la machine signale une nouvelle erreur de syntaxe (1.....).

On peut l'utiliser indifféremment lorsque le micropoche est en mode PRO, en mode RUN ou en mode DEF. On peut encore, en mode RESERVE assigner à cette commande l'une ou plusieurs des 18 touches préprogrammables de l'ordinateur (A : CLOAD 1 " PRG-1 ", par exemple).

Sans parler des facilités qu'elle offre et dont chacun — j'en suis certain — saura tirer parti, il est intéressant de détailler un peu cette fonction, car elle permet de mieux comprendre la façon dont le micro-poche utilise les numéros de lignes des programmes.

Après de patients essais, j'en suis arrivé à la conclusion que la commande CLOAD 1 est strictement identique à CLOAD à une différence près : elle n'efface pas la " mémoire-programme ". Lorsqu'elle est exécutée, le programme enregistré sur la cassette est entré en mémoire à la suite de celui qui s'y trouve déjà, et ce chargement s'effectue sans aucune vérification du numéro des lignes. Autrement dit, les programmes sont mis bout à bout, enchaînés (pour faire chic, on dit " concaténés "). Remarquons à ce propos que l'on peut ainsi mettre à la queue leu leu autant de programmes que l'on veut tant qu'il reste de la place disponible en mémoire.

Peu importe que les numéros de lignes ne soient pas rangés par ordre croissant ou même que plusieurs lignes portent le même numéro. Le programme n° 1 (qui a été obtenu par une série de CLOAD 1 " P1 " et

Exemple 1 : c'est bien d'un programme qu'il s'agit, et il tourne aussi normalement qu'un autre.

CLOAD 1 " P2 ") montre que toutes les lignes d'un programme peuvent très bien porter un seul et même numéro. Malgré ses anomalies, ce programme est listé aussi bien que les autres et il fonctionne parfaitement : il affiche ou il imprime sur la CE-122 " P1 " puis " P2 " à 15 reprises.

Nous verrons plus loin ce que cela entraîne lorsque l'exécution du programme passe par un branchement (GOTO ou GOSUB) à un numéro de ligne. Contentons-nous pour l'instant de remarquer qu'un programme de ce genre fait ce que l'on attend de lui et qu'il est possible de le sauver sur cassette, sous le nom de " P3 " par exemple, et de le réintroduire en mémoire à la suite de lui-même grâce à CLOAD 1 " P3 ".

De drôles de numéros

On parvient, avec le même procédé, à obtenir des programmes qui se déroulent par numéro d'ordre décroissant (exemple n° 2). Il suffit de sauver d'abord la ligne 100 sous un nom quelconque (choisissons P1), puis la ligne 90 sous le nom de P2, la ligne 80 sous le nom de P3 et ainsi de suite. Dans un second temps, on efface la mémoire du programme en faisant NEW, on rembobine la cassette et, sans quitter le mode PRO, on demande CLOAD " P1 ", puis CLOAD 1 " P2 ", CLOAD 1 " P3 ", etc. Le programme ainsi chargé sera sauvé à son tour et rechargé à volonté sans difficulté sous le nom de " mons-tre " par exemple.

De tout cela, on conclura avec raison que l'interpréteur BASIC ne tient pas compte des numéros de lignes : il les exécute comme si de rien n'était les unes après les autres telles qu'elles sont stockées en mémoire.

On remarquera d'autre part que ces suites bizarres de lignes sont listées normalement dans leur intégralité et dans l'ordre où elles sont en mémoire, grâce à l'ordre LIST. Toutefois, avec l'exemple que nous avons pris, seule la première peut être listée par son numéro ; autrement dit, dans notre cas, LIST 100

```
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
10:PRINT "P1"
10:PRINT "P2"
```

```
100:PRINT "LIGNE
  100"
90:PRINT "LIGNE
  90"
80:PRINT "LIGNE
  80"
70:PRINT "LIGNE
  70"
60:PRINT "LIGNE
  60"
50:PRINT "ETC."
```

fonctionne, mais LIST 90 provoque l'affichage d'un message d'erreur n° 2 (erreur de ligne).

En sens inverse, on peut modifier ou détruire tout ou partie du **dernier** des programmes qui a été chargé grâce à CLOAD 1, mais cela est impossible pour les programmes chargés antérieurement. C'est ainsi qu'il est vain d'essayer de supprimer ou de modifier la ligne 100 (la première rentrée) de notre programme.

Toute tentative pour modifier la ligne 100 conduira en fait à créer une nouvelle ligne 100 qui viendra s'insérer à sa place dans le dernier programme entré, en l'occurrence après la ligne 50, mais la première ligne 100 n'en sera pas détruite pour autant !

Vous vérifierez également que si l'on crée une ligne dont le numéro

est inférieur à toutes celles qui sont déjà dans la mémoire du micropoche, cette ligne vient tout de même s'insérer à la tête de la liste. C'est ainsi que 5 REM ENTER va se placer tout au début du programme, comme 49 REM ENTER, tandis que 51 REM ENTER ira tout à la fin, après la ligne 50.

Cela nous apprend que lorsque l'on crée une ligne ou lorsqu'on en modifie une, le Sharp/Tandy parcourt sa mémoire en commençant **par la fin** et va de ligne en ligne jusqu'à trouver la première ligne dont le numéro est égal ou inférieur à celle qui vient d'être rentrée. Il a donc été conçu de telle sorte qu'il ne fonctionne bien que si les lignes sont rangées dans le bon ordre, ce qui précisément n'est pas toujours le cas quand on utilise la commande CLOAD 1...

A titre d'exercice, je vous propose de rentrer, grâce à votre magnétophone, le " programme " suivant dans votre micropoche :

```
10 : REM 10
20 : REM 20
15 : REM 15
17 : REM 17
```

Une fois que la chose est faite (c'est l'affaire de quelques minutes), vous pourrez vérifier que :

- si vous tapez une ligne 5, elle se place avant la ligne 10,
- si vous tapez une ligne 16, elle se

Exemple 2 : numérotation des lignes en ordre décroissant.

place entre les lignes 15 et 17,

- si vous tapez une ligne 12, elle se place entre les lignes 10 et 20,
- si vous tapez une ligne 10, elle remplace la ligne 10,
- si vous tapez une ligne 18 (ou 50, ou 385, etc.), elle se place après la ligne 17.

Dans ces conditions, on peut se demander ce qui se produit lorsque l'on utilise des GOTO ou des GOSUB dans un programme dont une partie a été chargée par CLOAD 1. Pour le savoir, nous commencerons par obtenir en mémoire la suite de lignes :

```
10 : GOTO 100
100 : PRINT " A " : END
100 : PRINT " B " : END
```

RUN et RUN 100 conduisent à l'affichage de " A " (c'est la première des deux lignes 100 qui est

Comment faire ?

Pour utiliser la commande CLOAD 1, on procèdera de la sorte :

- sauver sur cassette (CSAVE " xxx ") le programme qui fera l'objet de la commande CLOAD 1,
- faire NEW,
- introduire au clavier ou par CLOAD " yyy " un autre programme,
- rembobiner la cassette jusqu'au début de l'enregistrement du programme " xxx ",
- demander, en mode RUN, PRO ou DEF, CLOAD 1 " xxx ", ENTER.

Le programme " xxx " est alors introduit depuis la cassette dans la mémoire du micropoche à la suite du programme " yyy ".

N.B. Pour des raisons de clarté, tous les exemples que nous avons retenus ici portent sur des programmes d'une seule ligne. Bien entendu, la commande CLOAD 1 permet de mettre bout à bout des programmes de plusieurs lignes.

```
exécutée). Faisons maintenant NEW
et entrons un nouveau programme :
100 : PRINT " A " : END
100 : PRINT " B " : END
150 : GOTO 100
100 : PRINT " C " : END
```

Cette fois-ci, RUN conduit encore à l'affichage de " A ", mais RUN 150 en revanche affiche " B ". D'où l'on peut en déduire que les recherches de branchements par GOTO s'effectuent depuis l'instruction (ici ligne 150) vers le début ou la fin du programme selon que le numéro de la ligne où doit se faire le branchement est plus petit ou plus grand que celui de la ligne où se trouve inscrit le GOTO. On constate encore une fois que le micropoche a été conçu pour fonctionner avec des programmes dont les lignes sont rangées par numéro d'ordre croissant.

```
150 : GOTO 100
100 : PRINT " A "
```

conduit à une erreur n° 2 (erreur de ligne) et il n'y a pas à s'en étonner : ne trouvant aucune ligne 100 avant la ligne 150, il signale une erreur. De la même façon,
150 : PRINT " A "
100 : GOTO 150

provoque le même résultat quand on demande RUN 100. Le branchement devant s'effectuer à une ligne dont le numéro est plus élevé que 100, l'ordinateur ne recherche pas dans le début de la liste, mais après la ligne 100, et il ne trouve rien.

Tout semble indiquer qu'il en va de même pour GOSUB, et je vous laisse le soin de vous en convaincre vous-même. Tous ces renseignements vous permettront, je l'espère, d'utiliser sans surprise la commande CLOAD 1. Vous verrez à l'usage qu'elle peut rendre de grands services.

□ Antoine Vaussy-Lesbaudy

système

A l'intérieur d'un micropoche... le destin des nombres

Les ordinateurs de poche ont une façon bien particulière de traiter les nombres. Le plus souvent, on ne soupçonne pas ce qu'ils peuvent subir ! Une petite enquête s'imposait.

■ En dehors de l'intérêt pratique qu'ils présentent, les codes-barres sont l'expression visible, noir sur blanc, des signaux que « comprend » et utilise un calculateur. Quand on les examine à l'œil nu, il faut avouer qu'à moins d'être sérieusement entraîné, ils sont parfaitement incompréhensibles. A la vérité, je ne connais personne qui se soit exercé à la lecture rapide des codes-barres. Et cependant, le

micropoche s'en arrange très bien.

Cela montre en tout cas que les ordinateurs ont une manière bien à eux de régler leurs comptes avec les chiffres. Comment font-ils ? En attendant un article sur les codes-barres, je vous propose aujourd'hui d'explorer la façon dont les micropoches représentent les nombres. Ce sera l'occasion, en tout bien tout honneur, de vous présenter Mademoiselle la calculatrice.

Présentations très mondaines, c'est-à-dire faites par un programmeur et non par un technicien du fer à souder. Que « ceux qui savent déjà » me pardonnent ce chapitre, tous les lecteurs n'en sont pas à ce stade ; et quand bien même... A quelle époque vivait Pic de la Mirandole ? (Question bleue) (1).

(1) *J'ose à peine poser la première question bleue : qui a inventé la machine à calculer ? C'est comme Marignan, tout le monde connaît.*

Étant donné, en outre, que la demoiselle a une ascendance anglo-saxonne assez marquée, je donnerai de temps à autre le terme anglais correspondant au mot français : mot (anglais : world).

— Quand on essaie —
— de compter —
— sur deux doigts —

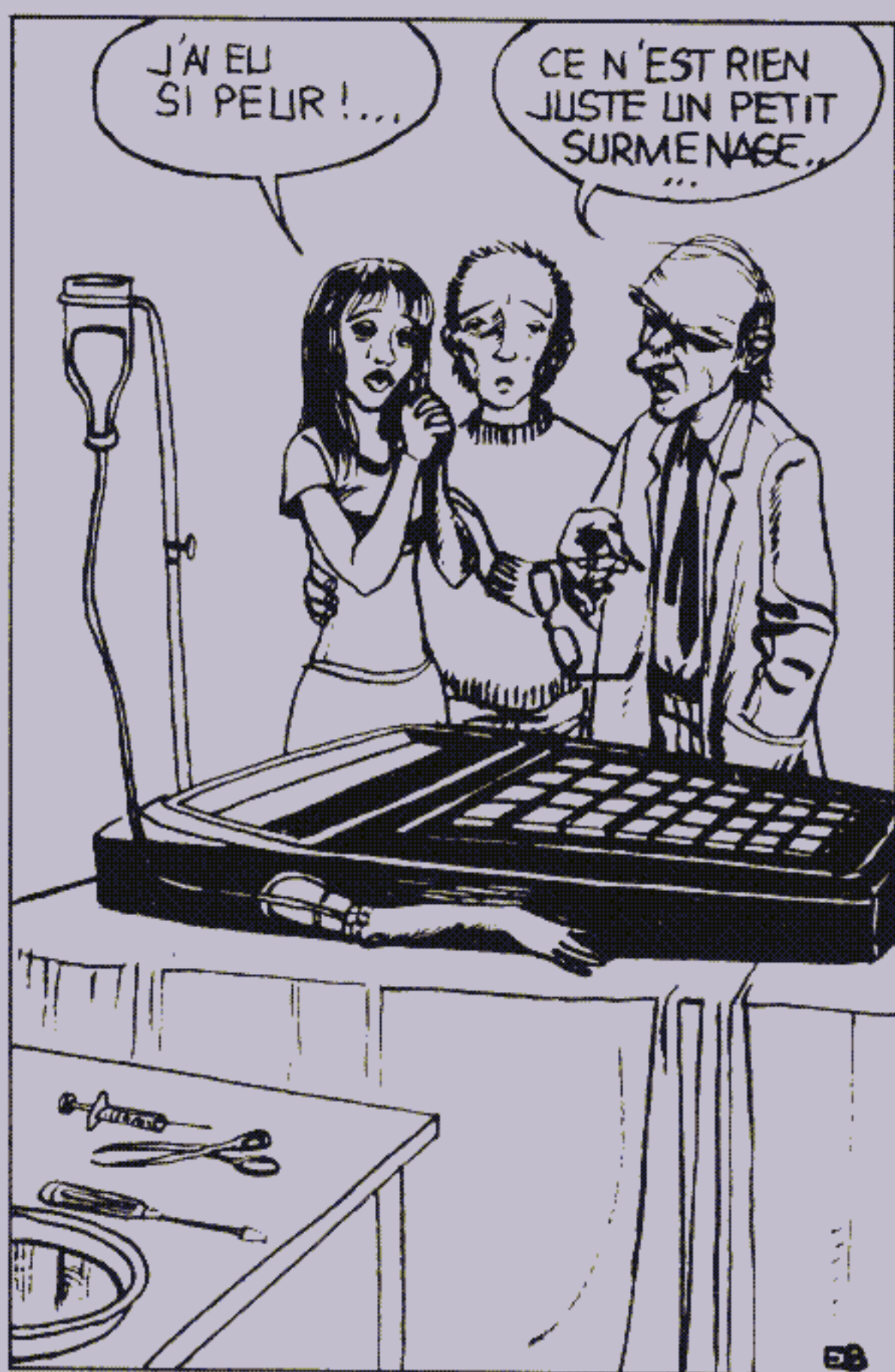
Les circuits électroniques utilisés dans les calculateurs numériques (en anglais : digital computer ; d'où l'adjectif « digital » employé à tort en français ; ceci tend par ailleurs à prouver que les Anglais comptent sur leurs doigts et... l'avouent ! Keep smiling, please) ne savent travailler qu'avec des signaux élémentaires — 1 ou 0. actuellement — réduisant au besoin à cette forme des informations du même type :

• tout ou rien = occupé ou vide = (circuit) conducteur ou non = etc.,

- + ou - (c'est le mode de stockage des mémoires magnétiques, par exemple),
- fort ou faible = long ou bref.

Cette « façon de voir » a donné ses lettres de noblesse à la numération en base 2, dite binaire. Les données, instructions ou autres que l'on a à faire connaître au calculateur sont transcrites sous la forme de codes binaires, suite définie en longueur de signaux 1 ou 0, et plus le choix d'informations possibles est important, plus long est le code permettant de les différencier.

La vocation première du calculateur ayant été de « jouer » avec les chiffres, commençons donc par les



chiffres. Pour des raisons pratiques liées à la notation décimale usuelle, beaucoup d'opérations s'effectuent en système décimal : il convient alors de coder en binaire chaque chiffre décimal, ce qui est réalisé par une suite de 4 informations binaires, le 9 (chiffre le plus élevé de la numération décimale) étant codé en binaire : 1001. Un nombre est donc exprimable par autant de groupes de 4 « bits » qu'il y a de chiffres dans le nombre (on n'y a pas échappé ! « bit » est la contraction de « Binary digIT », chiffre binaire en français ; note 2). 87 est ainsi traduit par $\underbrace{1000}_8 \underbrace{0111}_7$.

On dit alors que le nombre est exprimé en « décimal codé binaire » ou DCB (A : binary coded decimal, or BCD, l'esprit de contradiction des

Anglais étant bien connu ! As well as no one understands why the French persist in writing so many acronyms in reverse order).

Le nombre décimal 87 exprimé en binaire pur s'écrirait 1010111. Doit-on rappeler que, quelle que soit la base b (10 compris), un nombre exprimé sous la forme $a_n a_{n-1} a_{n-2} \dots a_1 \dots a_2 a_1 a_0$ vaut : $a_n \cdot b^n + a_{n-1} \cdot b^{n-1} + \dots + a_1 \cdot b^1 + \dots + a_2 \cdot b^2 + a_1 \cdot b^1 + a_0 \cdot b^0$, l'écriture de a_i s'effectuant à l'aide de symboles allant de 0 à $(b - 1)$, la base b étant toujours représentée par 10_b , et l'indice b précisant — *en décimal* — la base.

Je vois aussitôt plusieurs doigts qui se lèvent et ouvrent la bouche pour me demander : « qu'en est-il des nombres négatifs, et fractionnaires ? » et : « me voyez-vous écrire 8×10^{91} avec un 1 suivi de 365 zéros ? ».

En dehors de l'arithmétique pure, alias « théorie des nombres », les nombres que l'on manipule sont l'expression de grandeurs physiques et il est assez exceptionnel que celles-ci soient connues avec plus de 10 chiffres *significatifs* (3).

Les constructeurs de calculateurs le savent et limitent en conséquence la place accordée à un nombre, avec des variantes :

- chez H.P., sauf quelques cas (en trigonométrie), calculs et résultats avec 10 chiffres significatifs ;
- chez Sharp, calculs avec 13 chiffres mais résultats avec 10 seulement ;
- pour la TI 58 et la TI 59, calculs et résultats avec 13 chiffres.

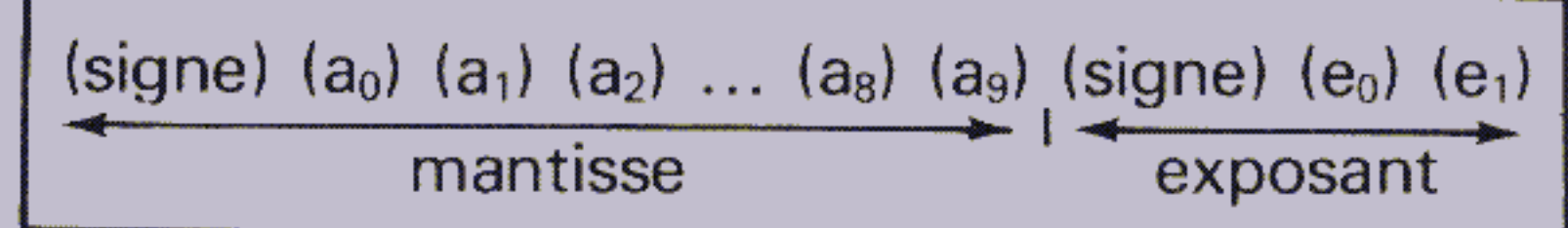
Cela étant dit, ces chiffres ne se suffisent pas : il faut préciser où se trouve la virgule décimale (A : decimal point, or D.P.), c'est le rôle de l'exposant (A : exponent) qui indique de combien de chiffres il faut se

(2) Rassurez-vous, nous avons notre revanche en vitesse puisque l'unité de vitesse ou bit/seconde, s'appelle le baud, du nom (3^e question bleue) d'Emile BAUDOT, ingénieur français inventeur du télégraphe imprimeur.

(3) En astronomie toutefois, des observations faites plusieurs siècles avant notre ère (éclipses...), malgré une certaine imprécision sur l'heure de l'observation, permettent de préciser le 11^e chiffre de certaines données. Il faut dire que les astronomes, non contents d'être de remarquables physiciens, sont de redoutables mathématiciens doués d'un esprit de déduction peu commun... d'où le 11^e chiffre.

(4) Question blanche : qui est considéré comme le père des calculateurs modernes ? Charles Babbage.

déplacer par rapport à la droite du premier chiffre significatif (celui de gauche) pour placer la virgule... en rajoutant des 0 sur la gauche (exposant négatif) ou sur la droite si l'on dépasse le nombre de chiffres significatifs (exposant positif). Se trouve ainsi définie la notation scientifique, appelée aussi semi-logarithmique, et cette analogie avec les logarithmes a fait étendre à cette notation le terme de *mantisse* (A : mantissa) donné aux chiffres significatifs. Il ne reste plus qu'à placer le signe de la mantisse et celui de l'exposant. On obtient ainsi un nombre parfaitement standardisé qui s'insère dans le moule prévu par le constructeur. Voici par exemple le moule H.P. :



chaque valeur entre parenthèses étant exprimée par un groupe de 4 bits ; le tout sans aucune séparation forme ce que l'on appelle une chaîne (A : string). Vous pouvez remarquer au passage que le constructeur a aussi limité le nombre de chiffres de l'exposant : d'où le refus par le calculateur de continuer à travailler avec des nombres trop grands qui ne rentrent plus dans le moule et qu'il ne saura pas relire ; c'est sans grande incidence pratique, sauf pour les mathématiciens et les statisticiens qui s'amuse avec la fonction Factorielle par exemple.

Beau gaspi, me direz-vous, dans le cas des signes qui sont codés sur 4 bits alors qu'ils représentent une information typiquement binaire et donc codable avec un seul bit. C'est vrai pour le signe de l'exposant, mais non pour celui de la mantisse : nous allons voir pourquoi.

Nous n'avons parlé jusqu'à présent que du codage des nombres, mais il y a encore les instructions, les chaînes alphanumériques, etc. ; or vous ne disposez pas d'autre moyen de raconter ce que vous voulez à votre machine (4) que ces 1 et ces 0 qui constituent tout son alphabet. Il faut donc lui indiquer — et le plus tôt c'est le mieux — comment elle doit comprendre ce que vous lui dites : comme un nombre, une instruction, un caractère... Cela, elle l'apprend grâce au premier groupe de 4 bits ; c'est ainsi que si ce groupe est différent de 0 (nombre positif) ou 9 (nombre négatif ; le signe est donc codé à 4 bits et non un seul) dans la chaîne qui occupe

A l'intérieur d'un micropoche... le destin des nombres

le registre X, elle refuse d'effectuer une opération mathématique sur une chaîne qu'elle considère alors comme alphanumérique, et elle affiche un message d'erreur (ALPHA DATA pour la HP 41) ; vous ne pourrez donc pas lui faire extraire la racine de "MANIOC" ou de "NAVET", ou trouver une tangente à un "CERCLE", en toutes lettres.

————— L'octet : 256 —————
————— combinaisons —————
————— possibles —————

Nous n'avons jusqu'à présent considéré que des groupes de 4 bits, parce qu'ils suffisaient pour exprimer chiffres et nombres. Mais 4 bits ne permettent de coder que $2^4 = 16$ "trucs" différents, ce qui est peu eu égard à ce que l'on peut attendre d'un calculateur et à ce que l'on lui a appris à faire. Comme il sait déjà découper une chaîne en tranches de 4 bits, on lui demande de regrouper ces tranches par paires : nous disposons donc de codes de 8 bits, soit $2^8 = 256$ "instructions" possibles — ce qui devient convenable —, et l'on trouve l'unité de compte bien connue qui est l'octet (A : byte).

Ces octets sont ensuite regroupables par paquets de 7 (chez H.P.) ou 8 (chez T.I. ou SHARP) pour occuper des registres (A : register).

L'apparition des code-barres va vous permettre de modifier vos relations avec la Demoiselle. Jusqu'à maintenant vous ne disposiez que d'un interprète, le clavier, qui se chargeait de traduire vos phrases et veillait comme un chaperon à la bienséance de vos relations ; si vous voulez pousser le marivaudage un peu plus loin, les codes-barres vont vous procurer le moyen de, vous exprimer directement dans le langage de la demoiselle en vous dispensant du truchement : vous obtiendrez alors de la calculatrice ce qui vous sied.

Rassurez-vous, d'une part nul ne vous oblige à sortir du domaine autorisé par le truchement, mais

Algorithmes de conversion des nombres entiers :

● Décimal en binaire (le nombre binaire s'élabore à partir de la droite) diviser le nombre par 2
si la partie fractionnaire n'est pas nulle (c'est-à-dire si le nombre était impair) poser 1, sinon poser 0, garder la partie entière, et recommencer le processus jusqu'à ce que la partie entière soit nulle.

● Binaire en décimal (c'est comme le remontage des armes : en sens inverse du démontage). Prendre le chiffre binaire de gauche, le multiplier par 2 et lui ajouter le chiffre qui est à sa droite... jusqu'à épuisement.

Ex. : $13_{10} \div 2 = 6,5$ pose 1 et garde 6
 $6 \div 2 = 3,0$ pose 0 et garde 3
 $3 \div 2 = 1,5$ pose 1 et garde 1
 $1 \div 2 = 0,5$ pose 1 et garde 0 => Fin

A l'inverse, 1101_2 : écrivons-le verticalement

1	x 2 = 2
+ 1	= 3 x 2 = 6
+ 0	= 6 x 2 = 12
+ 1	= 13

c'est la preuve par 9, si l'on peut dire.

Algorithme humainement praticable pour les nombres décimaux jusqu'à 4 chiffres.

d'autre part si l'envie vous en venait, apprenez que seuls des gestes maladroits sont susceptibles de provoquer chez mademoiselle HP 41 soit un certain affolement, soit un étourdissement passager, soit, si vous manquez de chance, une léthargie plus ou moins prolongée.

Pour vous exprimer dans ce nouveau langage, vous ne pourrez plus utiliser uniquement le vocabulaire SIN, COS, LOG, etc. dont vous vous servez habituellement et que le truchement traduit ; il vous faudra encore savoir énoncer le code de l'instruction que vous voulez transmettre. Soit, par exemple, le code (sur un octet) : 10100111_2

● Vous pouvez l'énoncer comme s'il s'agissait d'un nombre décimal : "dix millions cent mille cent onze" ; il se réécrit facilement, mais il est long à énoncer ;

● Vous pouvez encore, puisqu'il s'agit d'un "nombre" binaire, le convertir en décimal (de 0 à 255) : 167 ; trois chiffres et tout est dit, mais la reconversion en binaire n'est plus immédiate ;

● Vous pouvez aussi le scinder en 2 demi-octets (A : nybble) $1010/0111$ et énoncer chaque demi-octet comme ci-dessus. Cela donne mille dix/cent onze ou 10/7 mais aussi, pour l'écrire, 10-7 ou 10,7 ou 1007

Ainsi décrit, l'octet occupe entre la moitié et un peu plus du tiers de sa place primitive.

Les programmeurs et autres techniciens préfèrent, quant à eux, utiliser un code sur mesure, le code hexa (ou hexal) — abréviation (ou contraction) de hexadécimal c'est-à-dire à base 16. Comme le système décimal ne fournit de symboles que de 0 à 9, ils les complètent par le début de l'alphabet (en majuscules) : A pour 10 jusqu'à F pour 15. Ainsi, le code ci-dessus s'énonce A7 : la description de l'octet n'occupe plus que 2 caractères.

Question blanche en forme d'exercice : exprimer en binaire puis en décimal 5EB.

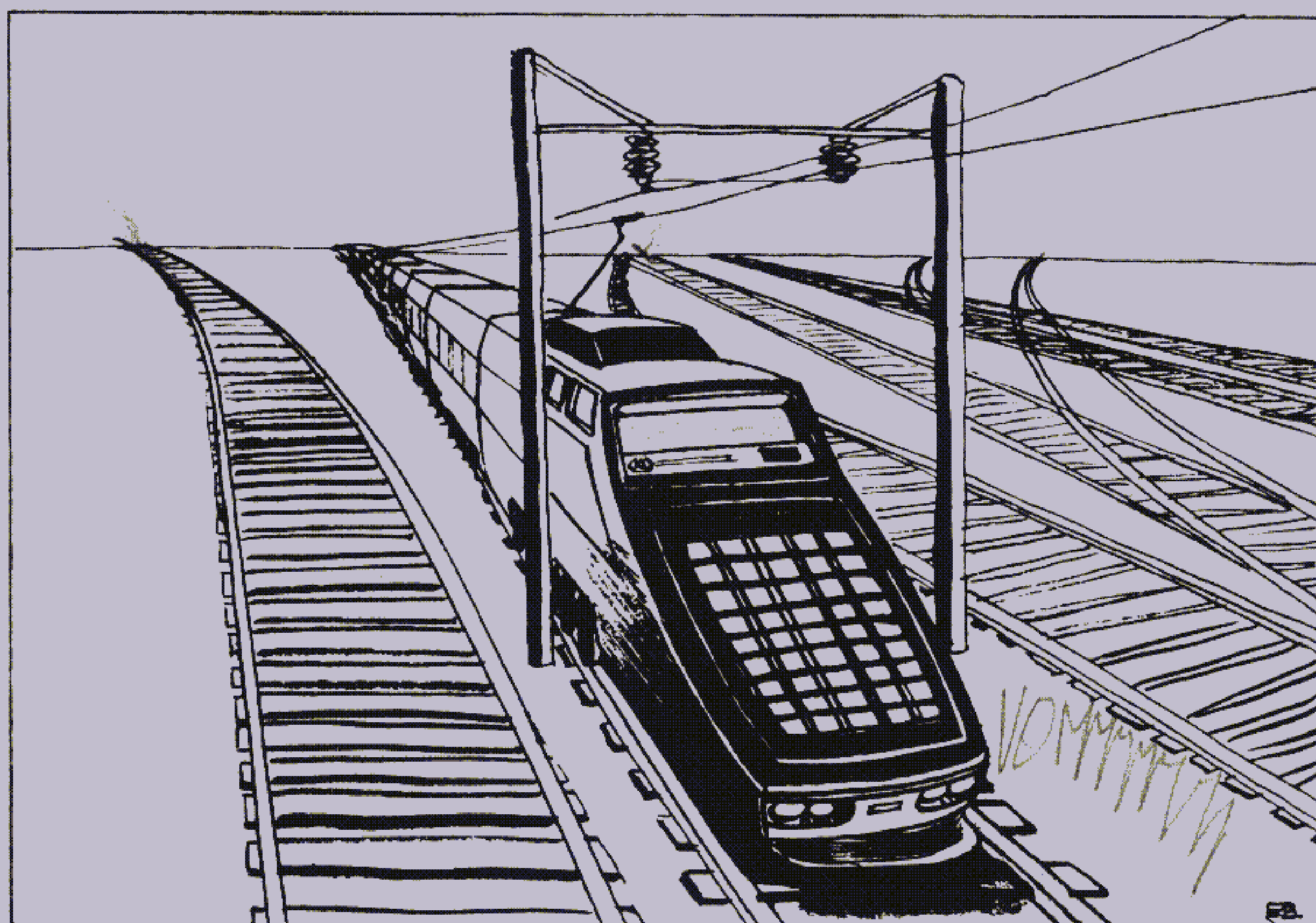
Signalons que l'apprentissage du code hexa(l) n'a rien de sorcier et que, de toute façon, comme la littérature technique y fait souvent appel, il n'est pas mauvais de l'ajouter à l'arsenal de ses connaissances.

Notons enfin l'existence de l'octal qui découpe un nombre binaire en tranches de 3 bits et convertit en décimal chaque tranche de 3 bits, n'utilisant donc les chiffres que de 0 à 7. Cette numération est comode dans certains cas : citons par exemple l'introduction de données d'un jeu sur tableau (1=pion, 0=vide) où cette numération per-

met d'entrer en un seul nombre de 10 chiffres la position de 30 pions. Par contre, elle est pratiquement inutilisée pour énoncer le contenu d'un octet, 3 étant premier avec 8.

A ce point de la présentation, prenons un air un peu bovin pour regarder un instant passer les trains. Un calculateur peut en effet être considéré comme un circuit de chemin de fer (miniature, échelle "puce" (A : "chip" scale). Sur ce circuit fermé circulent à se toucher un certain nombre de trains : chacun comporte des voitures (registres) dont les compartiments (octets) à 8 places sont occupés par un certain nombre de voyageurs occupant des places (bits) réservées ; dans le premier train voyagent des éléments que le personnel chargé du réseau doit pouvoir rapidement et facilement retrouver (exemple HP : registres de la pile, registres alphanumériques, drapeaux, adresses de retour des sous-programmes, etc.) ; dans le train suivant figurent les instructions du programme ; dans le dernier train figurent les données, numériques ou autres. Admettons que seule la motrice du premier train soit parfaitement repérable. Ces trains défilent sans discontinuer et à vitesse déterminée devant un poste de contrôle ; le surveillant qui s'y trouve dispose également de quelques petites voies de manœuvre armées par du personnel qualifié. Vivons quelques microsecondes de la vie de ce réseau.

Le pointeur signale que la prochaine instruction à exécuter occupe le 6^e compartiment de la 4^e voiture du train "Programme", lequel compartiment passera quelques microsecondes après le passage de la motrice du premier train. A l'heure d'horloge prévue donc, le compartiment susdit passe devant le poste de contrôle ; le surveillant prend une photo des passagers et l'interprète : c'est l'instruction LOG ; le surveillant sort alors d'un tiroir le mode de calcul d'un LOG (placé là par les ingénieurs avant qu'il ne prenne son poste), il copie au passage le contenu de la voiture "X" et envoie le tout sur une voie de manœuvre pour le calcul de LOG X ; c'est là que l'on vérifie que X est bien un nombre et qu'il est positif non nul — l'inverse rendrait le personnel fort perplexe —, et que l'on calcule LOG X ; une fois prêt, on attend le passage de la voiture "L" (Last x) pour en vider les passagers et y installer ceux de la voiture "X" qui cèdent la place à LOG X ; puis l'on



attend le passage du 7^e compartiment de la 4^e voiture du train "Programme" pour connaître l'instruction suivante, etc. Et tout se déroule au rythme de la diabolique petite horloge. A la vérité c'est un train d'enfer.

Tout ceci est une image simplifiée, bien sûr, mais l'horloge, le pointeur, les voies de manœuvre, les modes de calcul... existent, et ceci est bien une (petite) partie du travail qu'effectue le microprocesseur.

Question rouge : sujet de dissertation. "Prolonger la pensée de l'auteur" en expliquant la mémoire continue, le rôle de .END., de SIZE, PACK... (non limitatif).

———— Le jeu ————
 ——— des 1 000 octets : ———
 ——— les réponses ———

Première question bleue : si vous ne connaissez pas la réponse, envoyez-moi d'urgence tous vos billets portant l'effigie de Blaise Pascal, vous n'êtes pas digne de les posséder !

Deuxième question bleue : il devient de plus en plus difficile d'être un Pic de la Mirandole, qui avait la réputation de tout savoir. Qui, par exemple, peut m'apporter la réponse à la question suivante : pourquoi un angle obtus est plutôt dans le genre ouvert alors qu'un esprit obtus est dans le genre fermé ?

Première question blanche : si Charles Babbage a bien défini le principe de fonctionnement des calculateurs et construit un calculateur,

Un circuit miniature de train électr(on)ique : le microprocesseur.

il a fallu attendre l'apparition de la triode, inventée par Lee De Forest en 1906, pour se libérer des contraintes de la mécanique et pouvoir commencer à développer des systèmes "tout électr(on)iques", sans inertie, sans retards (ou presque), etc. ; puis celle des transistors suivie de celle des circuits intégrés pour faire tenir dans votre poche ce qui, il y a seulement 30 ans, tenait à peine dans le Panthéon.

Deuxième question blanche :
 $5EB = \frac{10111101011_2}{5 \ E \ B} = 1515_{10}$

Question rouge : je vous laisse ces sujets de dissertation pour d'éventuels devoirs de vacances.

Questions banco :

- avez-vous appris quelque chose ?
- avez-vous parfois eu envie de sourire ?

Si vous répondez "non" à l'unanimité, je suis désolé... de vous dire que vous avez perdu les 1.000 octets du banco. Et mille excuses — et merci — à Lucien Jeunesse, "s'il le veut bien".

Pour moi, avant de quitter ce costume de scène que j'ai pris le parti de revêtir pour vous dire ce prologue, je peux vous assurer qu'il a été sérieusement écrit et qu'il comporte juste la dose supportable d'inexactitudes et d'approximations pour que l'arbre ne cache pas la forêt.

□ Jean Thiberge

Le BASIC et les chaînes de caractères

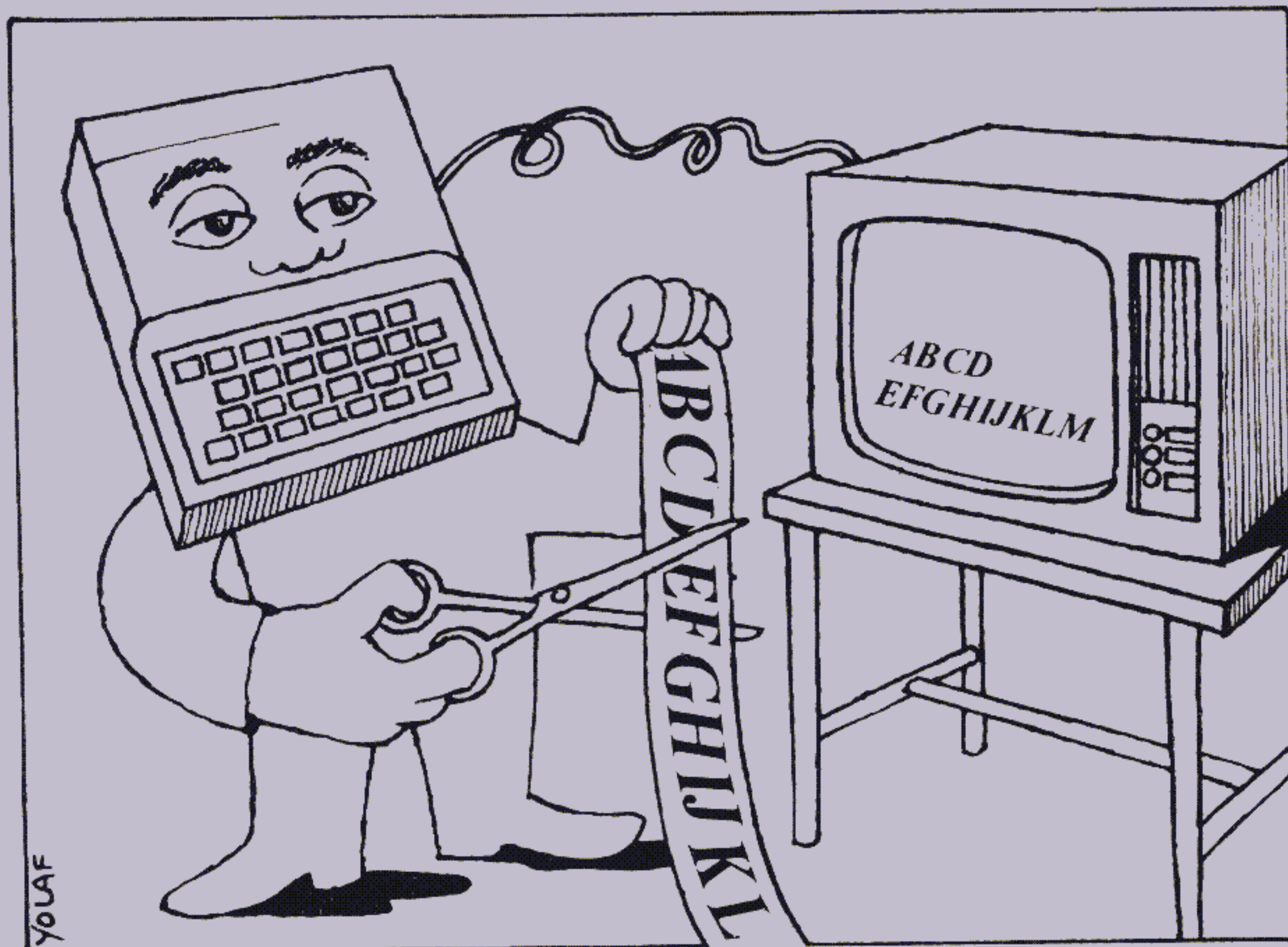
Il y a, c'est bien connu, les chiffres et les lettres. Ce sont deux choses bien différentes. Les ordinateurs ne les confondent pas. Voyons comment le BASIC s'en arrange.

■ Le langage BASIC, puisque c'est de lui qu'il s'agit aujourd'hui, est conçu pour traiter séparément les valeurs numériques et les caractères alphanumériques. Commençons, judicieusement, par définir ce qu'est une variable alphanumérique.

D'aucuns diront (définition a contrario) que « c'est une variable qui n'est pas numérique ». C'est exactement cela, mais cela ne nous avance pas beaucoup. Soyons plus précis : une variable alphanumérique est un ensemble couramment appelé *chaîne* (en anglais *string*) qui peut contenir des lettres, des signes de ponctuation, des caractères variés (% , \$, Π , # , etc.) parmi lesquels figurent entre autres les dix chiffres.

Attention cependant : quand ils font partie d'une chaîne alphanumérique, les chiffres ne sont pas interprétés comme représentant des valeurs numériques, mais comme des signes, au même titre que les lettres M ou R, par exemple, ou que le point d'interrogation.

Dans tous ces signes, il y en a un qui fait exception : le guillemet. Il est en général impossible de l'introduire dans une chaîne alphanumérique BASIC, car il est justement utilisé pour délimiter les chaînes de caractères. Lorsque l'ordinateur rencontre un guillemet pour la première fois, il y reconnaît un message qui lui signale que la chaîne alphanumérique va débuter avec le caractère



suivant. Et quand il rencontre un second guillemet, il l'interprète comme le signe que la chaîne alphanumérique vient de se terminer. On dit que le guillemet est un caractère séparateur. On ne peut donc pas mettre un guillemet entre guillemets...

Certains BASICs, particulièrement bien fournis en instruction de traitement des chaînes, permettent toutefois d'y parvenir, mais ce n'est pas courant. Sur les ordinateurs de poche les plus répandus, cette opération n'est pas prévue et il faut se livrer à de véritables prouesses (quand on y arrive) pour entrer un guillemet dans une chaîne alphanumérique. La chose est possible par exemple sur les PC 1211 et les TRS 80 Pocket qui répondent au sésame (1), mais il faut le faire.

Comme chacun le sait, il est vain d'essayer de multiplier un mot par un autre, ou d'en rechercher le

carré. Les opérations que l'on peut faire sur les chaînes alphanumériques diffèrent beaucoup de celles qui s'appliquent aux valeurs numériques. Les « manipulations » de chaînes (c'est ainsi qu'elles s'appellent) constituent un ensemble très efficace du langage BASIC. On peut couper, souder, comparer les chaînes, déterminer quel est le caractère qui occupe telle ou telle position à l'intérieur d'une chaîne, le remplacer par un autre, etc.

Malgré quelques différences de syntaxe (il y a presque autant de variantes du BASIC que de modèles d'ordinateurs...), on peut décrire rapidement le traitement des chaînes en examinant de ce point de vue les deux ordinateurs de poche qui utilisent le BASIC et dont on parle le plus en ce moment : le PC 1211/TRS 80 pocket d'une part et le Casio FX 702 P d'autre part.

Pour élargir un peu le panorama, nous avons joint à cette étude deux autres ordinateurs BASIC : un poids plume, le ZX 81 de Sinclair et un

(1) Sur le sésame, voir l'Op n° 1, pages 63 à 66.

ordinateur de table, le TRS 80 modèle 1 niveau 2. Leurs possibilités nettement supérieures nous permettront de compléter notre tour d'horizon du traitement des chaînes en BASIC.

Le tableau présenté ci-contre récapitule les principales fonctions alphanumériques. La première ligne indique la longueur maximale d'une chaîne alphanumérique. C'est évidemment un point de comparaison très important. On s'aperçoit ainsi que le ZX 81 et le TRS 80 modèle 1 sont dotés de variables alphanumériques pouvant contenir chacune jusqu'à 255 caractères (tout un poème...), alors que les variables du PC 1211 ne comptent que 7 caractères (plutôt télégraphique); même remarque pour le FX 702 P à cette différence près qu'il possède en plus une variable (\$) dont la capacité est de 30 caractères.

D'une façon générale, les variables alphanumériques sont spécifiées par le suffixe \$: A\$, B\$, etc. Comme on l'a déjà dit, on peut comparer deux chaînes, et en premier lieu pour vérifier si elles sont ou non identiques. Ce type de comparaison est très utile pour le contrôle d'une entrée de données et il conduit à des programmes tels que :

```
10 INPUT A$
20 IF A$ = « OUI » THEN GOTO 50
30 PRINT « AU REVOIR »
40 END
50 PRINT « NOUS SOMMES D'ACCORD »
```

etc.

ce qui revient à dire en français : (ligne 10) entrez une chaîne au clavier ; (ligne 20) si cette chaîne est « oui », alors le programme se poursuit à la ligne 50 pour afficher « nous sommes d'accord », sinon il se poursuit à la ligne 30 avec l'affichage d'« au revoir » et s'achève à la ligne 40 (fin).

C'est à cette comparaison d'égalité entre des chaînes d'au plus 7 caractères qu'est limité le PC 1211/TRS 80 pocket. Si cet ordinateur a ses qualités propres par ailleurs, ses fonctions de manipulations de chaînes sont réduites à l'état embryonnaire. Nous en avons dit l'essentiel et nous n'y reviendrons plus.

Sur le BASIC des autres machines, la panoplie des opérations que l'on peut effectuer sur les chaînes alphanumériques est beaucoup plus riche. C'est ainsi que les comparaisons ne portent plus sur la seule

égalité, mais aussi sur la supériorité, l'infériorité, la différence (>, <, <>, >=, <=). Ce qui fait l'objet de la comparaison, dans ce cas-là, ce n'est pas la longueur des chaînes,

chaîne. L'argument V, obligatoirement mis entre parenthèses, peut être une variable alphanumérique (A\$ par exemple) ou une concaténation de chaînes (A\$ + B\$ + C\$);

	PC 1211 et TRS 80 pocket	Casio FX 702 P	Sinclair ZX 81	TRS 80 Modèle 1
Longueur de chaîne	7	7 + une mémoire \$ de 30 caractères	255	255
Comparaison d'égalité	OUI	OUI	OUI	OUI
Comparaison <>, <, <=, >, >=	NON	OUI	OUI	OUI
Concaténation (A\$ + B\$)	NON	OUI	OUI	OUI
LEN	NON	OUI	OUI	OUI
MID\$	NON	OUI	A\$ (xTOy)	OUI
LEFT\$ et RIGHT\$	NON	NON	NON	OUI
STR\$	NON	NON	OUI	OUI
VAL	NON	NON	OUI	OUI

mais leur valeur en code ASCII pour le FX 702 P et le TRS 80 modèle 1, et leur valeur dans une codification non standard pour le ZX 81.

En fin de compte, et quel que soit le code, c'est l'ordre alphabétique qui est utilisé. Si l'on compare l'une à l'autre les chaînes « AB » et « AC », on s'aperçoit que la première est plus petite que la seconde, autrement dit qu'« AB » se place avant « AC » dans un classement par ordre alphabétique. Et le mécanisme de cette comparaison est simple : le code ASCII de B est 66 (39 dans le code utilisé par le ZX 81), le code ASCII de C est 67 (40 pour le ZX).

Avec un tel système, il devient très facile de classer une liste de chaînes par ordre alphabétique et d'y effectuer des recherches.

On peut également mettre bout à bout plusieurs chaînes alphanumériques et conserver la chaîne qui en résulte dans une variable. Cette opération que l'on appelle la concaténation est représentée par le symbole +.

```
10 A$ = « ORDINA »
20 B$ = « TEUR DE »
30 C$ = « POCHE »
40 D$ = A$ + B$ + C$
50 PRINT D$
```

affiche « ORDINATEUR DE POCHE »

La fonction LEN (V) fournit le nombre de caractères composant la

dans notre dernier exemple, la valeur de LEN (D\$) serait 19.

Le plus souvent, cette fonction est utilisée conjointement avec une autre, MID\$ (chaîne, p, l) qui fournit le fragment de chaîne de longueur l commençant au caractère p. C'est ainsi que E\$ = MID\$ (D\$, 5, 15) rangera dans la variable alphanumérique E\$ le mot poche.

Le micropoche Casio FX 702 P possède cette fonction qui, en l'occurrence, s'appelle MID et non pas MID\$, mais il ne permet de l'utiliser que sur la chaîne contenue dans la seule mémoire \$ qui peut compter jusqu'à 30 caractères. Il est donc inutile avec cet ordinateur de spécifier sur quelle variable va porter la fonction MID, et l'on écrit simplement MID (p, l). Evidemment, il s'agit là d'une sérieuse limitation, mais c'est mieux que rien, et en « jonglant » un peu avec la variable \$, on parvient tout de même à de jolis résultats.

Sur le ZX 81, les choses se passent encore de manière différente : on ne trouve pas la fonction MID\$, mais CODE A\$ (p) qui fournit le code (c) du caractère situé à la position p de la chaîne A\$. Une fois ce code numérique obtenu (38 pour le A, 39 pour le B, 40 pour le C, etc.), on lui applique la fonction CHR\$ (c) qui effectue le travail inverse puisqu'elle fournit le caractère dont le code numérique est c. Autre pos-

Le BASIC et les chaînes de caractères

sibilité très précieuse du ZX 81 : A\$(2 TO 5) extraira la sous-chaîne comprise entre le 2^e et le 5^e caractère de la chaîne A\$.

Présentes seulement sur le TRS 80 modèle 1 (mais on arrive à se débrouiller sans elles grâce à MID, MID\$ ou leurs équivalents), les fonctions LEFT\$(chaîne, l) et RIGHT\$(chaîne, l) s'apparentent à MID\$(chaîne, p, l). Dans ces deux fonctions, le paramètre p est implicitement constitué dans le premier cas par l'extrémité gauche (en anglais *left*) de la chaîne et dans le second cas par son extrémité droite (anglais *right*).

Il existe encore deux autres fonctions très importantes que l'on trouve à la fois dans le BASIC du TRS 80 modèle 1 et dans celui du ZX 81.

La première est la fonction STR\$(V) : elle permet de convertir une valeur numérique (avec notre exemple, c'est celle qui est contenue

dans la variable numérique V) en une chaîne de caractères alphanumériques. Voyons, pour illustrer cette fonction, le petit programme suivant :

```
10 PRINT STR$(A)
20 PRINT STR$(B)
30 PRINT STR$(A + B)
40 PRINT STR$(A) + STR$(B)
```

Si la variable A contient la valeur 10 et la variable B la valeur 15, l'exécution des lignes 10 et 20 conduit à l'affichage de 10 et de 15 qui sont, remarquons-le, devenus des chaînes de caractères. La ligne 30, quant à elle, affiche 25, c'est-à-dire la somme de A et de B convertie elle aussi en chaîne de caractères.

La ligne 40 enfin effectue une concaténation des deux chaînes « 10 » et « 15 » et conduit donc à l'affichage de 1015.

Cette transformation des valeurs numériques en chaînes alphanumériques a de multiples applications. La fonction inverse, qui permet de convertir les chiffres faisant partie d'une chaîne en leur valeur numérique, est également très utile. Elle s'écrit VAL(chaîne).

Si la chaîne spécifiée contient à la fois des chiffres et des lettres, seule

la première portion de chiffres rencontrée fait l'objet de la conversion. Ainsi, lorsque A\$ contient « 100 FRANCS » la séquence P = VAL(A\$) rangera dans la variable numérique P la valeur 100.

Comme on a pu le voir dans tous les exemples précédents, l'ensemble des fonctions alphanumériques est un outil très puissant. A la lecture du tableau récapitulatif, on voit ce dont sont capables dans ce domaine précis les quatre ordinateurs passés en revue. On remarquera que le 702 P est assez riche de possibilités. En revanche, le PC 1211 est presque absent du tableau : à cet égard, il est très démuné, même si, par ailleurs, son BASIC présente certaines particularités rares. Il est vrai qu'il n'a pas été conçu pour cela.

Quant au ZX 81, il peut aller la tête haute. Malgré certaines entorses au BASIC standard (il n'utilise pas le code ASCII par exemple, et il possède certaines fonctions *suigeneris*), il dispose de tout ce qu'il faut pour manipuler sans trop de complications les chaînes de caractères.

□ Jean-Pierre Richard

1 MICRO + 1 METHODE = le Basic enfin chez vous

UN ENSEIGNEMENT SUR MESURE :

1 MICRO
1 METHODE
PEDAGOGIQUE
SPECIFIQUE

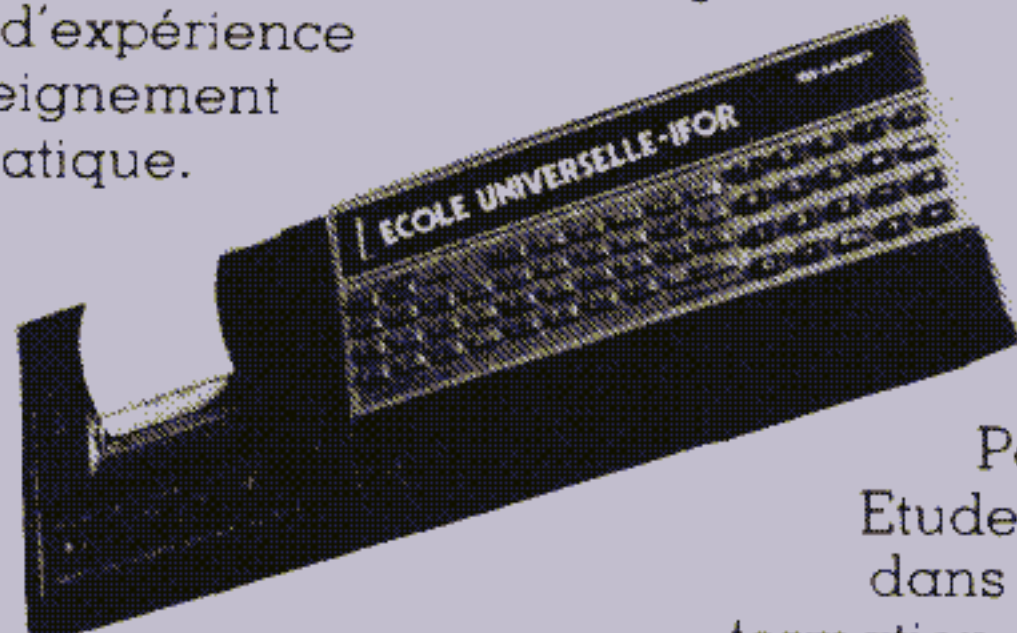
- ➔ ● 1 micro-ordinateur sharp PC 1211 fourni (ou non si vous en possédez un). Possibilité Interface ou Imprimante.
- ➔ ● Notions fondamentales (si vous ne possédez pas de connaissances en Informatique)
- ➔ ● Un cours complet de basic (accès direct si notions en informatique).
- ➔ ● Une assistance pédagogique permanente.
- ➔ ● De nombreux sujets de composition — Plus de 200 exercices sur machine.

APPRENDRE - RAPIDEMENT - EFFICACEMENT - A SON RYTHME - PAR CORRESPONDANCE

ECOLE UNIVERSELLE-IFOR - 28, rue Pasteur 92551 Saint-Cloud Cedex. Tél. : 771.91.19

Etablissement privé d'enseignement à distance

15 années d'expérience
dans l'enseignement
de l'Informatique.



Possibilité
Etude gratuite
dans le cadre
formation continue

Bon pour une documentation gratuite N935

Nom, prénom :
Adresse :
Niveau d'études Age :
désire recevoir une documentation gratuite sur le cours Initiation/basic.

ECOLE UNIVERSELLE-IFOR - 28, rue Pasteur 92551 Saint-Cloud Cedex. Tél. 771.91.19

Votre HP 41 C est-elle en avance ou en retard sur son temps ?

01*LBL "REVEIL"	
02*LBL A	
03 "H DE REVEIL?"	
04 PROMPT	
05 STO 01	
06 1 E4	
07 *	_ Rappel des secondes de l'heure de réveil
08 2	
09 /	
10 FRC	
11 X#0?	_ Si le nombre de secondes est impair, on en soustrait une de façon à ce que l'on puisse trouver une égalité entre l'heure de réveil et l'heure courante (qui sera elle aussi paire)
12 .0001	
13 ST- 01	
14 SF 05	_ Utilisation du réveil
15*LBL C	
16 FIX 0	
17 CF 29	
18 "H DE DEPART?"	
19 PROMPT	
20 STO 00	
21 25	_ Si le réveil ne doit pas servir, on stocke 25 heures comme heure de sonnerie : le programme ne sonnera jamais
22 FC?C 05	
23 STO 01	
24 X<>Y	
25 1 E4	
26 *	
27 2	
28 /	
29 FRC	_ On rend pair le nombre de secondes de l'heure courante
30 X#0?	
31 .0001	
32 ST- 00	
33*LBL 00	
34 2] Temporalisation
35 LOG	
36 RDN	
37 RDN	
38 RDN	
39 RDN	
40 RDN	
41 RCL 00	
42 .0002	
43 HMS+	_ (ou HMS -) ajouter (ou soustraire) 2 secondes
44 FIX 4] Eliminer les décimales parasites dues à l'approximation lors du calcul avec HMS+ ou HMS-
45 RND	

Size 002

La HP 41 C permet de compter quantité de choses. Essayons aujourd'hui de lui faire compter les secondes. Tout le problème est d'obtenir que le programme ne soit ni trop rapide, ni trop lent.

■ Quand on est jeune, il y a une occupation très appréciée qui consiste à démonter avec un *gros* tournevis un réveille-matin hors d'usage qui était promis à la poubelle.

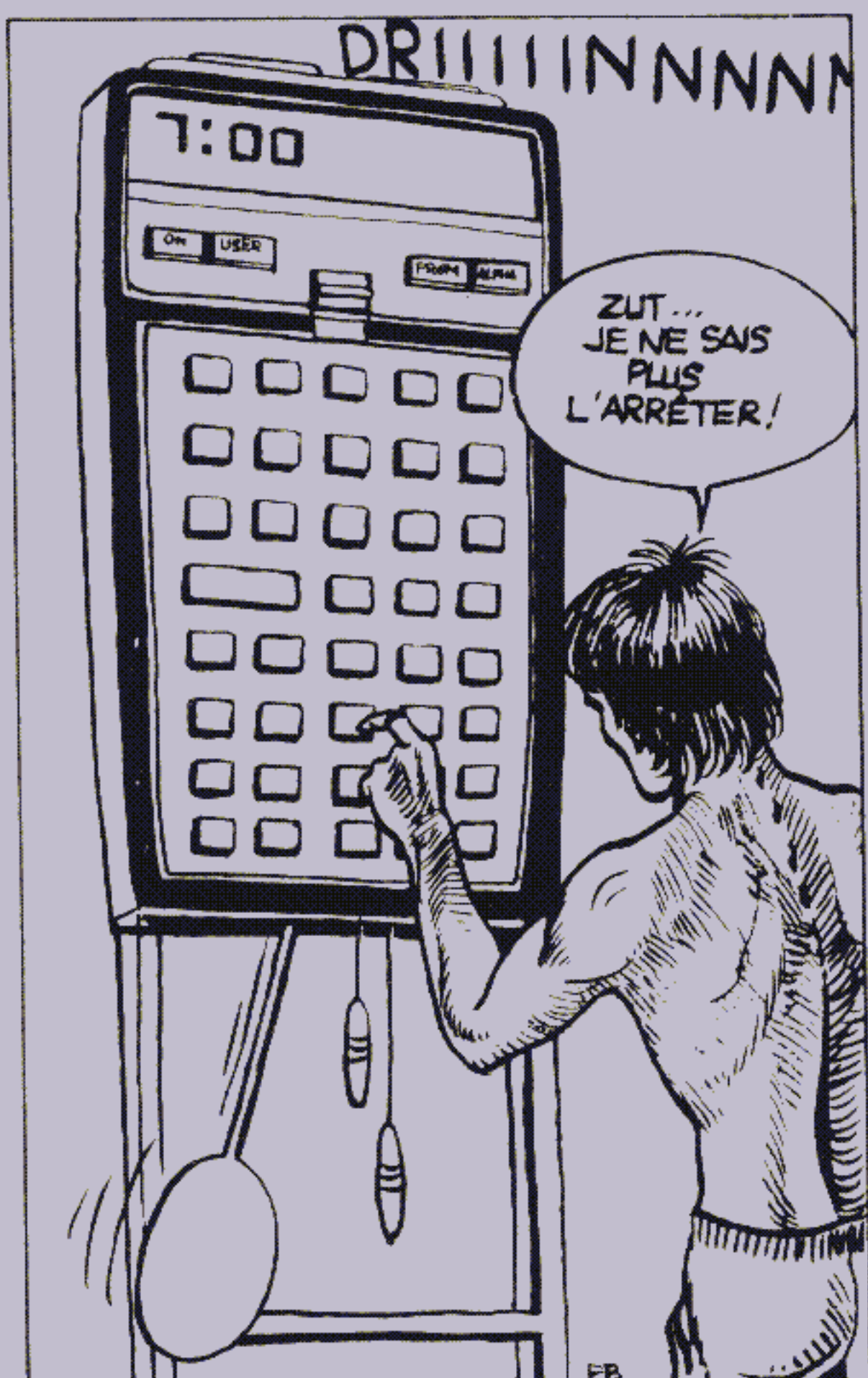
Il arrive également que l'on se livre (clandestinement) à la même opération mais sur une toute petite montre en état de marche en se promettant que l'on parviendra à la reconstituer de façon impeccable. On est alors certain que personne ne s'apercevra que l'on est passé par là. Le plus souvent, on apprend à cette occasion que l'horlogerie est un métier.

Quoi qu'il en soit, on découvre ce jour-là des ressorts, des engrenages, un type d'échappement, etc.

Avec les montres électroniques, le démontage ne présente plus l'attrait d'une découverte de la mécanique car on a vite fait d'isoler les différents éléments : l'afficheur à cristaux liquides, le processeur, la pile, les contacts. Le remontage n'est pas plus facile, mais le plus souvent on n'a rien appris du tout sur le fonctionnement de la petite machine.

Comme le démontage est en général fatal à la montre et qu'il ne présente pratiquement plus d'intérêt pédagogique, il est à déconseiller formellement.

Votre HP 41 C est-elle en avance ou en retard sur son temps ?



Pour bien faire, il faut adopter la démarche inverse : il faut s'efforcer de réaliser de toutes pièces une montre électronique aussi précise que possible en partant d'un micro-poche. Peu importe que le programme utilisé ne soit pas vraiment celui qui tourne dans les montres du commerce, c'est le résultat qui compte et ce que l'on apprend pour y parvenir.

Si la montre obtenue a une précision de quelques centièmes de seconde par minute, elle est même utilisable pendant une bonne dizaine d'heures sans inconvénient sérieux.

Pour mener à bien ce programme, je me suis inspiré d'une idée publiée dans le n° 17 de *L'Ordinateur Individuel* (page 111, mai 1980). Cela dit, vous verrez qu'il utilise un cycle de 24 heures et non pas de 12, et qu'il transforme votre HP 41 en montre-réveil avec fonction chronomètre et compte à rebours.

Il y a quatre façons de l'utiliser :
 • **en horloge simple** (sans sonnerie), on demande XEQ C, le micro-poche affiche « H DE DÉPART ? » et il suffit d'introduire l'heure actuelle sous la forme HH.MMSS puis de presser sur la touche R/S ;

l'heure est alors affichée toutes les deux secondes sous la forme HH MM SS ;

• **en réveil**, on demande d'abord XEQ A, puis à la question « H DE RÉVEIL ? », on répond en affichant l'heure à laquelle on veut que le micro-poche sonne, et l'on presse sur R/S ; le micro-poche vous demande alors l'heure qu'il est (même introduction que pour l'horloge simple ; HH.MMSS SUIVI DE R/S), puis l'heure est affichée toutes les deux secondes jusqu'à l'heure dite où la sonnerie se déclenche ;

• **en chronomètre**, il suffit d'indiquer que l'heure actuelle est nulle ; on demande donc XEQ C, et l'on répond 0.0000, R/S ;

• **en compte à rebours**, on doit remplacer l'instruction HMS + du pas 43 par HMS -. Après avoir compacté le nouveau programme

(PACK), on fait XEQ A et l'on répond 0 R/S à la question « H DE RÉVEIL ? ». Le micro-poche affiche alors « H DE DÉPART ? », on indique dans combien de temps l'on veut que la sonnerie se déclenche et l'on presse sur R/S. Le temps restant avant l'échéance est affiché toutes les deux secondes et le micro-poche vous avertit quand le temps imparti est épuisé.

Si cette montre n'est pas assez précise à votre goût, inutile de demander à votre horloger de la régler, il ne pourra rien pour vous, sauf s'il sait programmer...

C'est à vous qu'il revient de retoucher le programme ici ou là jusqu'à ce qu'il vous donne entière satisfaction. Et je vous donne rendez-vous : dès qu'un micro-poche sera doté d'un synthétiseur vocal, nous ferons un programme d'horloge parlante !

□ Antoine Vaussy-Lesbaudy

46 FIX 0	
47 24	
48 X<>Y	
49 X=Y?	Passage de 24 H, 0 mn, 0 sec. à 0 H, 0 mn, 0 sec.
50 0	
51 STO 00	
52 INT	
53 " "	Heures
54 ARCL X	
55 " " "	
56 LASTX	
57 FRC	
58 100	Minutes
59 *	
60 INT	
61 ARCL X	
62 " " "	
63 LASTX	
64 FRC	
65 100	Secondes
66 *	
67 ARCL X	
68 AVIEW	Affichage de l'heure
69 RCL 00	
70 RCL 01	
71 X*Y?	L'heure courante est-elle égale à l'heure de réveil ?
72 GTO 00	Si ce n'est pas le cas, la boucle continue
73 "...DEBOUT...."	
74 AVIEW	
75 SF 25	
76 SF 99	Sonnerie et affichage « original » du message « ...DEBOUT... » qui se déplace à chaque Lbl 01 (astuce piochée dans <i>L'Ordinateur Individuel</i> n° 17, page 112)
77♦LBL 01	
78 TONE 9	
79 GTO 01	
80 .END.	

Le micropoche : une bonne façon d'apprendre à compter

En programmant, sur TI 57 par exemple, l'algorithme de la multiplication ou de la division, l'écopier enseigne en quelque sorte au micropoche ce qu'il faut faire. Par la même occasion, il apprend beaucoup.

■ « Les enfants ne savent plus compter... » (soupir de désespoir). Combien de fois a-t-on entendu pareille antienne ? Les plus pessimistes ajoutent même : « ...et ce n'est pas la calculatrice qui va arranger les choses ! ».

Ils ont tout à fait raison, mais ils ont aussi tout à fait tort (non, ce n'est pas une erreur d'impression !). En fait, une des causes de la non maîtrise du calcul est la mauvaise assimilation des techniques opératoires ; car ne pas savoir compter, c'est :

- soit ne pas connaître ses tables,
- soit ne pas savoir quelle opération faire,
- soit ne pas savoir comment faire l'opération,
- parfois un peu des trois !

C'est sur le troisième point que nous allons nous pencher aujourd'hui : l'apprentissage de méthodes de calcul à l'aide d'un micropoche. Car si la calculatrice « n'arrange pas les choses » quand elle est secrètement utilisée par un « petit malin », elle peut, en revanche, se montrer un auxiliaire précieux pour la mise en œuvre pratique de raisonnements simples, d'où un apprentissage moins scolaire car plus attrayant.

La programmation doit son intérêt pédagogique à deux facteurs essentiels :

- la connaissance parfaite d'un processus est nécessaire au traitement d'un sujet (l'ordinateur ne pardonne pas) ;

• la motivation : tous les bienheureux contaminés par le virus de la programmation connaissent l'ineffable contentement doublé d'un soupir satisfait à la vue d'un de leurs programmes qui « tourne ».

———— Savoir faire ———— ———— l'opération ————

Si vous dites à un adolescent que vous allez lui expliquer comment faire une multiplication ou une division, sa première (et dernière) réaction sera de penser : « Quelle barbe, encore ce truc... ».

Mais s'il se rend compte qu'en

divisant 37 par 43 avec sa calculatrice, il obtient un affichage limité à 8 chiffres (0,8604651) ou qu'en multipliant 6327,19 par 7863,51, il n'obtient que 49753922, ce qui est manifestement incomplet dans les deux cas, il acceptera volontiers l'idée de concevoir des programmes améliorant les performances de sa machine.

La construction d'un tel programme l'obligera à suivre les différentes étapes de l'algorithme (multiplication ou division), un nécessaire retour aux sources en quelque sorte. Nous allons voir comment grâce à deux applications précises.

□ Marc Ferrant

Tous les chiffres d'une multiplication

■ Nous avons rencontré dans l'introduction une multiplication dont le résultat exact ne pouvait pas être indiqué par l'affichage d'une calculatrice, faute de place. En effet $6327,19 \times 7863,51 = 49753921,8369$; l'affichage ne nous indique que les huit chiffres de la partie entière. Nous allons chercher comment obtenir la totalité des chiffres du résultat d'une telle multiplication.

Un exemple simple pour commencer : 47×358 .

47 × 358 ----- 376 235 141 ----- 16826	}	47 × ••8 ----- 376	47 × •58 ----- 376 235 ----- 2726	47 × 358 ----- 2726 141 ----- 168
---	---	-----------------------------	---	---

Cette décomposition permet de comprendre comment nous calculons :

- $47 \times 8 = 376$: 6 est un chiffre du résultat
37 est la retenue
- $47 \times 5 = 235$: On cumule ce résultat à la retenue
 $235 + 37 = 272$
2 est un chiffre du résultat
27 est la nouvelle retenue
- $47 \times 3 = 141$: On cumule ce résultat à la retenue
 $141 + 27 = 168$
8 est un chiffre du résultat
16 est la nouvelle retenue et la suite du résultat puisque la multiplication est terminée.

Etablissons maintenant le tableau 1 qui détaille ces différentes étapes. Cette phase est la conception du programme. Ici $a = 47$ et $b = 358$.

► Tous les chiffres d'une multiplication

Tableau 1

actions	opérations	opérations	opérations
	b = 358	b = 35	b = 3
on divise b par 10	$358 : 10 = 35,8$	$35 : 10 = 3,5$	$3 : 10 = 0,3$
on prend la partie décimale	0,8	0,5	0,3
on garde la partie entière qui donne une nouvelle valeur de b	b = 35	b = 3	b = 0
on multiplie la partie décimale par 10	$0,8 \times 10 = 8$	$0,5 \times 10 = 5$	$0,3 \times 10 = 3$
on multiplie a par ce nombre	$47 \times 8 = 376$	$47 \times 5 = 235$	$47 \times 3 = 141$
on cumule ce résultat avec la retenue précédente	$376 + 0 = 376$	$235 + 37 = 272$	$141 + 27 = 168$
on divise ce total par 10	$376 : 10 = 37,6$	$272 : 10 = 27,2$	$168 : 10 = 16,8$
on prend la partie décimale	0,6	0,2	0,8
on garde la partie entière qui constitue la retenue	R = 37	R = 27	R = 16
on multiplie la partie décimale par 10 : c'est un chiffre du			
Résultat	6	2	8
Et on recommence	↗	↗	

Les chiffres du résultat sont donc obtenus un par un à partir des unités. Lorsque $b = 0$ le dernier reste (16) constitue donc les derniers chiffres de ce résultat. La réalisation du programme est la traduction de la démarche adaptée au langage de la calculatrice.

- a (47) sera introduit dans la mémoire 0,
- b (358) sera introduit dans la mémoire 1,
- pour gagner de la place, 10 sera placé dans la mémoire 6,
- la mémoire 7 contenant zéro permettra de contrôler le moment où b prend la valeur zéro,
- les retenues seront conservées dans la mémoire 2.

On trouvera la liste du programme, commentée, dans le tableau 2.

Tableau 2

Mode d'emploi :

• Initialisation :

- 1 - enregistrer 10 dans R_6 (10 STO 6)
- 2 - enregistrer 0 dans R_7 (0 STO 7)

• Calcul :

- 3 - se placer au début du programme (RST)
- 4 - annuler le contenu de R_2 (0 STO 2)
- 5 - enregistrer a (ici 47) dans R_0 (47 STO 0)
- 6 - enregistrer b (ici 358) dans R_1 (358 STO 1)
- 7 - lancer le programme (R/S) et lire les chiffres : 6, 2, 8, 16, ce qui donne bien 16826. L'arrêt est automatique.

- Pour une autre multiplication, reprendre au n° 3.

Dans le cas de nombres décimaux, introduire ceux-ci sans tenir compte des virgules. Vous placerez la virgule après l'obtention du résultat. On pourra ainsi vérifier que :

$$6327,19 \times 7863,51 = 49753921,8369.$$

Pas	Code	Touches	Commentaires
00	33 1	RCL 1	Rappel de b
01	66	x = t	Contrôle de la valeur de b ($b = 0 ?$)
02	51 1	GTO 1	si $b = 0$ aller au Lbl 1
03	45	÷	} on divise b par 10
04	33 6	RCL 6	
05	85	=	
06	32 1	STO 1	on conserve ce résultat dans R_1
07	- 49	INV INT	on en prend la partie décimale
08	- 34 1	INV SUM 1	on garde la partie entière dans R_1 (nouveau b)
09	55	x	} on multiplie la partie décimale par 10
10	33 6	RCL 6	
11	85	=	
12	55	x	} on multiplie a par ce nombre
13	33 0	RCL 0	
14	85	=	
15	75	+	} on cumule ce résultat avec la retenue
16	33 2	RCL 2	
17	85	=	
18	45	÷	} on divise ce total par 10
19	33 6	RCL 6	
20	85	=	
21	32 2	STO 2	on conserve ce nombre dans R_2
22	- 49	INV INT	on en prend la partie décimale
23	- 34 2	INV SUM 2	on garde la partie entière dans R_2 (retenue)
24	55	x	} on multiplie la partie décimale par 10
25	33 6	RCL 6	
26	85	=	
27	36	Pause	arrêt pour lecture
28	71	RST	on recommence
29	86 1	Lbl 1	traitement du cas $b = 0$
30	33 2	Rcl 2	on rappelle la dernière retenue qui constitue les derniers chiffres cherchés
31	81	R/S	

Tous les chiffres d'une division

■ Dans le même esprit, nous allons maintenant aborder le problème de la division. Nous avons vu dans l'introduction que $37 : 43$ donnait 0,8604651 avec une calculatrice affichant 8 chiffres. En cherchant les trois chiffres internes ($\times 1000 = \text{INV 2nd Int}$), on constate que ce n'est pas fini... (on trouve 0,4651163), et ce n'est qu'un début ! En prolongeant la division on peut espérer trouver la période de la suite décimale de ce rationnel (voir encadré). Pour obtenir cette suite de chiffres, on détaille toutes les opérations qui constituent la réalisation d'une division.

Prenons, par exemple, $37 / 43$:

370	43
260	0,860465...
200	
280	
220	
5...	

37 43	370 43	260 43	20 43	200 43	280 43	220 43
- 0 0	- 344 8	- 258 6	- 0 0	- 172 4	- 258 6	- 215 5
37	26	2	20	28	22	5

Cette décomposition permet de comprendre comment nous calculons. A chaque étape de la division de a par b nous déterminons le quotient entier : $a = b \times q + r$ où q est le quotient entier et r le reste.

- $37 / 43$: 0 est le quotient entier q

- 37 est le reste r
- $370 / 43$: - 8 est le quotient entier q
- 26 est le reste r
- etc.

Si nous savions mieux calculer nous pourrions brûler les étapes. Ainsi, en multipliant le reste par 1000 nous obtiendrions des quotients de 3 chiffres.

37 43	37000 43	20000 43
- 0 0	- 36980 860	- 19995 465
37	20	5

Ici encore, le reste est calculé en effectuant $a - b \times q$

- $37 / 43$: - 0 est le quotient entier q
- 37 est le reste r
- $37000 / 43$: - 860 est le quotient q
- 20 est le reste r
- $20000 / 43$: - 465 est le quotient q
- 5 est le reste r
- etc.

Les chiffres du résultat sont donc obtenus trois par trois après l'obtention de la partie entière (on multiplierait le reste par 1000000 pour obtenir ces chiffres par groupe de 6).

Utilisation des mémoires :

- a (37) sera introduit dans la mémoire 0
- b (43) sera introduit dans la mémoire 1
- pour gagner de la place, 1000 sera placé dans la mémoire 7.

La liste du programme est donnée et commentée dans le tableau 3.

Mode d'emploi :

- **Initialisation :**
 - 1 - enregistrer 1000 dans R_7 (1000 STO 7)
- **Calcul :**
 - 2 - se placer au début du programme (RST)
 - 3 - afficher a (ici 37) (37 R/S)
 - 4 - afficher b (ici 43) (43 R/S)
 - 5 - lire la partie entière, en l'occurrence 0, puis les décimales, par groupes de 3 : 860, 465, 116, 279, 69, 767, 441, 860, 465, etc. Pour arrêter, appuyer sur R/S.

Ainsi, $37 : 43 = 0,860465116279069767441860465...$
- **Pour une autre division**, reprendre au n° 2.

Remarque : si un affichage n'est pas de trois chiffres (69 par exemple), il faut remplacer les chiffres manquant à gauche par des zéros (069).

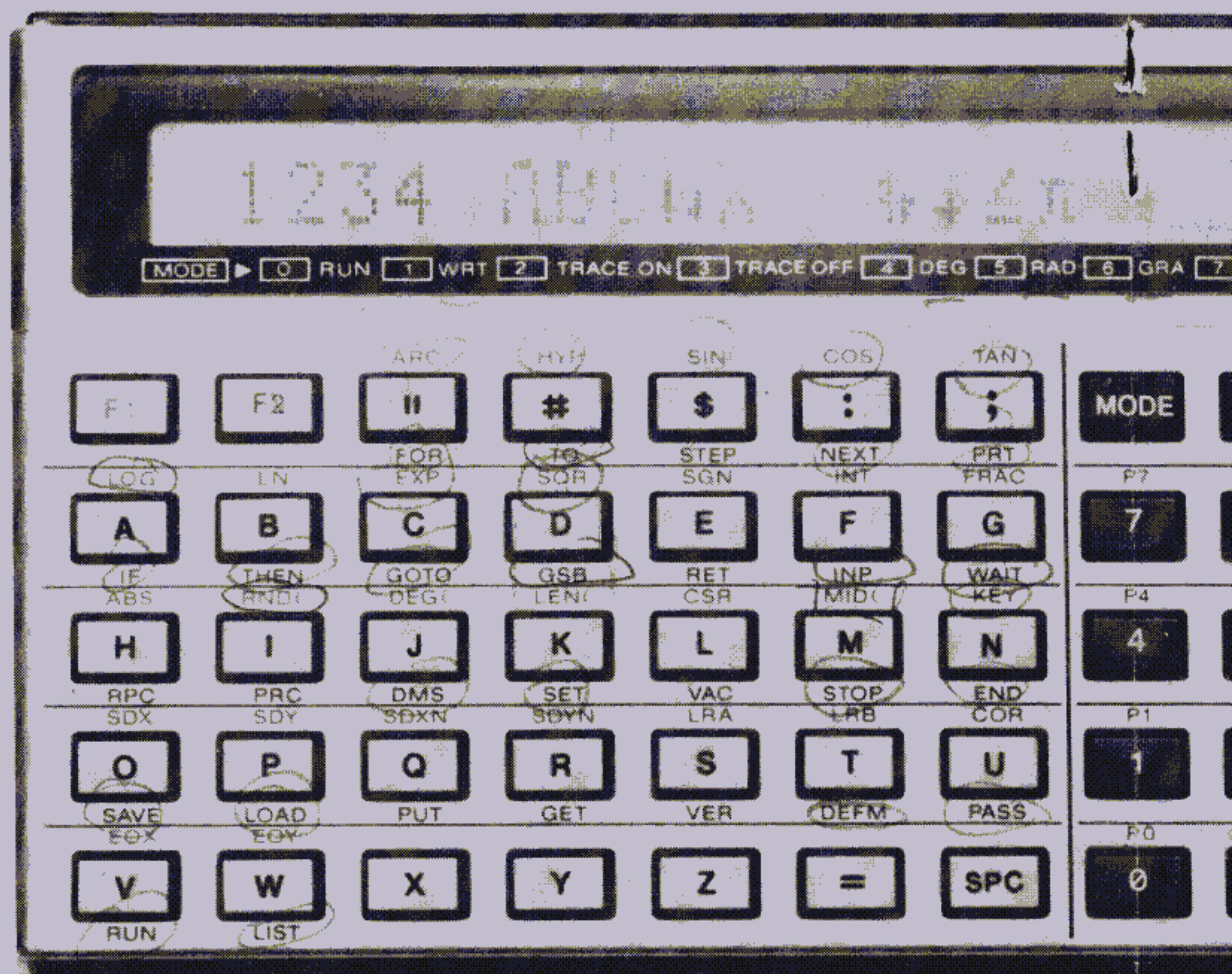
□ M. F.

Tableau 3

Pas	Code	Touches	Commentaires
00	32 0	STO 0	a étant affiché est placé dans R_0
01	81	R/S	arrêt pour introduction de b
02	32 1	STO 1	b est placé dans R_1
03	86 0	Lbl 0	début de traitement
04	33 0	RCL 0	
05	45	÷	} on divise a par b
06	33 1	RCL 1	
07	85	=	
08	49	Int	on prend la partie entière du quotient (q)
09	36	Pause	arrêt pour lecture
10	55	×	
11	33 1	RCL 1	on multiplie cette partie entière par b ($b \times q$)
12	85	=	
13	- 34 0	INV SUM 0	on retranche ce résultat de a (r)
14	33 7	RCL 7	} le reste, multiplié par 1000, donne une nouvelle valeur de a
15	39 0	Prd 0	
16	51 0	GTO 0	on recommence le traitement

Le Casio FX-70

Après le PC 1211 et le HHC, un troisième ordinateur de poche parlant BASIC vient de faire son apparition. Jusqu'à présent, vous teniez aux fonctions des calculatrices programmables dont les poquettes BASIC ne disposaient pas d'origine. Certes, on pouvait programmer une fonction similaire, mais il fallait le faire. Grâce aux fonctions de type calculatrice du BASIC de cette machine, vous allez devoir reconsidérer votre opinion.



Le micropo

■ Le premier coup d'œil sur le micropoche laisse une impression de sérieux, et même de sévérité. Un clavier très dense, avec des touches en plastique toutes de la même dimension et parfaitement alignées. Pas le moindre décrochement, le plus petit espace pour se reposer les doigts. Les sur et sous-titrages des touches, de couleurs respectivement rouge et bleue, occupent pratiquement tous les espaces de la platine avant.

C'est du concentré d'ordinateur, de la haute densité de fonctions au dm². Nous avons compté 13 colonnes de 5 touches soit 65 touches sur 165 mm × 55. Et deux à trois fonc-

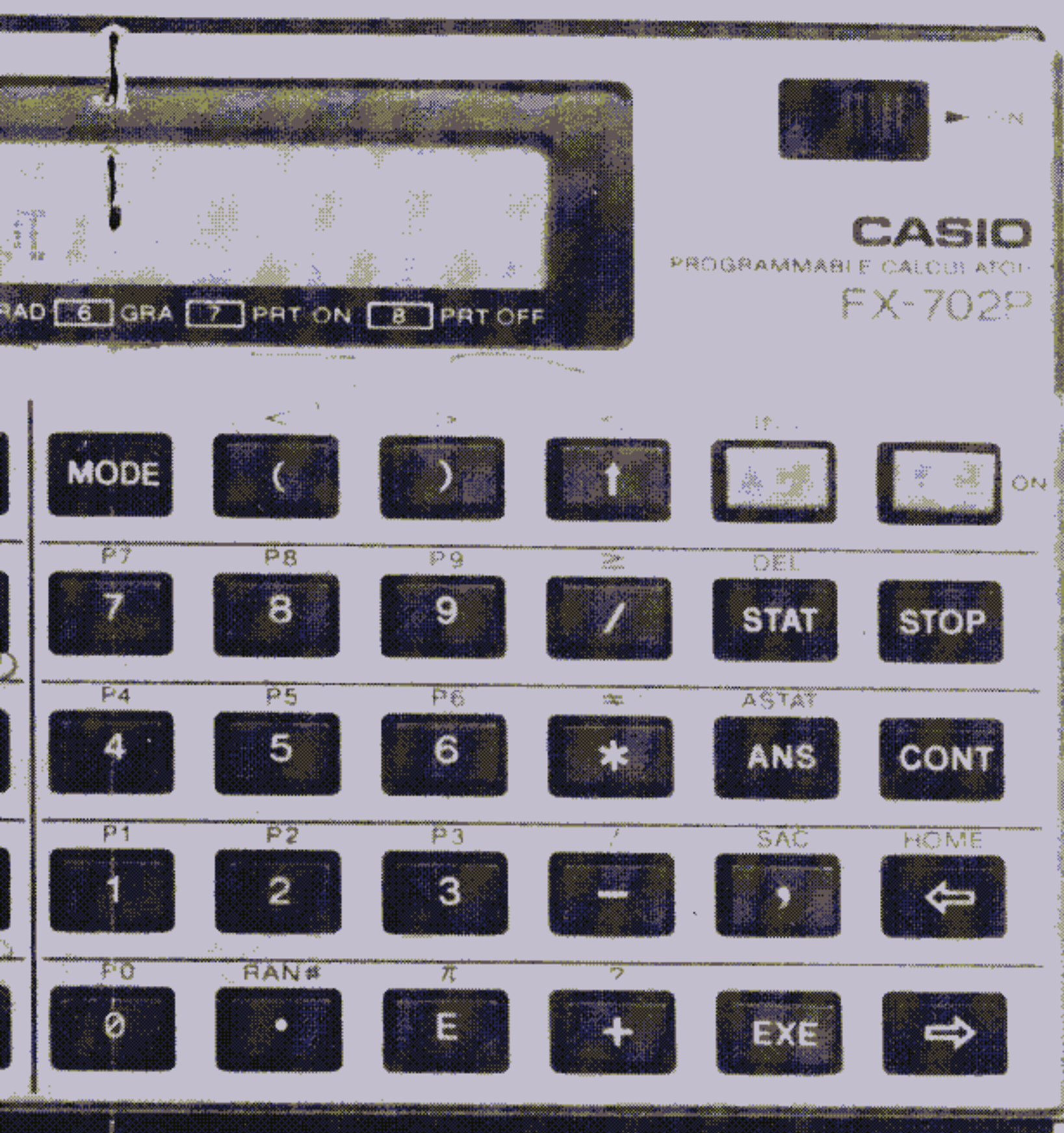
tions par touche ; bref, tout ce qu'il faut pour vous faire loucher abondamment. Au début, on se dit qu'il y en a trop, que l'on ne s'y retrouvera jamais... Le clavier est divisé en deux par un trait noir sur la platine. A gauche, le clavier alphabétique où les lettres sont disposées (paradoxe ?) dans l'ordre alphabétique. Cette disposition qui semble aller de soi n'est en fait pas courante du tout. Les habitués de claviers d'ordinateurs pesteront peut-être contre cette disposition qui ne leur est pas familière : au début, ils ne la trouveront pas commode. En contrepartie, les débutants et ceux qui n'hésitent pas à changer leurs habitudes pourraient bien juger que

cette disposition est beaucoup plus logique et plus agréable. L'avenir le dira...

La partie droite du clavier est elle aussi un peu déroutante à première vue : rien ne vient distinguer franchement les touches numériques et les touches d'opérateurs. La virgule se trouve exilée à côté du signe -, mais les deux points et le point virgule sont en haut de la partie alphabétique du clavier. Un bon point à relever toutefois : on y accède directement. Un clavier un peu déroutant donc au premier abord.

La touche EXE, qui joue le rôle d'ENTER, a le même format que les

-702 P



Le micropoche de Casio grandeur nature

autres et elle occupe sa petite place entre le + et la flèche à droite. Cette touche EXE est un héritage de la 502 P. Les adeptes des calculatrices auraient sans doute préféré une touche R/S (Run/Stop). Mais le STOP a sa touche à lui, un peu plus haut, juste au-dessus de la touche CONT (pour Continue), et c'est bien vu, même s'il n'est pas très facile, au début, de s'y retrouver lorsque l'on a déjà l'habitude d'un autre micropoche. La touche STOP, par exemple, ne joue pas le rôle de BREAK, et il faut avoir recours à une autre procédure pour arrêter un programme qui attend l'introduction d'une donnée grâce à l'ordre INPUT, mais on y reviendra.

En haut à droite, deux touches oranges, C et AC, que l'on trouve sur certaines calculatrices, servent l'une à effacer le dernier caractère et l'autre le registre d'affichage en entier. Vraiment très pratique. Complétons cela par la touche MODE qui, comme sur la 502 p, sélectionne les différents types de travail : de RUN (MODE 0) à PRT OFF, c'est-à-dire Printer off (MODE 8) en passant par WRT, TRACE ON, TRACE OFF, DEG, RAD, GRAD et PRT ON. La signification de chacun de ces modes est judicieusement rappelé juste sous l'affichage.

Sur la gauche du clavier, deux touches blanches portant l'inscription F1 en rouge et F2 en bleu permettent de sélectionner les fonctions sur et sous-titrées des autres touches. Cela facilite grandement l'introduction des instructions BASIC, des fonctions mathématiques et autres... De ce point de vue, on se croirait sur une calculatrice : F1\$ correspond à Sinus et F2\$ correspond à STEP. Cela permet de gagner du temps quand on écrit des programmes, et comment ! Il suffit d'appuyer sur deux touches pour obtenir n'importe quelle instruction BASIC.

Enfin, les touches numériques sont surtitrées avec les indications P0, P1, ..., P9. Ce sont des touches « utilisateur » qui définissent 10 zones de programmes indépendantes les unes des autres mais qui peuvent néanmoins s'appeler entre elles si nécessaire.

De très nombreuses fonctions sont donc disponibles, mais pour les utiliser toutes, il faudra un peu de pratique. Ne nous plaignons pas : abondance de biens ne nuit pas.

Continuons notre exploration de la machine et arrêtons-nous maintenant sur l'afficheur. Il est capable de reproduire jusqu'à 20 chiffres, lettres ou signes, chacun dans une matrice de 5 x 7 points rectangulaires assez joliment dimensionnée. Rien à redire à la lisibilité. En haut à droite de l'afficheur, une deuxième zone fait apparaître, en mode programme, jusqu'à 4 chiffres dessinés grâce à 7 segments. Ils indiquent le nombre de pas de programme qui restent disponibles.

Une petite innovation qui pourrait bien faire école : la densité des points et des segments de l'affichage est réglable au moyen d'un potentiomètre discrètement situé au-dessus du micropoche.

On peut ainsi obtenir le meilleur contraste possible quel que soit l'angle sous lequel on lit les indications présentes à l'affichage. C'est bien pratique pour travailler sur une table notamment.

L'interrupteur de marche-arrêt est du genre à glissière. On peut juger plus pratiques les interrupteurs à bascule (HP 41 C) ou à touche (PC 1211), mais cela se discute.

Au dos de l'appareil, un large couvercle amovible découvre un compartiment qui, outre les deux piles elles-mêmes protégées par un second couvercle, renferme un connecteur à 11 contacts et un emplacement vide où viendront probablement se loger dans le futur des mémoires mortes supplémentaires. On ne peut que l'espérer, à moins que Casio ne nous réserve d'autres surprises encore.

Sur le côté gauche du micropoche, un autre connecteur, protégé par un cabochon de plastique, permet le raccordement à l'interface-cassettes ; il compte 7 broches.

—En calculatrice :—
—la simplicité—

Le mode 0 autorise l'exécution des programmes et le calcul direct au clavier. On retrouve, ici encore, une grande ressemblance avec la manière de procéder sur une calculatrice. C'est ainsi que l'on se retrouve dispensé de taper le PRINT habituel du BASIC avant d'entrer opérandes et opérateurs. La machine fonctionne en notation algébrique parenthésée. Aucune difficulté pour les calculs en chaîne qui peuvent entrer sur une ligne com-charge défile. On écrit donc les calculs désirés de la même façon que sur une feuille de papier, on peut même y introduire des variables, et il suffit d'appuyer sur la touche EXE pour obtenir le résultat (si la feuille de papier faisait cela, on crierait au miracle, mais certains micropoches nous y ont déjà habitués : on se blase).

Il n'est malheureusement pas possible de mettre en réserve ces expressions ni de les rappeler une fois que la touche EXE a été pressée. Pour compenser en partie cette lacune, la touche ANS permet de rappeler le résultat du dernier calcul effectué ; on peut ainsi mémoriser ce résultat en l'affectant à une variable : A = ANS EXE stocke en A le

coup d'œil sur...

Le Casio

FX-702 P

dernier résultat obtenu par un calcul au clavier.

Les fonctions mathématiques et statistiques dont la machine est pourvue sont un de ses atouts majeurs. Elles y sont nettement plus nombreuses que sur la plupart des BASICs. Là encore, nous sommes dans le domaine de prédilection des calculatrices haut de gamme : moyenne, écart-type, fonctions hyperboliques, régressions linéaires, fonctions logarithmiques, exponentielles, etc. La factorielle par exemple se calcule directement en inscrivant le ! après le nombre. Un rêve pour les statisticiens. A remarquer encore que le mode Degrés - Minutes - Secondes exploite avec bonheur les possibilités de l'affichage : les signes °, ', '' sont écrits en clair (je veux dire en noir...). Quand on fait un calcul à partir de nombres écrits de la sorte, le passage en degrés décimaux se fait automatiquement, ce qui épargne bien des erreurs.

Ainsi, quand un calcul aboutit à un résultat décimal de 12,5 heures, l'affichage en degrés sexagésimaux s'obtient tout bonnement en appuyant sur F2 DMS ANS EXE, on lit alors 12° 30' 0.00, et si l'on vient ensuite diviser ces 10° 30' par 2, il suffit de demander / 2 EXE : l'affichage indique 12.5 / 2.

Toujours pour les calculs au clavier, on découvre une autre disposition très intéressante : SET. Cette instruction permet de limiter le nombre de décimales ou de passer en notation scientifique : SET E 3, par exemple, conduit à une notation scientifique avec 3 chiffres significatifs. SET 2 fixe la notation à deux décimales de la même façon que le FIX 2 des calculatrices, et les résultats sont arrondis au nombre de chiffres souhaités. Ce genre d'instruction ne se retrouve dans aucun des BASICs que je connais, et pourtant cela s'utilise bien plus simplement qu'un PRINT USING...

Qui plus est, le 702 P dispose aussi, pour le formatage des résultats dans un programme, des possibilités d'un USING classique sous la forme PRT # # . # # ↑ (ce n'est

qu'un exemple). Bref, une débauche de possibilités !

Encore un mot cependant sur l'instruction SET qui est décidément très astucieuse : lorsque le résultat d'une opération est affiché dans le format spécifié par SET et qu'il est réutilisé comme opérande pour un autre calcul, il réapparaît avec ses dix chiffres significatifs à l'affichage (douze en réalité pour les calculs). Je regrette seulement que l'extinction du micropoche fasse repasser en virgule flottante...

Inutile, je pense, de continuer : ce micropoche n'a pas grand'chose à envier aux machines existantes en ce qui concerne les calculs au clavier.

Et pour programmer ?

De prime abord, vous disposez de 1 680 pas de programme et de 26 variables. En fait, on verra que la mémoire est négociable entre données et programme. Après avoir compulsé la notice pour connaître les rudiments du BASIC, vous pourrez vous lancer dans l'écriture de votre premier programme. Vous prendrez vite le pli, car ce n'est pas très compliqué : vous pressez sur MODE 1 et l'afficheur indique alors une série de chiffres compris entre 0 et 9, ce sont les zones de programmation non encore utilisées. A gauche de ces chiffres, l'indication READY P0 ou READY P1, P2, etc., indique dans quelle zone se trouve le pointeur de programme. Si vous

voulez intervenir dans une autre zone, il vous suffit de la spécifier en pressant sur F1 suivi de son numéro. Comme dit plus haut, le nombre d'octets disponibles en mémoire est affiché à l'extrême droite.

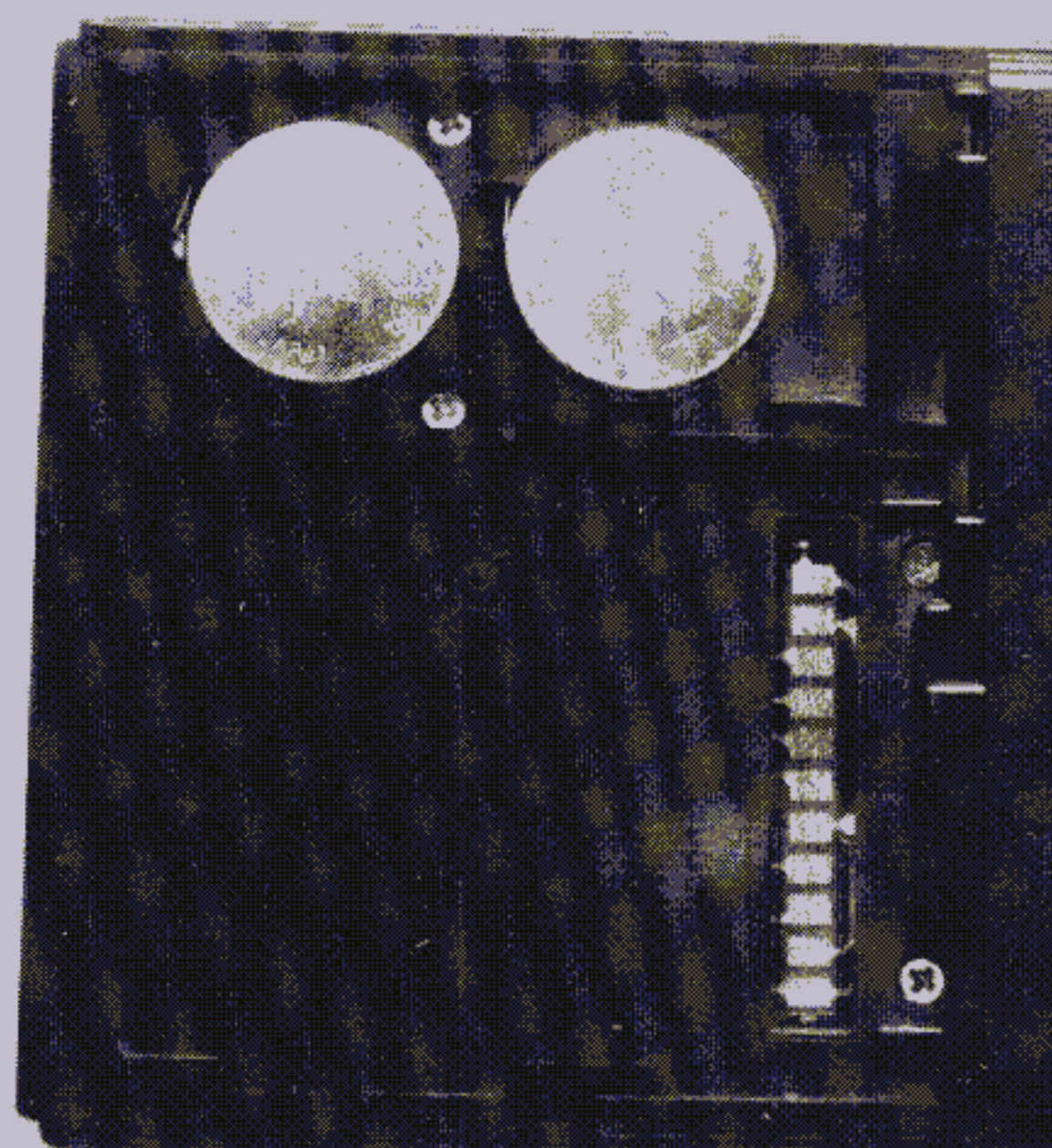
L'existence des zones de programme est une fois de plus un héritage des calculatrices programmables, et spécialement de la 502 P. Chaque zone est indépendante et se comporte comme une mémoire programme à part entière. Cela dit, pour les besoins d'un algorithme, on peut très bien utiliser des appels d'une zone à l'autre avec GOTO et GOSUB. Tout cela est vraiment très souple, et pour un peu, on croirait travailler avec une disquette, capacité de stockage mise à part. Cette organisation est particulièrement intéressante, et nous y reviendrons quand il sera question de l'interface-cassettes.

Après avoir choisi une zone de travail, il ne reste qu'à introduire le programme : n° de ligne de 0 à 9999 (par zone... ça laisse de la marge !), puis instructions. Pour entrer ces instructions, vous avez le choix entre deux méthodes : soit appuyer une touche préfixe F1 ou F2 suivie de la touche adéquate (méthode calculatrice : les ordres sont entrés en abrégé), soit l'intituler en toutes lettres comme sur un ordinateur classique.

Dès que l'on a le clavier bien en tête, ce choix est précieux. Comme de surcroît, les 12 ordres les plus courants du BASIC sont ingénieusement disposés dans les deux premières rangées de touches, avec un peu d'habitude, il est facile et rapide de programmer.

La palette d'instructions est vaste : elle correspond à un BASIC étendu présentant toutefois certaines particularités : pas de ELSE après IF ... THEN, et une orthographe simplifiée pour plusieurs instructions classiques : PRT pour PRINT (affichage), INP pour INPUT (entrée des données), GSB et RET à la place de GOSUB et RETURN (aller-retour à un sous-programme), etc.

Pour ne pas provoquer d'erreur de syntaxe, il est préférable et plus pratique de recourir dès le début aux touches de préfixe. Toutes les fonctions mathématiques ou statistiques sont utilisables dans les programmes, ce qui permet souvent de rendre ces derniers particulièrement compacts.



Les 2 batteries au lithium et le connecteur à 11 contacts

Autre particularité intéressante : la boucle FOR ... NEXT n'est pratiquement pas limitée : elle peut s'effectuer avec un pas positif ou négatif et *non nécessairement entier*. On peut imbriquer jusqu'à huit niveaux de boucles et utiliser jusqu'à dix niveaux de sous-programmes.

— Des tableaux — — à deux dimensions —

Si l'on regarde maintenant du côté des variables alphanumériques, il ne faut pas trop se plaindre non plus, mais ce n'est pas un ordinateur de table, il s'en faut. Chaque variable accepte jusqu'à 7 caractères, ce qui est un peu juste, mais il existe heureusement une zone mémoire spéciale, nommée \$, qui permet de travailler sur 30 caractères. Cette zone spéciale sera certainement très utilisée ; avec un peu de gymnastique et en usant abondamment des concaténations, il sera possible de s'en contenter dans la plupart des cas.

Les tests de comparaison opèrent sur les chaînes de caractères et deux instructions sont disponibles : LEN (qui donne la longueur de la chaîne) et MID (qui remplace le MID\$ habituel du BASIC). Une petite tracasserie à prévoir cependant : les variables numériques et alphanumériques se répondent : A et A\$ occupent la même zone mémoire. Chacun devra donc, pour éviter les surprises, noter son tableau de variables quand il écrira un programme.

Au démarrage, 26 variables sont disponibles, mais ce nombre peut augmenter si l'on accepte d'empiéter sur la mémoire du programme. La partition n'est pas automatique : il faut la définir soi-même à l'avance (instruction DEFM). On peut ainsi obtenir jusqu'à 226 variables, mais il ne reste plus alors que 80 pas de programme.

Les 26 premières mémoires sont notées A à Z (ou A\$ à Z\$). La tranche suivante comprend 10 autres mémoires (de A0 à A9), puis vient une troisième tranche (B0 à B9), etc., jusqu'à T9. De plus, on peut indiquer les variables en utilisant la notation A(1), A(2) ... A(10)... qui correspondent respectivement à A, B, ... B0... Mais cela ne s'arrête pas là, car il est possible d'utiliser des tableaux à deux dimensions ; les variables se notent alors A(0,0) ... (A19,9). Il faut une certaine atten-

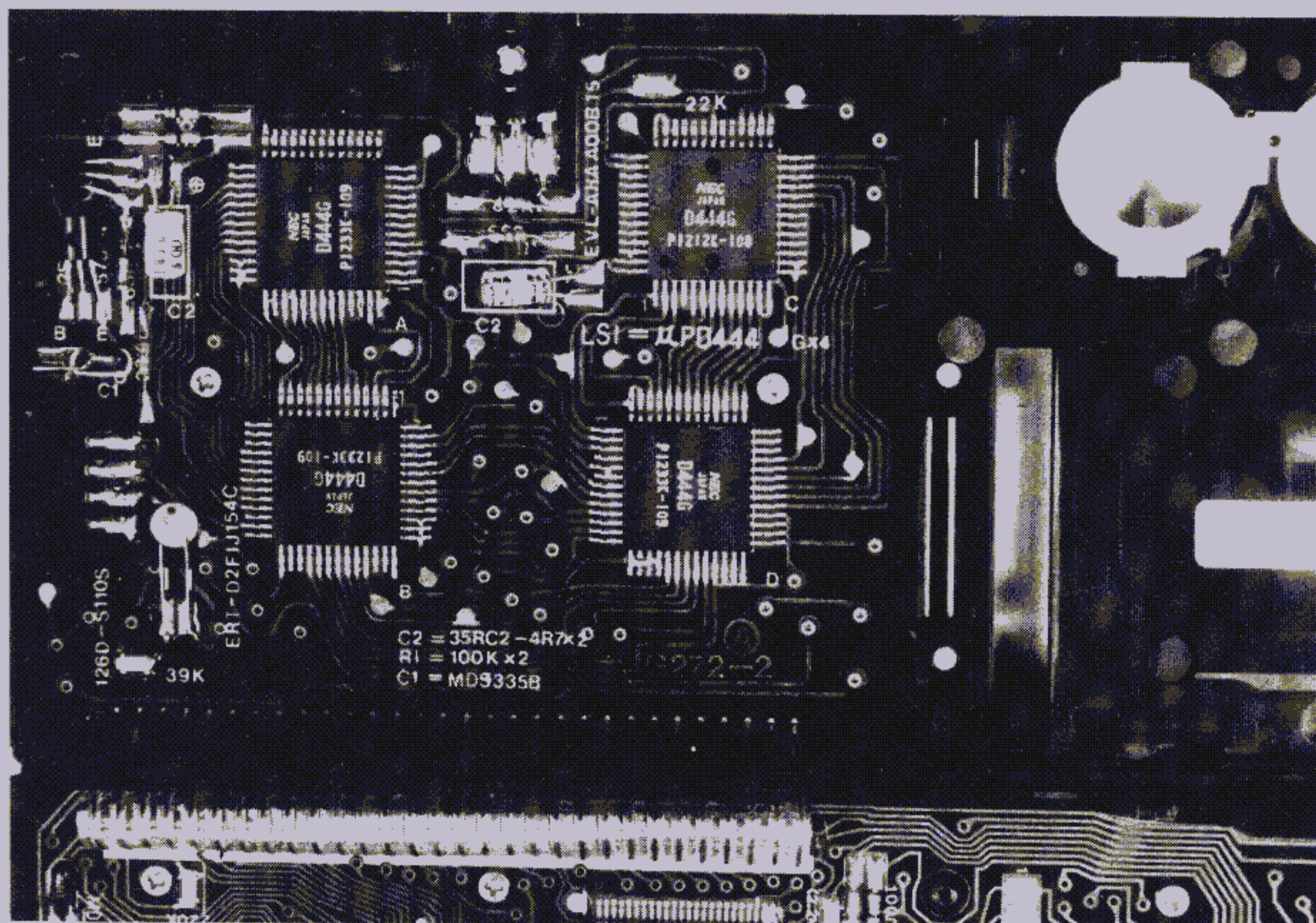
tion pour s'y retrouver, mais ça fonctionne, et c'est une très bonne chose pour un si petit système.

Un petit mot sur la vitesse d'exécution des programmes vous intéressera certainement. Le test le plus simple consiste en une boucle que l'on fait effectuer un certain nombre de fois (FOR N = 1 TO 100 : NEXT). Les 100 tours de circuits sont parcourus en 2 secondes et 3 dixièmes, contre 24,6 secondes sur le PC 1211, et une demi-seconde sur le TRS 80 modèle 1.

Quand le programme ne tourne pas du premier coup, c'est bien connu, il faut le corriger. L'ordre LIST permet de visualiser le contenu d'une zone programme. En mode 0, c'est-à-dire RUN, le défilement est automatique et chaque ligne de pro-

TRACE ON et OFF, autorisent l'exécution d'un programme pas-à-pas avec affichage des numéros de lignes.

Et maintenant, un petit gadget : PASS. Il permet de privativer un programme en enregistrant un mot de passe de 8 caractères au plus. Ce mot devra être introduit après chaque ordre LIST ou CLR pour qu'ils soient effectivement exécutés. Pour effacer ce mot de passe, il faut soit vider tout le contenu de la mémoire programme, soit réintroduire le même mot après PASS. Malgré nos efforts, nous sommes toujours à la recherche de la touche BREAK. AC et STOP arrêtent bien le déroulement d'un programme, mais le premier ne permet pas le redémarrage, à la différence du second. Cela dit, si le programme est arrêté sur



gramme est déroulée in extenso : elle défile vers la gauche si elle dépasse la capacité de l'afficheur. En mode 1 (WRT), LIST joue le rôle d'éditeur et l'accès à une instruction se fait à l'aide des deux flèches avant et arrière. Les touches INS et C permettent l'insertion, le remplacement et la suppression. Il ne manque que la possibilité de remonter ou de descendre le long de la liste.

Il existe en plus du LIST classique un LIST V qui permet de voir en déroulement le contenu de toutes les variables après affichage de la partition mémoire. LIST ALL donne à la fois les programmes et les données et, avant chaque programme, le nombre de pas occupés par celui-ci.

Les modes 2 et 3, respectivement

En haut, le potentiomètre de l'afficheur. Au centre, 4 circuits spécialisés ayant chacun 42 broches

l'attente d'une entrée de donnée avec INP et que vous vouliez interrompre l'exécution, il faut chercher longtemps avant de pouvoir s'en sortir. En fait, la solution n'est pas compliquée : il suffit de refaire MODE 0. Mais il fallait le trouver : ce n'est pas signalé dans la notice (ou alors cela nous a échappé...).

Pour finir, deux fonctions intéressantes : WAIT et KEY WAIT ajustent la durée d'affichage entre 0 et 40 secondes et KEY permet de « saisir au vol » un caractère frappé au clavier (elle correspond à INKEYS\$

coup d'œil sur...

Le Casio

FX-702 P

ou GET) : l'une et l'autre seront bien appréciées par ceux qui veulent préparer des programmes interactifs. Comme nous sommes très curieux, nous avons même ouvert le micro-poche pour voir ce qu'il avait dans le ventre. Trois vis à enlever et un doigté d'ouvreur de coffre-fort pour faire sauter délicatement les petits crochets en plastique (fragile) et l'on découvre deux circuits imprimés repliés l'un sur l'autre et reliés par un câble en nappe. A l'intérieur de l'espèce de portefeuille ainsi constitué, 8 circuits à très haute intégration couverts de pattes sur tous les côtés, quelques composants discrets : condensateurs, résistances, diodes et transistors. De la belle ouvrage, et très dense. Pas de buzzer en vue, et donc une machine muette. C'est dommage, mais il ne reste pas beaucoup de place et le petit espace disponible sera beaucoup mieux employé pour recevoir de futurs modules de mémoire morte ou — mais ce n'est pas sûr — vive.

Un petit tour des périphériques

A l'heure actuelle, rien n'est encore disponible en France dans le domaine des accessoires. Nous avons pu néanmoins essayer l'interface cassette qui sera bientôt importée. Elle se présente sous la forme d'un berceau dans lequel vient s'enficher le micro-poche. Un gros cordon en sort, qui contient 3 fils et se termine sur des prises « jack » de 3,5 pour le signal et 2,5 pour la télécommande.

Tout fonctionne bien : j'ai fait des essais avec un magnéto-cassette ordinaire et un microcassette. Dans les deux cas, pas de problème. SAVE # 0 à 9, puis une identification (file name) comportant jusqu'à 8 caractères, permet d'enregistrer une zone de programme. L'identification est facultative. Si le numéro est oublié, le programme sauvé sera celui où se trouve le pointeur. SAVE ALL enregistre tout le contenu de la mémoire, programmes et données.

LOAD fonctionne sur le même mode. Si l'enregistrement contient une identification, elle est annoncée à l'affichage, puis remplacée par READY Px si tout s'est bien passé.

Il est possible de programmer ce LOAD. Et du fait de la séparation en zones de la mémoire programme, on arrive à faire des manipulations vraiment intéressantes avec le chaînage des programmes. Le changement n'efface pas le contenu d'une zone, et avec des numéros de lignes qui ne s'écrasent pas on peut faire de très jolies choses. Cela revient en fait à travailler comme avec une disquette et permet donc d'exécuter séquentiellement des programmes qui dépassent de beaucoup la taille de la mémoire (on parle de recouvrement). Couplée avec les appels d'une zone à l'autre par GOTO et par GOSUB, cette disposition augmente considérablement les possibilités initiales du FX 702 P (1 680 pas de programmes) pour en faire le (quasi) équivalent d'ordinateurs bien plus volumineux.

Les données sont sauvées et chargées par PUT et GET, suivis d'une identification et de l'intervalle de variable concerné. Ces deux instructions sont elles aussi programmables. De quoi bien s'amuser ! La vérification d'un enregistrement s'effectue par VER.

Ajoutons pour être complet que l'interface cassette fonctionne également avec les deux calculatrices 502 P et 602 P.

Une imprimante thermique est disponible aujourd'hui au Japon au prix de 16 500 yens, ce qui laisse espérer un prix en France compris entre 500 et 600 FF... Elle permet

Le Casio FX-702 P

Caractéristiques

- Dimensions : 17 × 165 × 82 mm
- Poids : 176 g
- Alimentation : 2 piles lithium CR 2032
- Affichage : 1 ligne de 20 caractères en matrice 5 × 7 et quatre chiffres à 7 segments.
- Langage de programmation : BASIC
- Plus une bonne quarantaine de fonctions mathématiques et statistiques préprogrammées.
- En option : interface-cassette FA-2 et imprimante à papier métallisé prochainement disponibles en France.

l'impression du contenu de la mémoire, des ordres PRINT et INPUT et donne 20 caractères par ligne sur un ruban de papier métallisé de 35 mm de large, et son autonomie atteint jusqu'à 13 000 lignes sur batterie rechargeable. Vitesse d'impression : 40 caractères par seconde.

Sont également prévus des modules de MEM qui contiendront des programmes tout faits. Ils seront logés à côté du compartiment de piles.

Notices : Casio doit mieux faire

Toutes les notices de machines Casio que nous avons eues dans les mains étaient notoirement insuffisantes pour permettre à l'utilisateur de bien connaître le matériel. C'était le cas pour la 502 P. Pour la 702 P, ce n'est guère mieux. Si la documentation est plus épaisse, le micro-poche est beaucoup plus complexe, on n'est guère plus avancé et c'est dommage car cela complique énormément l'apprentissage. Il sera important d'acquérir des notions de programmation dans d'autres ouvrages.

Un livre d'exemples de programmes complète (ou remplace ?) la documentation. Au vu des listes de programmes en anglais, elles, bien sûr, et des schémas, nous pouvons affirmer que ce sont exactement les mêmes programmes que ceux qui étaient présentés avec la FX 502 P.

Le Casio FX 702 P est un ordinateur très séduisant. Ses principaux attraits sont un BASIC complet, des possibilités impressionnantes pour une machine de cette dimension, et sa grande souplesse d'utilisation. C'est le genre de machine dont il sera difficile de venir à bout. Ajoutons que son prix (1 250 FF ttc) en fait une concurrente sérieuse des micropoches haut de gamme.

Revers de la médaille : l'utilisation de cette machine est assez complexe, ce qui découle de l'importance même de ses possibilités. Espérons que cela ne découragera pas le programmeur débutant. C'est un risque, car la notice ne l'aidera pas beaucoup à s'initier.

□ Xavier de La Tullaye,
Jean-Pierre Brunerie

Obtenez de votre TI 57 qu'elle vous livre ses chiffres de garde

Quand elle effectue vos calculs, la TI 57 n'affiche pas ses résultats avec tous les chiffres qu'elle a obtenus : elle en conserve quelques-uns par devers elle. Ce sont les chiffres de garde.

■ Sur votre TI 57, effectuez par curiosité les opérations suivantes : $12 \div (2 \times 2) =$, et admirez le résultat qui est irréprochable = 3. Rien à redire, n'est-ce pas ? Quand on élève 2 au carré, on obtient bien 4, et quand on divise 12 par 4, le quotient est bien 3. Très bien. Jusque-là rien d'anormal, mais attendons la suite...

Demandons maintenant quelle est la partie entière du résultat que nous venons d'obtenir en appuyant sur 2nd Int. Aïe, rien ne va plus : la partie entière de 3 est 2 !

La première fois que l'on obtient ce genre d'aberration, on a toujours un petit doute, et l'on recommence aussitôt pour vérifier que l'on n'a pas soi-même commis d'erreur : $12 \div (2 \times 2) =$ 2nd Int. Comme la TI 57 s'obstine, on passe en revue les différentes solutions :

- essayer de se faire rembourser la machine par la personne qui l'a vendue,
- appliquer la méthode peu efficace (mais ô combien soulageante) du "coup de pied dans l'engin",
- lire les quelques lignes qui suivent.

En réalité, la TI 57 calcule tous les résultats que l'on lui demande avec 11 (et parfois 12) chiffres significatifs. Or sur ces chiffres, 8 seulement

sont affichés, le huitième étant correctement arrondi. C'est d'ailleurs grâce à cet arrondi de l'affichage que nous avons obtenu que $12 \div (2 \times 2)$ soit égal à 3. Mais c'est aussi à cause de lui que la partie entière de ce "3" est égal à 2.

Avec le petit programme que je vous soumetts, chacun pourra afficher les 11 chiffres que la calculatrice connaît. C'est beaucoup mieux que les huit qu'elle vous montre, non ?

Le mode d'emploi en est très simple : après avoir introduit (sans faute...) ledit programme, et être repassé en mode calcul, tapez : $2 \times 2 =$ puis RST et R/S. On obtient d'abord à l'affichage un premier nombre : 4,0000000, puis — après 2 secondes environ — un autre nombre : 0.5100000. Il ne faut pas tenir compte du point décimal dans le second nombre. En mettant les deux bout à bout, on obtient le résultat tel que la machine l'a stocké, c'est-à-dire avant que n'intervienne l'arrondi de l'affichage : $2 \times 2 = 4,0000000051$.

————— Voilà —————
————— l'explication —————

Si vous refaites maintenant $12 \div (2 \times 2) =$ puis RST et R/S, vous obtiendrez d'abord 2,9999999 puis 9,6100000, ce qui vous permet de savoir que le résultat qui a été réellement calculé est 2,9999999961. C'est donc uniquement grâce à l'arrondi de l'affichage que le 3, attendu, est apparu. Et cela explique aussi que la partie entière de ce "3" soit 2.

Grâce à ce programme, vous pourrez vérifier que, pour la calculatrice, le nombre pi est égal à 3,1415926536 (faire π RST R/S), que $\cos 100^\circ$ est égal à -0,17364817776 (faire 100 cos RST

PRECISEMENT			
AUTEUR : DAMIEN BOMMART			
COPYRIGHT L'ORDINATEUR			
DE POCHE ET L'AUTEUR			

0	32	0	STO 0
1	40		/X/
2	18		2ND LOG
3	49		2ND INT
4	65		-
5	07		7
6	85		=
7	40		/X/
8	32	1	STO 1
9	01		1
10	00		0
11	32	5	STO 5
12	35		Y PUISSANCE X
13	33	1	RCL 1
14	85		=
15	49		2ND INT
16	32	2	STO 2
17	55		MULTIPLIE PAR
18	33	0	RCL 0
19	85		=
20	32	3	STO 3
21	-49		INV 2ND INT
22	40		/X/
23	55		MULTIPLIE PAR
24	33	5	RCL 5
25	85		=
26	32	4	STO 4
27	33	3	RCL 3
28	49		2ND INT
29	45		DIVISE PAR
30	33	2	RCL 2
31	85		=
32	48	7	FIX 7
33	36		2ND PAUSE
34	36		2ND PAUSE
35	33	4	RCL 4
36	81		R/S

R/S) ou que le cube de 25 vaut 15625,000006 (faire $25 \times 3 =$ RST R/S). Etc.

Vous disposez maintenant d'un moyen d'en savoir plus sur la précision de votre micropoche.

□ Damien Bommart

Quand la TI 57 épluche les racines...

Les équations du second degré n'ont pas toujours de racine. Voyons comment programmer ce petit problème sur un micropoche.

■ Dès la classe de seconde, l'équation du deuxième degré fait partie du quotidien du "potache". Le problème consiste à trouver quelle valeur il faut donner à x pour que $y=2x^2-3x+1$ (par exemple) soit égal à zéro.

Les esprits rapides ont déjà trouvé : "si $x=1$, alors ça marche". Bravo ! Mais tout de même, cela ne suffit pas. En effet, si $x = \frac{1}{2}$, ça "marche" aussi. Car l'équation du 2^e degré a la fâcheuse propriété d'admettre tantôt 2 solutions, tantôt une seule... et tantôt aucune.

Il est possible de représenter graphiquement une équation du 2^e degré (comme bien d'autres, d'ailleurs). On trace un axe horizontal :

25 lignes pour nourrir la mémoire de votre TI 57

Voici une solution pratique qui vous évitera les calculs à la main et déterminera rapidement les racines... si elles existent.

Mode d'emploi :

- Initialisation : RST.
- Calcul : introduire a, R/S, puis b, R/S (affichage : calcul intermédiaire), puis c, R/S.
- Si l'affichage clignote, pas de racine ($\Delta < 0$). Pour connaître la valeur de Δ , appuyer sur (CE ; x^2 ; + / -).
- 2 racines : après l'affichage de la 1^{re} racine, appuyer sur R/S pour obtenir la seconde.
- 1 seule racine : celle-ci est tout simplement répétée, comme s'il s'agissait de 2 racines distinctes.

Pour un nouveau calcul, recommencer à partir de l'initialisation.

```

                2E DEGRE
AUTEUR : OLIVIER DABEE
COPYRIGHT L'ORDINATEUR
DE POCHE ET L'AUTEUR
*****
0  32  0  STO  0
1  34  0  SUM  0
2  81                R/S
3  32  1  STO  1
4  23                X AU CARRE
5  65                -
6  33  0  RCL  0
7  55                MULTIPLIE PAR
8  02                2
9  55                MULTIPLIE PAR
10 81                R/S
11 85                =
12 24                RACINE CARREE DE X
13 32  2  STO  2
14 84                + / -
15 86  1  2ND LBL  1
16 65                -
17 33  1  RCL  1
18 85                =
19 45                DIVISE PAR
20 33  0  RCL  0
21 85                =
22 81                R/S
23 33  2  RCL  2
24 51  1  GTO  1
    
```

l'axe des abscisses (ou axe des x), et un axe vertical : l'axe des ordonnées (ou axe des y). Pour différentes valeurs de x , on détermine grâce à l'équation les valeurs de y correspondantes. Par exemple, pour l'équation : $y=2x^2-3x+1$, si l'on donne à x la valeur 2, on obtient $y=3$; de même, pour $x=0$, on obtient $y=1$.

A chaque valeur de x (et donc de y), on associe donc un point d'abscisse x , et d'ordonnée y (fig. 1). En calculant un nombre suffisant de valeurs, on définit assez de points pour tracer une courbe.

Dans le cas présent, la courbe représentative de $y=ax^2+bx+c$ s'appelle une *parabole*. On remarque (fig. 2) que l'on a 3 possibilités pour

placer la courbe par rapport à l'axe des abscisses :

- la courbe coupe l'axe des abscisses en 2 points distincts ; cela signifie qu'à ces 2 points correspondent 2 valeurs de x pour lesquelles $y=0$, c'est le cas où l'équation admet 2 solutions distinctes ;
- la courbe coupe l'axe des abscisses en 1 point (en fait, elle est tangente à cet axe) ; l'équation admet alors une seule solution (certains vous diront 2 solutions confondues, mais, de toute façon, le résultat est le même) ;
- la courbe ne coupe pas l'axe des abscisses : l'équation n'a pas de solution.

On comprend alors le caractère lunatique de l'équation du 2^e degré :

0, 1, ou 2 solutions, cela dépend de la position de la courbe par rapport à l'axe des x.

Si la construction de la courbe permet d'obtenir la représentation graphique d'une équation, elle s'avère peu précise quand on désire connaître très exactement les solutions (appelées aussi racines).

En revanche, la factorisation aboutit à une méthode de calcul adaptée à la résolution de toute équation du type $ax^2 + bx + c = 0$. On pose $ax^2 + bx + c = 0$ (avec $a \neq 0$).

En divisant par a :

$$x^2 + \frac{b}{a}x + \frac{c}{a} = 0, \text{ d'où :}$$

$$x^2 + \frac{b}{a}x + \left(\frac{b}{2a}\right)^2 - \left(\frac{b}{2a}\right)^2 + \frac{c}{a} = 0$$

$$\left(x + \frac{b}{2a}\right)^2 - \frac{b^2}{4a^2} + \frac{c}{a} = 0$$

$$\left(x + \frac{b}{2a}\right)^2 - \frac{b^2 - 4ac}{4a^2} = 0$$

$$x + \frac{b}{2a} = \frac{\sqrt{b^2 - 4ac}}{2a}$$

ou

$$x + \frac{b}{2a} = -\frac{\sqrt{b^2 - 4ac}}{2a}$$

$$\text{Soit : } x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

On pose : $\Delta = b^2 - 4ac$. Δ est appelé discriminant de l'équation.

• si $\Delta < 0$, alors $\sqrt{\Delta}$ n'existe pas, l'équation n'a pas de racine.

• si $\Delta = 0$, alors $\sqrt{\Delta} = 0$, et donc $x = -\frac{b}{2a}$, racine unique (on dit aussi racine double)

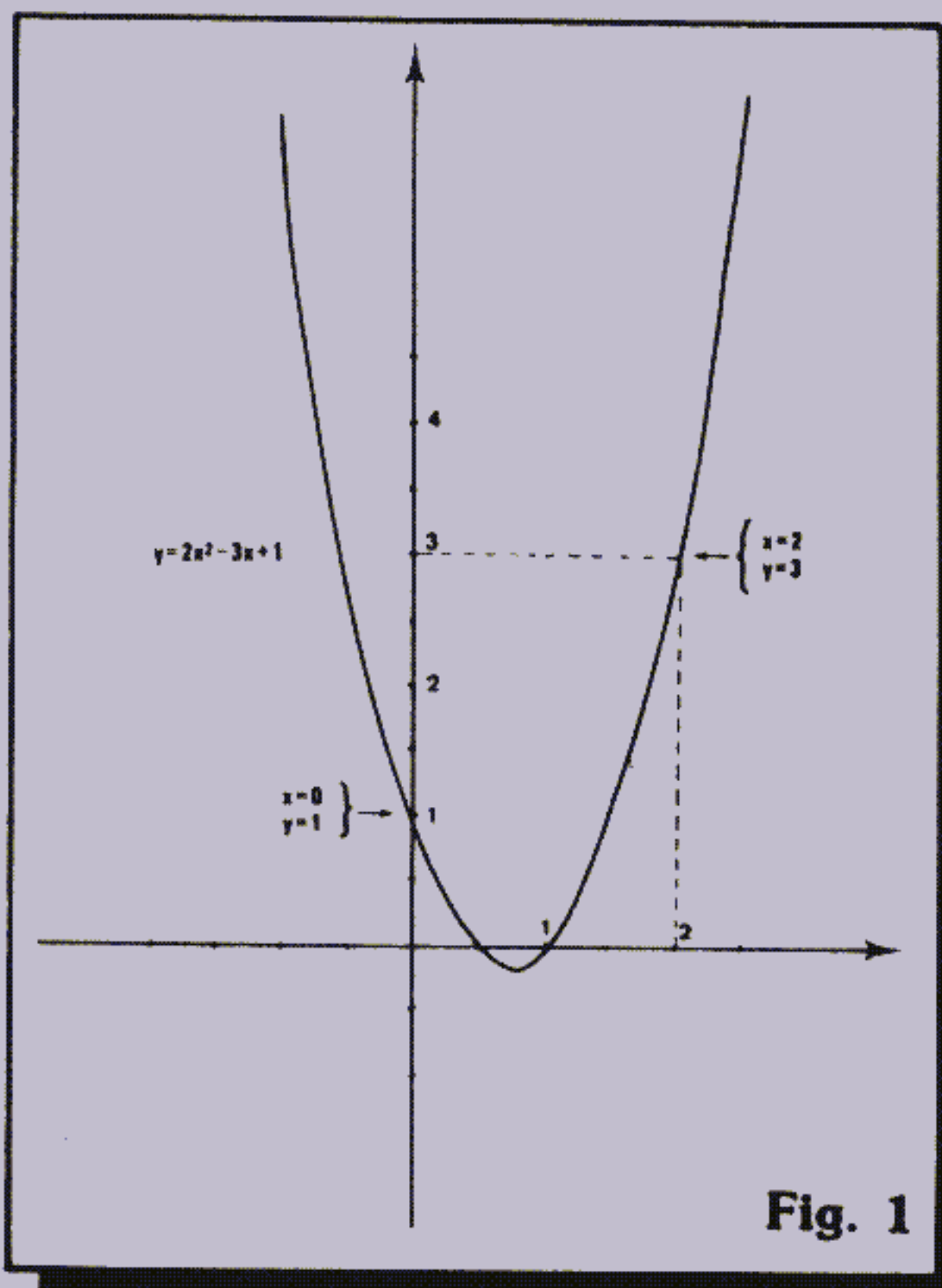


Fig. 1

• si $\Delta > 0$, on a 2 racines :

$$x_1 = \frac{-b - \sqrt{\Delta}}{2a}$$

$$x_2 = \frac{-b + \sqrt{\Delta}}{2a}$$

Fig. 2 : représentation graphique des différentes hypothèses d'école

	2 racines	1 racine	Pas de racine
a > 0			
a < 0			

Ces 3 cas traduisent les 3 positions possibles de la courbe par rapport à l'axe des x (fig. 2).

Parmi les nombreuses applications physiques et mathématiques des équations du second degré, j'en ai retenu une qui vous permettra de poser des "colles" en déterminant facilement vous-même la solution. Nous avons vu que si $b^2 - 4ac > 0$, on a 2 racines :

$$x_1 = \frac{-b - \sqrt{\Delta}}{2a}; x_2 = \frac{-b + \sqrt{\Delta}}{2a}$$

Effectuons la somme $S = x_1 + x_2$. On obtient :

$$S = \frac{-b - \sqrt{\Delta} - b + \sqrt{\Delta}}{2a} = -\frac{b}{a}$$

De même, effectuons le produit $P = x_1 \times x_2$.

On obtient :

$$P = \frac{(-b - \sqrt{\Delta})(-b + \sqrt{\Delta})}{4a^2}$$

$$= \frac{b^2 - b\sqrt{\Delta} + b\sqrt{\Delta} - \Delta}{4a^2}$$

$$= \frac{b^2 - (b^2 - 4ac)}{4a^2}$$

$$= \frac{4ac}{4a^2}$$

$$= \frac{c}{a}$$

En remarquant que si $a \neq 0$, $ax^2 + bx + c = (x^2 + \frac{b}{a}x + \frac{c}{a})a$,

d'après le calcul de S et de P, on peut remplacer $\frac{b}{a}$ par $-S$ et $\frac{c}{a}$ par P, ce qui nous donne : $x^2 - Sx + P = 0$. Ainsi, connaissant la somme et le produit de 2 nombres, on peut en déduire ces 2 nombres par simple résolution d'une équation du 2^e degré.

Exemple : Trouver les 2 nombres dont la somme est $\frac{3}{2}$, et le produit

$$\frac{1}{2}$$

On pose : $x^2 - Sx + P = 0$

$$\text{soit : } x^2 - \frac{3}{2}x + \frac{1}{2} = 0$$

Et l'on résoud

$$x_1 = \frac{\frac{3}{2} - \sqrt{\frac{9}{4} - \frac{4}{2}}}{2} = \frac{1}{2}$$

$$x_2 = \frac{\frac{3}{2} + \sqrt{\frac{9}{4} - \frac{4}{2}}}{2} = 1$$

Ces 2 nombres sont $\frac{1}{2}$ et 1. Et l'on vérifie bien que $\frac{1}{2} + 1 = \frac{3}{2}$ et que $\frac{1}{2} \times 1 = \frac{1}{2}$. Le tour est joué.

□ Olivier Dabée

Bouillon de culture pour FX 502 P

Certains utilisateurs de la Casio FX 502 P se sont sentis un peu délaissés dans nos colonnes. L'un d'entre eux, à l'attention de ses confrères, nous a adressé ce qui suit.

■ Passionnante étude que celle de la "génétique" mathématique. Avec le "jeu de la vie", l'Ordinateur de poche invitait à un cours de pseudo-microbiologie appliquée. Il ne s'en fallait que d'un virus pour passer à la pratique de ce jeu — vital — et pouvoir à notre tour, 502 - Péistes, transformer notre micropoche en usine bactériologique.

Le jeu consiste (voir le rappel en encadré) à semer, au gré de votre fantaisie et de votre humeur, une

colonie microbienne de départ, génération spontanée en quelque sorte, puis à examiner son évolution au cours des populations successives.

Les certificats de naissance sont, bien sûr, délivrés par votre micropoche qui se charge ainsi de gérer tout ce petit monde.

En parfait "scientifique", vous aurez donc tout le loisir d'observer le comportement de l'univers que vous avez créé.

Chaque colonie microbienne se compose de 10 membres répartis à l'affichage de la manière suivante : la présence d'un microbe dans une case se traduit par un 1 et chaque case vide se traduit par un zéro.

De ce fait, les premières cases à partir de la gauche, quand elles sont vides, ne sont pas représentées. Ainsi, une population comportant à partir de la gauche 2 cases vides, puis 3 microbes, puis 5 cases vides, donnera le nombre 11100000.

A vous de jouer maintenant :

Déroulement du programme

- En appuyant sur P0 (initialisation), le pointeur se place à la ligne 001 (HLT). On introduit à ce moment la génération de départ.
- Après avoir appuyé sur EXE, de l'instruction 002 à l'instruction 029 chaque contenu de case est mémorisé : M_1 à M_0 .
- Lignes 030 à 105, pour chaque case, les 4 voisines sont cumulées puis rangées dans les registres M_1 à M_7 .
- Lignes 106 à 201, chaque position est analysée, et la nouvelle génération est mémorisée.
- Lignes 202 à 229, les différents registres sont transmis en M_7 , puis rappelés à l'affichage pour indiquer la nouvelle génération qui, bien entendu, servira de point de départ pour la suivante, et ainsi de suite.

Les règles du Jeu de la Vie

1. Le voisinage d'une case est constitué par les quatre cases qui l'entourent, deux à droite et deux à gauche.
2. Les cases vides dont le voisinage comprend deux ou trois microbes enregistrent une naissance.
3. Les microbes dont le voisinage est composé de quatre cases vides meurent. Il en va de même pour ceux dont le voisinage comprend une ou trois cases occupées.
4. Il ne se produit aucun changement dans les cases qui ne satisfont pas aux conditions des règles 2 et 3.

après avoir introduit les lignes qui composent le programme, appuyez sur P0, affichez une colonie de votre choix, puis pressez sur la touche EXE.

Quelques 17 secondes pour que ce petit monde évolue... et vous pouvez alors voir l'aspect de la nouvelle colonie, qui devient à son tour colonie de départ.

Pour continuer le jeu : EXE, et ainsi de suite, jusqu'à ce que la boucle soit bouclée (on revient à une colonie que l'on a déjà vue), ou bien jusqu'à ce que mort s'ensuive.

□ Jean-Michel Muratet

Une vie où n'interviennent que les lois mathématiques...

001 HLT	078 MR 6	154 Min 0
002 Min.F	079 +	155 GOTO 5
003 1	080 MR 8	156 LBL 5
004 0	081 +	157 1
005 Min 0	082 MR 9	158 0
006 LBL 1	083 -	159 M-0
007 MR 0	084 Min.7	160 INV DSZ
008 -	085 MR 6	161 GOTO 6
009 1	086 +	162 MR.0
010 -	087 MR 7	163 INV X=0
011 INV 10 ^x	088 +	164 GOTO 7
012 INV 1/x	089 MR 9	165 2
013 x	090 +	166 Min F
014 MR.F	091 MR.0	167 MR.F
015 -	092 -	168 INV X=F
016 INV INT	093 Min.8	169 GOTO 8
017 ÷	094 MR 7	170 4
018 1	095 +	171 Min F
019 0	096 MR 8	172 MR.F
020 -	097 +	173 INV X=F
021 INV FRAC	098 MR.0	174 GOTO 8
022 x	099 -	175 GOTO 9
023 1	100 Min.9	176 LBL 7
024 0	101 MR 8	177 2
025 -	102 +	178 Min F
026 INV IND	103 MR 9	179 MR.F
027 Min 0	104 -	180 INV X=F
028 INV DSZ	105 Min.F	181 GOTO 8
029 GOTO 1	106 9	182 3
030 MR 2	107 Min 0	183 Min F
031 +	108 LBL 6	184 MR.F
032 MR 3	109 INV IND	185 INV X=F
033 -	110 MR 0	186 GOTO 8
034 Min.1	111 INV X=0	187 GOTO 9
035 MR 1	112 GOTO 2	188 LBL 8
036 +	113 1	189 1
037 MR 3	114 0	190 Min.0
038 +	115 M+0	191 INV SAC
039 MR 4	116 2	192 9
040 -	117 Min F	193 Min 0
041 Min.2	118 INV IND	194 GOTO 0
042 MR 1	119 MR 0	195 LBL 9
043 +	120 INV X=F	196 0
044 MR 2	121 GOTO 4	197 MIN.0
045 +	122 4	198 INV SAC
046 MR 4	123 Min F	199 9
047 +	124 INV IND	200 Min 0
048 MR 5	125 MR 0	201 GOTO 0
049 -	126 INV X=F	202 LBL 0
050 Min.3	127 GOTO 4	203 MR 0
051 MR.1	128 GOTO 3	204 -
052 +	129 LBL 2	205 1
053 MR 5	130 1	206 -
054 +	131 0	207 Min 2
055 MR 6	132 M+0	208 1
056 -	133 2	209 0
057 Min.4	134 Min F	210 M+0
058 MR 3	135 INV IND	211 MR 2
059 +	136 MR 0	212 INV 10 ^x
060 MR 4	137 INV X=F	213 x
061 +	138 GOTO 4	214 INV IND
062 MR 6	139 3	215 MR 0
063 +	140 Min F	216 -
064 MR 7	141 INV IND	217 M+7
065 -	142 MR 0	218 1
066 Min.5	143 INV X=F	219 0
067 MR 4	144 GOTO 4	220 M-0
068 +	145 GOTO 3	221 INV DSZ
069 MR 5	146 LBL 3	222 GOTO 0
070 +	147 0	223 9
071 MR 7	148 INV IND	224 INV 10 ^x
072 +	149 Min 0	225 X
073 MR 8	150 GOTO 5	226 MR.0
074 -	151 LBL 4	227 -
075 Min.6	152 1	228 M+7
076 MR 5	153 INV IND	229 MR 7
077 +		

Trouvez le bon mot : un jeu alphanumérique sur PC 1211/TRS 80 pocket

```

10: "Z"Q=1370:Q=
Q-INT Q
20: M=44+INT 6Q:
L=A(M):A(M)=
A(50)
30: FOR K=50 TO P
-1:A(K)=A(K+
1):NEXT K:A(K
P)=L
40: GOSUB 160:N=
0
50: BEEP 1:L=Q:N
=N+1
60: "F"INPUT "LE
TTRE 1 ? ";A
$
70: INPUT "LETTRE
2 ? ";B$
80: INPUT "LETTRE
3 ? ";C$
90: INPUT "LETTRE
4 ? ";D$
100: INPUT "LETTRE
5 ? ";E$
110: FOR K=1 TO 5:
IF A$(K)=A$(
11-K)LET L=L
+1
120: NEXT K:IF L=
5BEEP 7
130: PRINT N;" ";
A$;B$;C$;D$;
E$;" ";L
140: IF L<5GOTO 5
0
150: END
160: FOR K=6 TO 10
:O=INT (L/10
0)
170: A$(K)=A$(L-1
00*O):L=O:
NEXT K:
RETURN
200: "S"PRINT "C1
A 015 REC
210: PRINT # "MOT"
:A(16):END
300: "L"PRINT "C1
A 015 PLAY
310: INPUT # "MOT"
:A(16):END
400: "X"INPUT A$,
B$,C$,D$,E$
410: PRINT A$;B$;
C$;D$;E$
420: L=0: FOR K=1
TO 5:N=18
430: IF A$(K)=A$(
N)GOTO 450
440: N=N+1:GOTO 4
30
450: L=100L+N:
NEXT K
460: A(P+1)=L:P=P
+1:BEEP 1:
GOTO 400
500: "V"INPUT M
510: L=A(M):GOSUB
160
520: PRINT M;" ";
J$;I$;H$;G$;
F$;M=M+1:
GOTO 510

```

Il y a beaucoup de mots de cinq lettres en français, mais la plupart sont très corrects.

Voilà l'occasion de faire un jeu de mots sur votre micropoche.

■ Le BASIC Sharp du PC 1211 est assez mal loti quant au traitement des chaînes de caractères. Il permet seulement d'en comparer deux et de déterminer si elles sont ou non identiques. Un point, c'est tout. C'est mieux que rien, il est vrai, mais c'est tout de même un peu court quand on veut l'utiliser pour jouer sur les mots.

Peu importe : le plaisir de la programmation consiste en grande partie à contourner des obstacles, à aboutir malgré tout à une solution qui n'était pas évidente au départ. Si la plupart des instructions qui permettent de traiter les chaînes de caractères font défaut sur le PC 1211/TRS 80 pocket, on parvient tout de même à obtenir ce que l'on veut.

A preuve, ce programme de jeu qui vous proposera de deviner un mot de cinq lettres tiré au hasard dans une liste modifiable que la machine conserve en mémoire.

A chaque mot que vous propose-

Chaque mot est épilé par le programme et comparé avec celui que le joueur propose.

Si la liste des mots utilisés par le programme est longue, il est utile de pouvoir en faire un classement alphabétique. En effet, il arrive que l'on veuille introduire un mot supplémentaire sans savoir s'il fait déjà partie de la liste. On utilise alors le petit programme suivant :

```

10: " C " FOR j = 45 TO P : i = J - 1 : C = A (j)
20: IF A (i) < = C GOTO 40
30: A (i+1) = A (i) : i = i - 1 : IF i>43
GOTO 20
40: A (i+1) = C : NEXT J : BEEP 9 : END

```

Avec 84 mots, le classement peut durer plus de 20 minutes. Une fois que le tri a été effectué, il ne reste plus qu'à exécuter le programme Vérification, et à restituer bien entendu les quatre premières lignes du programme de jeu.

rez, le micropoche vous répondra en indiquant le nombre de lettres dans votre essai qui sont identiques, prises une par une, à la lettre qui occupe la même position dans le mot mystérieux. Pour bien me faire comprendre, je vous invite sans attendre à jouer une partie dont le résultat est connu d'avance.

Imaginons que la solution soit TABLE. Si l'on propose ARBRE, on obtient en réponse : 1. ARBRE 2., c'est-à-dire qu'à notre premier essai nous avons deux lettres bien placées (la chance nous sourit...). Ce sont bien évidemment la troisième lettre, B, et la cinquième, E, qui se retrouvent dans TABLE aux mêmes positions respectives. Mais à cette phase du jeu, comme on ne connaît pas encore la solution, on ne sait pas que ce sont ces deux lettres-là.

En essayant maintenant AIGLE, nous obtenons 2. AIGLE 2. Nous apprenons donc que deux lettres sont de nouveau bien placées ; on peut penser qu'il s'agit de A (1^{re} position) et de E (5^e position). Troisième essai : ACIDE. La réponse : 3. ACIDE 1. nous détrompe, puisqu'il n'y a qu'une seule lettre à sa place. On essaie alors successivement :

4. NOBLE 3.
4. SABLE 4.
6. TABLE 5.

C'est par conséquent le mot TABLE qu'il fallait découvrir, et nous avons gagné en 6 coups : une prouesse...

Le programme est composé de 5 parties qui sont chacune repérées

Trouvez le bon mot :
un jeu alphanumérique
sur PC 1211
et sur TRS 80 pocket



par une lettre et qui s'utilisent en mode DEF :

- " Z ", le jeu proprement dit, des lignes 10 à 170,
- " S ", séquence d'enregistrement sur cassette de la liste des mots, lignes 200 et 210,
- " L ", lecture de la liste des mots depuis une cassette, lignes 300 et 310,
- " X ", qui permet de créer une liste en introduisant des mots dans la mémoire du micropoche, lignes 400 à 460,
- " V ", enfin, qui permet de vérifier quels sont les mots de la liste, lignes 500 à 520.

_____ Codons _____
 _____ les lettres _____
 _____ en chiffres _____

Comme le BASIC Sharp du poquette ne permet pas d'isoler une partie du contenu d'une mémoire alphanumérique, il a fallu se livrer à un petit travail de codage pour que les mots soient transcrits sous forme numérique. C'est grâce à cette astuce que l'on parvient à simuler les fonctions de traitement de chaînes.

Bien entendu, ce code n'a rien de secret, et il est même très simple. Les 17 premières mémoires (A à Q)

étant utilisées par ailleurs, A est codé 18, B est codé 19, C est codé 20, etc. jusqu'à Z dont le code est 43. Ainsi " numérisé ", le mot *table* devient :

T A B L E
 37 18 19 29 22

soit un nombre de 10 chiffres, 3718192922, qui n'occupe qu'un seul registre numérique et que l'on peut facilement décoder en un mot de 5 lettres.

Le nombre 18 correspondant à la lettre A, 19 à B, 20 à C, etc., on rangera dans les mémoires A\$ (18) la lettre A, A\$ (19) la lettre B, et ainsi de suite jusqu'à Z qui sera stocké dans la variable A\$ (43).

En découpant par tranches de deux chiffres le nombre qui représente le mot codé, on obtiendra directement les indices des variables contenant les lettres qui le composent. Dans notre exemple, 3718192922 sera décodé ainsi :

37 → A\$ (37) → T
 18 → A\$ (18) → A
 19 → A\$ (19) → B
 29 → A\$ (29) → L
 22 → A\$ (22) → E

Avant de jouer, il va de soi qu'il faudra ranger une fois pour toutes les 26 lettres de l'alphabet dans les variables A\$ (18) à A\$ (43). A cette

fin, on pourra s'aider d'un petit programme tel que

600 : FOR K = 18 TO 43

610 : INPUT A\$(K)

620 : NEXT K

630 : END (facultatif)

Après avoir demandé RUN 600, on aura seulement à réciter l'alphabet : A, Enter, B, Enter, C, Enter, ..., Z, Enter. Une fois que ce sera chose faite, on effacera les lignes 600 à 630 de manière à conserver le plus possible de mémoire disponible.

Les mots, codés sous forme de nombres, seront enregistrés dans les variables A (44) et suivantes jusqu'à saturation de la mémoire, si l'on veut. Avec le programme tel qu'il est, on peut jouer sur 84 mots : variables A (44) à A (127), ce qui est amplement suffisant. Mais il est possible de rajouter encore 25 mots à la liste en ne gardant du programme que les lignes 10 à 310, les parties " X " et " V " étant détachées de l'ensemble et exploitées séparément (elles ne sont pas utiles pendant le déroulement du jeu proprement dit).

_____ Définir _____
 _____ les règles _____

En ce qui concerne le choix des mots de 5 lettres, pour éviter toute contestation lors des parties (certaines sont acharnées !), on a tout intérêt à se fixer par avance les règles qui valent pour ce type de jeu : tel dictionnaire fait foi, tout nom doit être au singulier, tout verbe à l'infinitif, on s'interdit les noms propres, etc.

C'est le programme " X " qui permet d'entrer le vocabulaire avec lequel on jouera. En premier lieu, on affecte à la variable P la valeur 43 en tapant P=43 Enter, puis, en mode DEF, on demande SHIFT X et le poquette attend l'introduction des mots, lettre par lettre (ligne 400) : chaque pression sur une touche doit donc être suivie d'Enter. Lorsque 5 caractères ont été introduits, la ligne 410 affiche le mot — ou l'imprime sur la CE-122 si cette dernière est connectée. Les lignes 420 à

450 effectuent alors le codage et l'on retourne en 400 pour l'introduction d'un autre mot. Le tout ne demande qu'une vingtaine de secondes.

Si l'on commet une faute de frappe ou si l'on estime avoir entré suffisamment de mots, on appuie sur BREAK. Il est toujours possible par la suite d'ajouter d'autres mots à la liste déjà mémorisée : il suffit de demander une nouvelle fois, en mode DEF, SHIFT X. C'est le code d'erreur n° 4 qui vous signalera que la mémoire de votre micro poche est pleine comme un œuf.

— Une comparaison — — à la lettre près —

Quelques remarques en passant sur la façon dont s'effectue le codage aux lignes 420 à 450. Nous conserverons le même exemple, le mot TABLE. Lors de la première exécution de la boucle, A\$ (K) n'est autre que la variable A\$ (1), autrement dit A\$, et elle contient la lettre T. Ce dernier caractère est comparé dans un premier temps au contenu de la variable A\$ (N), alias A\$ (18), c'est-à-dire " A " ; puis la comparaison s'effectue avec le contenu de la mémoire A\$ (19) (B), A\$ (20),... jusqu'au moment où il y a égalité : N est alors égal à 37 (c'est A\$ (3) qui contient la lettre T, initiale de *table*). On passe alors à la ligne 450 : $L = 100L + N$, ce qui équivaut à $L = (100 \cdot 0) + 37$, soit 37. Avec A\$ (2), B\$ donc, qui contient A, la deuxième lettre de *table*, on obtient $N = 18$, puis, à la ligne 450, $L = 100 \cdot 37 + 18 = 3718$. La troisième lettre, B, donne $N = 19$ et $L = 100 \cdot 3718 + 19 = 371819$. De même pour les deux dernières lettres, et l'on finit par obtenir $L = 3718192922$.

A la ligne 460, P est le numéro de la mémoire contenant le dernier mot de la liste. Le code numérique L du dernier mot entré est rangé dans la mémoire P + 1, si elle est disponible, auquel cas on passe au mot suivant : GOTO 400.

Le programme " V " (comme Vérification) est utile surtout lorsqu'on dispose de l'imprimante : il permet d'obtenir la liste des mots sur lesquels repose le jeu. La ligne 500 demande quel est le numéro du mot où l'on veut que la liste commence ; pour l'avoir in extenso, il faut donc répondre 44.

Comme on peut le voir, ce programme de vérification appelle le sous-programme des lignes 160 et 170 qui effectue le décodage des nombres en lettres. Ce décodage se fait *de droite à gauche* de la façon suivante : si $L = 3718192922$, à la première exécution de la boucle K, la fin de la ligne 160 conduit à ranger 37181929 dans la variable O. En effet, $L/100 = 37181929.22$ dont la partie entière est 37181929. A la ligne 170, l'expression $L - 100 O$ qui sert d'indice à une variable est donc égale à la partie décimale de $L/100$, soit 22. En A\$ (K), c'est-à-dire A\$ (6) ou encore F\$, est donc copié le contenu de la variable A\$ (22), autrement dit la lettre E qui sera la cinquième et dernière lettre du mot. On donne alors à L une nouvelle valeur : $L = O$, par conséquent 37181929, et l'on parcourt une nouvelle fois la boucle pour ranger fina-

réintroduire manuellement après que l'on ait effacé la mémoire du micro poche (pour y introduire un programme de calcul de taux d'intérêt par exemple).

Pour son bon fonctionnement, on veillera à ce que l'imprimante soit débranchée, faute de quoi l'ordre PRINT de la ligne 200 serait exécuté instantanément, tout comme celui de la télécommande du magnétophone : il y a gros à parier que vous n'auriez pas le temps d'appuyer sur les touches *play* et *record* de votre magnétophone...

En mode DEF donc, et l'imprimante déconnectée, il vous suffira de demander SHIFT S et l'affichage vous indiquera
C1 A 015 REC
Ce qui signifie :
C1 A : cassette n° 1, face A
015 : compteur du magnétophone
REC : magnétophone en position d'enregistrement.

Si tout est prêt, il ne vous reste plus qu'à appuyer sur Enter. Le nom du fichier est " MOT ", mais vous pouvez bien entendu choisir celui qui vous plaît en modifiant la ligne 210. La même remarque vaut d'ailleurs pour le message affiché par la ligne 200.

Comme on le verra plus loin, l'ordre des mots à l'intérieur de la liste est modifié à chaque partie. Le mot à deviner est choisi au hasard dans le début de la liste puis placé à la fin de telle sorte qu'il est impossible de tomber deux fois de suite sur le même mot.

Pour charger les mots depuis la cassette, la procédure est analogue : en mode DEF, on demande SHIFT L (comme Lecture) et,

**Vous avez bien sûr
toute latitude pour
modifier cette liste
à l'envi**

Quelques mots		
ASSEZ	CLOWN	CHAOS
HOTEL	SAVON	SIROP
GIGOT	STOCK	NUAGE
VINGT	PIZZA	REBUS
JOKER	DOIGT	HABIT
LIVRE	GRAND	HAMAC
WAGON	TALUS	LOYAL
BRAVO	TUYAU	GALOP
ECLAT	AIEUL	FRANC
CHAMP	CORPS	IMAGE
HIVER	IGLOO	JAUNE
CHIEN	RADIO	HAREM
CYGNE	SABRE	NOYAU
EMAIL	MYTHE	STADE
OUEST	TABLE	DEUIL
TWIST	CYCLE	FAKIR
LAMPE	ALIBI	TOTEM
STYLO	TONUS	ACHAT
FLASH	HYMEN	AUSSI
KAYAK	TABAC	AORTE
OBJET	VOYOU	MUSEE
OASIS	KYSTE	DIESE
VEXER	YACHT	APHTE
USUEL	BIBLE	BEMOL
HEROS	ENVOL	EXTRA
ETHER	HUTTE	REJET
PIANO	MATCH	ISSUE
PHOTO	CRAWL	MOYEU

lement en A\$ (7) le contenu d'A\$ (29), la lettre L, quatrième lettre du mot. Le même processus se répète pour les lettres B, A, T qui sont placées dans les mémoires 8, 9 et 10.

Pour utiliser le programme " S ", toujours en mode DEF, il est indispensable de disposer d'une interface cassette et d'un magnétophone. Ce programme permet de sauver la liste des mots et il évite donc d'avoir à la

l'imprimante étant déconnectée, l'affichage de " CA1 015 PLAY " rappelle où se trouve l'enregistrement du fichier et signale qu'il faut enfoncer la touche PLAY avant celle d'Enter...

L'enregistrement et le chargement du programme proprement dit ne peuvent s'obtenir que par l'intermédiaire des commandes directes CSAVE " xxx " et CLOAD " xxx " auxquelles on aura éventuellement

**Trouvez le bon mot :
un jeu alphanumérique
sur PC 1211
et sur TRS 80 pocket**



**Il faut parfois
de longues déductions
pour retrouver un
mot très commun**

dédiées deux touches de réservation.

Venons-en au jeu (lignes 10 à 170). Comme chacun — ou presque ? — le sait, le BASIC du PC 1211 est dépourvu de la fonction RND. On est donc contraint de se fabriquer soi-même sa loterie (*loterie* : générateur de nombres aléatoires). Voici comment fonctionne celle qui est à l'œuvre dans ce programme :

- à la ligne 10, Q est un nombre quelconque compris entre 0 et 1 et dont la valeur change au petit bonheur la chance entre chaque partie ;
- la première fois que l'on fait tourner le programme, il faut donner à Q une valeur comprise entre 0 et 1 avec au moins 6 décimales (par exemple $Q = .295147$) ;
- on a alors successivement :
 $Q = 137 Q = 137 * Q$ soit
 $Q = 40.435139$
 $Q = Q - \text{INT } Q$ soit
 $Q = 40.435139 - 40$ et donc
 $Q = 0.435139$

- cette nouvelle valeur de Q sera conservée pour la partie suivante ;
- à la ligne 20, Q est tel que $0 < Q < 1$
- toujours à la ligne 20, $6Q$ est tel que $0 < 6Q < 6$, et $\text{INT } 6Q$ est un nombre entier compris entre 0 (inclus) et 5 (inclus) ; on peut donc affirmer que $M = 44 + \text{INT } 6Q$ affecte à la variable M un nombre entier plus grand que 43 et plus petit que 50.

Si nous poursuivons notre exemple, nous voyons que notre loterie nous conduit à la mémoire n° 46 :

$Q = 0.435139$
 $6Q = 2.610834$
 $\text{INT } 6Q = 2$
 $M = 44 + 2 = 46$

Le mot à découvrir (variable L) sera donc celui dont le code numérique est contenu dans la mémoire 46.

À la même ligne 20 nous voyons que A(M), après avoir été recopié en L, change à son tour de contenu : $A(M) = A(50)$. On remplace donc le contenu de la variable qui a été tirée au sort par celui de la variable A(50) et l'on passe à la ligne 30 pour faire le ménage. Le mot retenu devient le dernier de la liste : $A(P) = L$ après que tous les codes contenus dans les variables A(51) à A(P-1) aient été transférés dans la variable immé-

diatement précédente : $A(K) = A(K+1)$.

La variable N qui est augmentée de l'unité à chaque passage ligne 50, comptabilise le nombre d'essais. Toujours à la ligne 50, la variable L est maintenant utilisée pour totaliser le nombre de lettres bien placées.

Des lignes 60 à 100, le programme demande, lettre par lettre, quel est le mot que le joueur propose puis il compare (110-120) si la première lettre est identique à celle du mot à découvrir, si la deuxième est bonne, et la troisième, etc. Quand les cinq lettres sont bonnes, c'est gagné, mais le plus souvent, la ligne 130 vous apprend combien de tentatives infructueuses vous avez déjà effectuées (N), quelle est votre dernière proposition mauvaise (A\$, B\$, C\$, D\$, E\$) et combien de lettres y sont à leur place (L).

_____ N'oubliez pas _____
_____ l'alphabet _____

Quand vous aurez chargé le programme et vérifié que vous n'y avez pas introduit de bogues(s), n'oubliez pas de ranger l'alphabet, lettre par lettre, dans les variables A\$(18) à A\$(43) et de donner à P la valeur 43 avant d'introduire les mots avec lesquels vous allez jouer. Enfin, avant d'entamer votre première partie, n'oubliez pas d'initialiser la loterie en affectant à Q une valeur comprise entre 0 et 1 avec au moins six décimales.

Une fois que le programme a démarré, c'est très simple, et l'on s'aperçoit qu'il n'y a rien de tel que de jouer avec les mots... Au début, il vous faudra peut-être plus de vingt essais avant de réussir, mais c'est tant mieux : vous allez faire de grands progrès ! Il se peut même que vous fassiez des erreurs de frappe en proposant vos solutions. Dans ce cas, appuyez sur BREAK puis sur SHIFT F (F comme fatigue, faux, etc.). C'est un peu à cause de moi : je crois que je vous ai un peu surmené avec toutes mes explications.

□ Jean Biron

Balade dans la bibliothèque de base

Pourquoi réinventer la roue... ou la poudre ? C'est déjà chose faite. La TI 58/59 est pourvue d'origine d'une grande quantité de programmes utilitaires. Inutile de les réécrire. Il suffit d'aller les chercher.

■ Pour les utilisateurs de TI 58/59, la bibliothèque de base (ou tout autre module préprogrammé) présente d'évidentes qualités dans des domaines variés. Cette mystérieuse petite boîte noire, disponible dès que la machine est allumée, a le côté confortable et séduisant du « pas besoin de programmer, c'est déjà fait ».

Si l'on s'en tient au manuel de Texas Instruments, on a déjà un large choix de possibilités ; mais avec un soupçon de curiosité et un chronomètre, on découvre des utili-

sations qui ne manqueront pas d'intéresser les programmeurs à court de lignes... ou de temps.

L'intérêt immédiat de la présence du module est de doter la calculatrice d'une puissance peu commune dans le mode « règle à calcul » : on ignore la touche « LRN », et l'on a cependant 25 programmes disponibles de suite.

Une autre possibilité offerte par la bibliothèque est l'appel d'un programme du module au cours de l'exécution du programme principal : économie et sécurité (il est peu probable que le pointeur se « plante » dans le logiciel préfabriqué). Pour les curieux, la séquence Op 99 permet le transfert d'un programme de bibliothèque en mémoire-programme. On peut alors lister celui-ci (tant mieux si vous avez l'imprimante, sinon : LRN ; SST ; SST ; SST ; ...).

Si le manuel mentionne cette possibilité d'exploration de la bibliothèque, il est pour le moins avare de commentaires sur le profit que l'on peut en retirer.

Quittons l'utilisation « normale » pour nous pencher sur ce logiciel occulte. L'examen de listes de programmes de bibliothèque, sans chercher à comprendre globalement « comment ça marche », fait apparaître des séquences d'instructions qui pourraient avantageusement prendre place dans un autre contexte.

Exemple :

009	76	LBL
009	25	CLR
010	29	CP
011	06	6
012	42	STO
013	01	01
014	00	0
015	72	ST*
016	01	01
017	97	DSZ
018	01	01
019	00	00
020	15	15
021	92	RTN

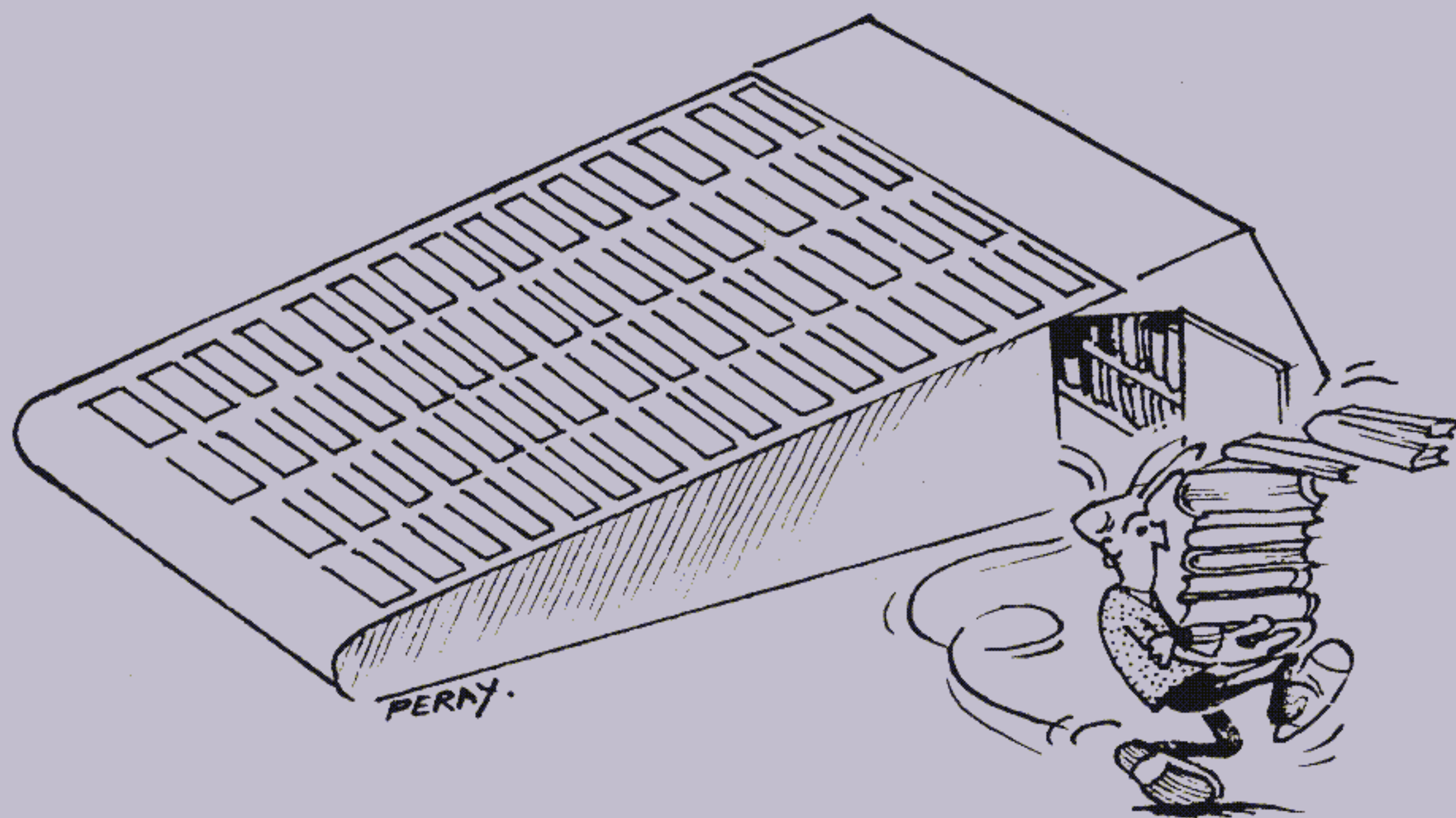
En provenance directe du Programme 01, cette séquence efface les registres R₀₁ à R₀₆ et T. Ça peut servir un jour ou l'autre ; alors pourquoi rentrer 12 lignes dans notre mémoire souvent limitée, quand on peut obtenir le même résultat avec seulement 4 lignes ?

Pgm 01 SBR CLR.

Dans le même ordre d'idées, voici un programme qui efface les registres R₀₁ à R_n :

Pgm 01 n SBR 012.

A l'origine, les programmes du module sont conçus pour la résolution de problèmes bien définis. Par exemple, exécuter la séquence : Pgm 01 SBR = a pour effet d'opérer la vérification du bon fonctionnement de la calculatrice (diagnostic). Cela, c'est ce qui est prévu. Mais



rien ne nous empêche d'appeler un programme de bibliothèque à l'aide d'une adresse numérique ou d'une étiquette quelconque, pourvu qu'elles existent dans ledit programme. Ce n'est plus l'aspect *macroscopique* qui nous intéresse, mais l'aspect *microscopique*.

———— Il faut ————
 ———— cependant ————
 ———— être prudent ————

Bouillonnement d'impatience devant un affichage muet : que fait-elle ? Elle « cherche », elle « réfléchit », elle « se trompe » ? Aucun risque de ce côté ; si erreur il y a, c'est vous le responsable, elle ne fait qu'exécuter. La solution consiste à appuyer sur R/S pour stopper l'exécution et chercher à comprendre ce qui a bien pu se produire. Le hic, c'est que R/S n'a aucun effet sur un programme s'exécutant en bibliothèque ; la seule façon de faire cesser cette cogitation incontrôlée est d'appuyer sur RST. Mais voilà, si votre programme s'exécute tantôt dans la bibliothèque, tantôt dans la mémoire-programme, vous risquez d'appuyer sur R/S, puis RST, puis R/S... jouant ainsi à cache-cache avec le pointeur. Quand la machine infernale est enfin arrêtée, il est difficile de savoir ce qui s'est passé car le contrôle d'erreur est impossible en bibliothèque (essayez d'intercaler des R/S dans le module !).

Avant de choisir une séquence de la bibliothèque pour votre usage personnel, il est nécessaire de vérifier que l'exécution se fera sans surprises. Le pointeur ne peut quitter la bibliothèque que s'il rencontre l'instruction RTN ; le déroulement se poursuit alors dans la mémoire-programme.

———— Et la vitesse ————
 ———— d'exécution ? ————

On pourrait penser que l'appel d'un programme de bibliothèque dure un temps non négligeable, son exécution aussi, et que si l'on gagne des lignes, on se retrouve perdant en ce qui concerne la vitesse d'exécution. Il n'en est rien ; un programme de bibliothèque s'exécute beaucoup plus rapidement que le même programme transféré en mémoire-programme ; donc, gain de place et gain de temps. Pour vous en convaincre, essayez ceci, montre en main :

Calcul de 69 !

Bibliothèque : Pgm 16 69 A
 C. Temps d'exécution de C : environ 13 secondes.

Mémoire-programme : Pgm 16 Op 09 RST 69 A C.

Temps d'exécution de C : environ 26 secondes, soit une durée double pour le programme « transféré ».

Tout cela est bien joli quand on peut trouver le programme en question dans la bibliothèque. Mais comment savoir si la séquence recherchée existe dans le module ? Réponse sans illusion : aucun moyen rationnel, et précision utile : la liste complète de la bibliothèque vous donnerait une bande de papier d'environ... 22 mètres. Renonçons donc à la méthode qui consisterait en une exploration systématique des quelque 5 000 lignes, pour y trouver les 12 qui feraient votre bonheur.

Prenons plutôt le problème à l'envers. Voici une méthode assez empirique qui occupera vos (rares) instants désœuvrés : en listant un programme de la bibliothèque, notez les suites d'instructions qui, selon la formule consacrée, « pourraient vous intéresser ». Rappel : il faut que la suite se termine par RTN (code combiné, correspondant à INV SBR), sinon le pointeur risque de se perdre dans les dédales de la bibliothèque pour en revenir avec des résultats bizarroïdes.

Un cahier ou un classeur est bien utile pour noter tout cela car on rencontre des tas de choses « pas bêtes du tout » dans le logiciel de Texas, et le cahier se remplit vite si vous pensez comme moi : « ça peut toujours servir ».

A côté de la liste du morceau de programme, vous pourrez utilement noter la fonction de cette suite d'instructions et, éventuellement, des applications personnelles où vous serez susceptible de l'utiliser.

Ce travail, un peu fastidieux j'en conviens, donne de bons résultats pour des applications classiques : effacement de registres, stockage indirect de données, etc. Il permet des économies souvent peu importantes (une dizaine de lignes), mais, en revanche, fréquemment utilisables. Vous en trouverez quelques exemples dans l'encadré ci-contre.

Cette liste d'applications est loin d'être limitative. A vous de trouver votre bonheur dans le petit module noir... Il ne vous reste plus qu'à

Exemple 1 :

093	99	PRT
094	72	ST*
095	01	01
096	92	RTN
097	32	X⇌T
098	01	1
099	44	SUM
100	01	01
101	32	X⇌T
102	61	GTO
103	00	00
104	93	93

Appel : x Pgm 02 SBR 097.

Fonction :

Incrément de R₀₁

Stockage de x dans R_{ind 01}

Impression de x

Avance papier

Utilisation : stockage et impression des coordonnées d'un vecteur. Pointeur : R₀₁

Exemple 2 :

860	01	1
861	44	SUM
862	01	01
863	44	SUM
864	04	04
865	73	RC*
866	01	01
867	99	PRT
868	92	RTN

Appel : x Pgm 02 SBR 860.

Fonction :

Incrément de R₀₁

Incrément de R₀₄

Impression de R_{ind 01}

Utilisation : rappel et impression des coordonnées d'un vecteur. Pointeur R₀₁. Attention, R₀₄ est aussi incrémenté.

Exemple 3 :

000	76	LBL
001	11	A
002	48	EXC
003	04	04
004	48	EXC
005	03	03
006	43	RCL
007	04	04
008	98	ADV
009	99	PRT
010	92	RTN

Appel : x Pgm 03 A.

Fonction :

Stockage de R₀₄ dans R₀₃.

Stockage de x dans R₀₄

Avance papier

Impression de x

Utilisation : Stockage dans R₀₃ et R₀₄ d'un vecteur à 2 coordonnées.

vous munir d'un stock de papier, d'un peu d'imagination, et vous disposerez d'une réserve presque gratuite pour tempérer les ardeurs de programmes trop lents ou... débordants.

□ Antoine Jennet

Sur le bout de la langue : la HP 41 C a elle aussi son mot à dire

Dans le présent numéro, on trouve un programme pour le poquette Sharp qui permet de jouer avec les mots. La HP 41 C est également dotée de fonctions alphanumériques...

■ Dans l'Op n° 2, Jacques Noe avait proposé le jeu du Chemin qui sortait vraiment de l'ordinaire. Ceux et celles qui se sont essayés à explorer la quatrième dimension ne l'ont sûrement pas oublié.

C'était un programme spécialement riche. En ce qui me concerne, j'en ai surtout retenu le sous-programme Lbl 11 qui permettait, grâce à la fonction ASHF de la HP 41 C, la décomposition d'une chaîne de caractères et la répartition des caractères dans des registres de données successifs.

C'est en exploitant cette idée que j'ai réussi à programmer un jeu qui s'apparente à l'option "mystère" du Speak and Spell de Texas Instruments. Comme quoi la HP 41 C est un véritable caméléon...

Le but du jeu est de deviner un mot en proposant l'une après l'autre les lettres qui pourraient composer ce mot. Le nombre d'essais malheureux est limité à neuf, mais on peut se faire aider par le micropoche en lui demandant d'afficher la première des lettres qui n'ont pas encore été trouvées; cet appel au secours coûte trois essais.

Initialement, les mots à découvrir sont placés, avec pour chacun l'indication de leur longueur, dans le programme et non pas dans des registres de données. Cette solution a le mérite de la simplicité et elle est, tout compte fait, économique en mémoire. On peut ainsi "caser" sans difficulté les quelque 180 mots du jeu. Il est d'ailleurs possible de

```

01*LBL *MYST-1*
02*LBL J
RCL 00 CLRG FRC ABS
X*0? GTO 00
"SEMPENCE ?" PROMPT

11*LBL 00
STO 00 "NOT-MYSTERE"
5.019

15*LBL 01
AVIEW CF IND X ISG X
GTO 01 "SELECTION"
AVIEW CF 29 FIX 0 9
STO 23 100 XEQ 08 130
* INT STO 02 "MYST-"
X<Y? GTO 07 "F4"
GTO 00

37*LBL 07
"12" 2 / FRC X*0?
GTO 00 XEQ 08 2 *
INT X=0? SF 05

50*LBL 00
ASTO 01 CLA 0

54*LBL 02
XEQ IND 01 FC?C 05
GTO 00 1 ST+ 02 RDN
GTO 02

62*LBL 00
+ STO 22 STO 24 7
X<Y? GTO 00 X=Y?
GTO 05 ASTO 19 ASHF
ASTO 20 CLA ARCL 19
"1*" SF 05 XEQ 05 14
STO 17 7 RCL 22 6 -
- CLA ARCL 20 ASTO 19

89*LBL 03
"1*" DSE X GTO 03 7
STO 15 GTO 06

96*LBL 00
X<Y -

99*LBL 04
"1*" DSE X GTO 04

103*LBL 05
8 STO 17 7 STO 15

108*LBL 06
1 ST- 17 ASTO 14 ASHF
ASTO IND 17 ARCL 14
DSE 15 GTO 06 FC?C 05
GTO "MYST-3" RTN

120*LBL 08
RCL 00 9821 * .211327
+ FRC STO 00 END

01*LBL "MYST-3"
11.02

03*LBL 17
0 STO IND Y RDN ISG X
GTO 17

09*LBL 06
CF 18 RCL 22 5 +
1 E3 / 6 + CLA

19*LBL 07
FS? IND X GTO 01
FS? 19 GTO 02 "1-"

25*LBL 08
ISG X GTO 07 10
RCL 22 X=Y? GTO 00 -

33*LBL 09
"1" DSE X GTO 09

37*LBL 00
"1" ARCL 23 FC?C 17
GTO 00 TONE 5 TONE 3

44*LBL 00
XEQ 05 FC? 16 GTO 00
PROMPT GTO "MYST-1"

50*LBL 00
RCL 22 STO 26 AON
CF 23 AVIEW

56*LBL 10
PSE FC? 23 GTO 10
ASTO X "?" ASTO Y
X*Y? GTO 03
"*** AIDE ***" AVIEW
SF 19 3 ST- 23 GTO 04

71*LBL 03
AOFF "RECHERCHE"
ARCL X AVIEW

76*LBL 11
RCL IND 26 X=Y? XEQ 12

80*LBL 16
RDN DSE 26 GTO 11
FC? 18 SF 17 FC? 18
DSE 23 GTO 06

89*LBL 15
RCL 22 "PERDU" 5.005
+

94*LBL 13
AVIEW TONE 5 SF IND X
DSE X GTO 13 0 STO 23
SF 16 GTO 06

104*LBL 12
RCL 26 10 + RCL IND X
X<Y RDN X*Y? GTO 00
RDN FS? 18 RTN
"DEJA PRIS" AVIEW RTN

119*LBL 00
RDN STO IND Z RCL 26
5 + SF IND X RDN
ISG 25 X<Y X

129*LBL 04
SF 18 RCL 23 X<=0?
GTO 15 DSE 24 GTO 16
XEQ 05 SF 16
"* VICTOIRE *" AVIEW
TONE 9 TONE 9 TONE 8
TONE 8 TONE 9 TONE 8
TONE 6 TONE 7 BEEP
GTO 06

150*LBL 01
5 - ARCL IND X 5 +
GTO 08

157*LBL 02
SF IND X 5 - CLA
ARCL IND X 10 +
ASTO IND X CF 19
GTO 06

168*LBL 05
RCL 25 X=0? RTN

172*LBL 14
BEEP DSE 25 GTO 14
END

```

Sur le bout de la langue : la HP 41 C a elle aussi son mot à dire

modifier le vocabulaire à condition de respecter la façon dont fonctionne le programme. L'important ici est de savoir que dans la partie "MYST-2", la liste des mots doit être organisée de telle sorte que chaque vocable puisse faire double emploi : il est d'une part un mot qui se suffit à lui-même (METRO ou POLES, par exemple), mais il peut également être l'une des deux parties d'un nouveau mot (pour poursuivre notre exemple, METRO et POLES donneront METROPOLES).

Les termes qui peuvent faire office de début de mot sont tous inscrits après les étiquettes de numéro pair et ils sont tous suivis de l'étiquette impaire où figure le mot qui peut éventuellement venir les compléter :

- étiquette paire : KILO,

- étiquette impaire suivante : METRES.

Pour introduire le dictionnaire "MYST-2", vous prendrez exemple sur les premières lignes et vous programmerez les Lbl dans l'ordre, chacun contenant le nombre de lettres du mot puis le mot sans oublier le RTN. Prenez garde d'introduire une fois sur deux un APPEND. Les mots vous sont indiqués dans l'encadré ci-dessous par groupe de deux pour faire apparaître leur unité.

Dans le programme MYST-1, si le nombre aléatoire produit par le sous-programme Lbl 08 (et placé en R 02 au début de chaque partie) est égal à zéro ou s'il est pair, c'est-à-dire si le mot choisi peut également faire office de début de mot, un deuxième tirage au sort intervient et décide si l'on se contente du terme

retenu (*gui* par exemple) ou si l'on opte pour la version "à rallonge" (*guimauve*). Notez bien que la "rallonge" (*mauve* en l'occurrence) est elle aussi un mot à part entière : ç'aurait très bien pu être le mot à deviner si le premier tirage au sort avait fourni un numéro impair.

Les mots difficiles sont placés dans une rubrique spéciale : le programme MYST-4 où l'on trouve des termes tels que *sphynx* ou *joyaux*. C'est dans cette liste que le micro-poche ira chercher de temps à autre une colle à vous proposer. Il le fera chaque fois que le premier nombre aléatoire sera supérieur à 99, ce dernier nombre représentant le label numérique le plus élevé adressable à partir de R 02 dans le programme MYST-3.

Pour modifier les programmes MYST-2 et MYST-4, il faudra donc vérifier — et corriger éventuellement — les nombres inscrits aux pas suivants :

- dans le programme MYST-1, le pas 026 doit contenir le numéro du dernier registre adressable indirectement dans MYST-2 par le contenu de R 02 augmenté de l'unité (ici $99+1 = 100$) ; quant au pas 028, il doit contenir le nombre total des labels adressables indirectement par R 02 dans MYST-2 et MYST-4 augmenté de 2 (ici $99+29+2 = 130$) ;
- dans le programme MYST-4, le pas 003 représente le nombre de labels utilisés par le programme MYST-2 (100).

Cette découpe générale du jeu en quatre tranches complètement séparées par des instructions END peut paraître assez déroutante mais il n'est pas dit que cela soit tout à fait inutile. Au contraire, les essais que j'ai effectués m'ont appris que, pour ce jeu, le temps d'exécution avec une version "programme unique" était nettement plus long.

Le programme principal est MYST-3. C'est lui qui se charge du jeu proprement dit. Si vous désirez le décortiquer, voici une indication qui devrait vous faciliter la tâche : chacune des lettres découvertes (registres R 11 à R 21) arme un indicateur binaire, de F 05 à F 14 : il

<pre> 01♦LBL "MYST-2" XEQ IND 02 RTN 04♦LBL 00 3 "ANA" RTN 08♦LBL 01 "GRAMME" RTN 11♦LBL 02 "BOU" RTN 14♦LBL 03 5 "BONNE" RTN 18♦LBL 04 4 "CHIC" RTN 22♦LBL 05 6 "FANERIE" RTN 26♦LBL 06 3 "DAM" RTN 30♦LBL 07 6 "NATION" RTN 34♦LBL 08 3 "GAZ" RTN </pre>	<p>Mots à introduire à partir du pas 39 ♦ LBL 10</p> <p>FER-MAGE ; GUI- MAUVE ; LECHE- FRITE ; MARE-CHALE ; VIN-AIGRE ; MONT- AGE ; LOIN-TAIN ; LONG-TEMPS ; PRE- CURSEUR ; QUEL- CONQUES ; ARC- EAU ; BAS-RELIEF ; CAME-LOTE ; CHASSE- LAS ; DENT-AIRE ; DOUCE-ATRE ; ECRIT- URES ; FAILLI-BLE ; FOU-LARD ; FORT- UNE ; IDE-ALITE ; HALE-TANT ; JUS- QUE ; VIS-AGES ; LAC- TAIRE ; MACON- NAGE ; METRO-PO- LES ; NOISE-RAIE ; OPE-RABLE ; PATE- NOTRE ; PORTE-MI- NES ; POSTE-RIEUR ; GRE-NOUILLE ; RAS- SEMBLER ; REG-EN- TER ; SON-NETTES ; PEMPE-RANCE ; UNI- FORME ; VINGT-AINE ; ALTER-NATIF ; MUR- MURES.</p>	<pre> 38♦LBL 09 4 "ELLE" RTN : : : 370♦LBL 92 4 "HOME" RTN 374♦LBL 93 3 "LIE" RTN 378♦LBL 94 4 "JOLIE" RTN 382♦LBL 95 4 "ESSE" RTN 386♦LBL 96 4 "KILO" RTN 390♦LBL 97 6 "METRES" RTN 394♦LBL 98 4 "OEIL" RTN 398♦LBL 99 3 "FLET" END </pre>
---	---	--

faut donc ne pas utiliser l'imprimante, faute de quoi l'on s'exposerait à des surprises.

Comme on va le voir dans la partie commentée qui suit, le joueur n'a droit qu'à 9 essais, et certains mots comptent jusqu'à 10 lettres, mais cela n'a rien de contradictoire. Seuls en effet sont comptabilisés les essais infructueux. Autre remarque : si la lettre proposée est plusieurs fois présente dans le mot-devinette, elle s'inscrit dans toutes les cases où elle figure ; il en résulte que les mots les plus longs ne sont pas nécessairement les plus difficiles à découvrir.

Pour suivre exactement la même partie que celle qui va être décrite, il faut, après avoir introduit les 4 programmes (size = 028), ranger 0.00006464 en R 00. Il est commode d'assigner MYST-1 à J : il suffira d'appuyer sur J pour entamer la partie, on évitera ainsi d'avoir à demander XEQ alpha MYST-1 alpha (9 touches...).

Si jamais le registre 00 qui contient le nombre utile au générateur de nombres aléatoires devait plus tard prendre la valeur zéro, le micro-poche afficherait " SEMENCE ? " et il faudrait alors réintroduire un nombre compris entre 0 et 1 en R 00 pour poursuivre le jeu.

Au début, l'affichage indique " MOT-MYSTERE " puis " SELECTION " et enfin une ligne composée de 3 à 10 tirets suivis par 9 (nombre d'essais que la machine nous alloue). Les tirets représentent la longueur du mot à découvrir. Dans notre exemple, cela donne :

— — — — — — — — — — 9

Nous allons donc devoir découvrir un mot de dix lettres, et nous n'avons droit qu'à neuf erreurs.

Le micropoche étant en mode alpha, introduisez la lettre E, et n'appuyez surtout pas sur R/S : le micropoche est simplement en attente de votre proposition sur une instruction PAUSE à l'intérieur d'une boucle. L'affichage indique maintenant " RECHERCHE E " puis, après un petit signal sonore, vous pouvez lire :

— — — — E — — E — E 9

Nous connaissons déjà trois lettres et nous n'avons pas entamé notre capital d'essais. Nous proposons A. Même processus que pour la lettre E, mais l'affichage final cette fois-ci présente :

— — — — E — — E — E 8

La lettre A ne fait donc pas partie

01♦LBL "MYST-4"	44♦LBL 09	84♦LBL 19
RCL 02 100 -	7 "EMBRYON" RTN	6 "SCHEMA" RTN
XEQ IND X 0 RTN		
08♦LBL 00	48♦LBL 10	88♦LBL 20
5 "YACHT" RTN	10 "HOCHQUEUE" RTN	7 "LUXUEUX" RTN
12♦LBL 01	52♦LBL 11	92♦LBL 21
5 "WAGON" RTN	5 "HYMNE" RTN	5 "POLKA" RTN
16♦LBL 02	56♦LBL 12	96♦LBL 22
7 "MYSTERE" RTN	4 "JAZZ" RTN	7 "RUGUEUX" RTN
20♦LBL 03	60♦LBL 13	100♦LBL 23
9 "RYTHMIQUE" RTN	6 "SPHYNX" RTN	6 "PARFUM" RTN
24♦LBL 04	64♦LBL 14	104♦LBL 24
6 "PUZZLE" RTN	5 "CHEIK" RTN	4 "ECHO" RTN
28♦LBL 05	68♦LBL 15	108♦LBL 25
8 "BLIZZARD" RTN	6 "JOYAUX" RTN	7 "CYCLONE" RTN
32♦LBL 06	72♦LBL 16	112♦LBL 26
7 "BIZARRE" RTN	7 "PSAUMES" RTN	8 "COURROUX" RTN
36♦LBL 07	76♦LBL 17	116♦LBL 27
8 "TYPHOIDE" RTN	6 "ZIGZAG" RTN	6 "THORAX" RTN
40♦LBL 08	80♦LBL 18	120♦LBL 28
9 "OLYMPIADE" RTN	5 "PIZZA" RTN	6 "REFLUX" RTN
		124♦LBL 29
		9 "PSYCHOSES" END

du mot, et cette erreur décrémente le compteur d'une unité.

Décidons de nous faire souffler une lettre : pressons sur la touche " ? " ; l'affichage indique " *** AIDE *** ", puis

H — — — E — — E — E 5

Nous connaissons maintenant la première lettre, mais cela nous a coûté trois essais. D'autre part, comme le micropoche, à chaque demande d'aide, ne donne que la première lettre inconnue, nous ne sommes pas certains qu'il n'y ait pas d'autres lettres H dans le mot. Appuyons sur H pour en avoir le cœur net :

H — — H E — — E — E 5

Bien joué ! Nous savons maintenant qu'il y a deux fois la lettre H, et nous avons toujours 5 essais. Essayons une nouvelle fois la touche " ? " pour obtenir :

H O — H E — — E — E 2

La troisième lettre du mot pourrait bien être C, n'est-ce pas ? Essayons C :

H O C H E — — E — E 2

Réfléchissons un peu... Cela ne vous dit rien ? Moi, j'ai trouvé, mais je n'ai aucun mérite, car je connais (et pour cause) la liste des mots. Le premier dictionnaire venu vous donnerait la réponse, mais faites un petit effort de mémoire. Si vous n'avez toujours pas d'idée, je vous suggère d'essayer la lettre U :

H O C H E — U E U E 2

Il ne vous reste plus qu'à proposer la lettre Q. L'affichage indique " *VICTOIRE* " pendant que se fait entendre une petite mélodie, puis il vous confirme qu'il s'agissait bien de *hochequeue*, le surnom de la bergeronnette, petit oiseau très amusant à observer. Vous avez gagné en 7 essais seulement ; (oui, c'est vrai, je vous ai peut-être un peu aidé...). Cela dit, si vous n'aviez pas trouvé le mot, le micropoche aurait donné la solution.

Maintenant, il ne vous reste plus qu'à faire tout seul une nouvelle partie en demandant XEQ alpha MYST-1 alpha ou en appuyant sur J si cette touche a été correctement assignée.

□ André Flédric

Votre bibliothèque informatique

Les ouvrages des éditions du P.S.I. sont répartis en quatre séries de difficulté croissante : - **Série verte** : initiation - **Série bleue** : perfectionnement - : approfondissement - **Série noire** : maîtrise de la technique.

Le langage ADA

par Daniel-Jean David

Langage moderne, ADA, créé par une équipe française, est promis à une grande diffusion sur toutes machines. Au moment où ce livre est écrit, il n'existe pas encore de compilateur ADA opérationnel. Le présent ouvrage donne les caractéristiques de ADA et le situe par rapport aux autres langages.

Série noire - 152 pages - 75 FF

par Daniel-Jean David

Après une information complète sur la programmation en langage APL, ce livre replace ce langage parmi les autres. La puissance d'APL est mise en évidence progressivement et de nombreux exemples d'applications sont traités. Plus de 60 exercices sont proposés et résolus.

- 128 pages - 65 FF

Programmer en Assembleur

par Alain Pinaud

Cet ouvrage constitue une introduction complète au langage machine, et à son frère l'assembleur, comprenant des exercices et des exemples. Bien qu'illustré par le code du Z 80, il sera d'une lecture tout aussi utile aux P.S.I. disposant d'un autre microprocesseur.

Série bleue - 144 pages - 75 FF

Programmer en Basic

par Michel Plouin

Ce livre a été écrit pour les utilisateurs d'ordinateurs individuels en particulier d'Apple II, TRS-80 et PET/CBM. Un répertoire Basic rend son utilisation

très pratique et facilite la transposition d'un programme écrit pour un P.S.I. sur un autre.

Série verte - 132 pages - 65 FF

Tome 1 - méthodes pratiques
par Jacques Boisgontier

Cet ouvrage s'intéresse à la programmation des applications utilisant des fichiers sur disquettes ou sur disques. La version de Basic retenue est le 5. de Microsoft fonctionnant sous CP/M. Les utilisateurs de TRS-80 et de P.S.I. à microprocesseurs Z 80 et TRS 80 sont donc directement intéressés.

- 144 pages - 75 FF

Tome 2 - programmes
par Jacques Boisgontier

Ce second tome est essentiellement consacré à des programmes, utilitaires comme le générateur de saisie d'écran ou le tri, rapide, de gestion comme la facturation ou la paie.

- 144 pages - 75 FF

Programmer en Fortran

par Daniel-Jean David

Destiné à l'apprentissage de la programmation en Fortran IV et Fortran Microsoft, ce livre comporte plus de 40 exercices résolus. Il fait le point sur l'intérêt du Fortran et sur son avenir sur les P.S.I. face aux autres langages. Il est complété d'une étude critique de la nouvelle norme Fortran 77.

Série bleue - 128 pages - 65 FF

Programmer en L.S.E.

par Stéphane Berche et Yves Noyelle

Cet ouvrage donne tous les éléments nécessaires pour l'utilisation d'un ordinateur programmable en LSE (langage français). L'ensemble exposé forme un tout cohérent permettant de s'initier au langage. Il est complété d'exemples pratiques et de plusieurs programmes opérationnels.

Série verte - 128 pages - 65 FF

par Daniel-Jean David
et Jean-Luc Deschamps

Ce livre contient un grand nombre de programmes-exemples et d'exercices résolus. Les déclarations Pascal concernant les types de données sont traitées avec précision et les notions générales sur les structures de données sont exposées.

- 160 pages - 75 FF

par Jean-Claude Barbance

Pour ceux qui ont déjà écrit plusieurs programmes et qui veulent s'attaquer à des réalisations plus ambitieuses. Une méthode, illustrée par trois exemples écrits en Basic :

- 1) Sous-programme de traduction d'un nombre en mots (101 = cent un) ;
- 2) Jeu du 421 ;
- 3) Comptabilité familiale.

- 164 pages - 75 FF

Comprendre les microprocesseurs

par Roland Dubois

Ce livre, qui est une introduction aux microprocesseurs, explique en détail mais d'une manière suffisamment générale, ce qu'est un microprocesseur, une mémoire ROM, une mémoire RAM, un coupleur. Cet ouvrage montre également comment associer ces différents circuits pour former un micro-ordinateur.

Série bleue - 128 pages - 65 FF

BON DE COMMANDE

OP 4

DESIGNATION	NOMBRE	PRIX
	TOTAL	

Les prix sont : taxes, emballage et port compris.
(par avion : ajouter 5 FF par livre)

Envoyer ce bon accompagné de votre règlement à
EDITIONS DU P.S.I.
41-51, rue Jacquard
BP 86 77400 Lagny-s/Marne
Tél. (6) 007.59.31.

NOM _____ PRENOM _____

rue _____ N° _____

Code post. | | | | |

Ville _____

LANGAGES



Le langage
ADA

NOUVEAU

Daniel-Jean David

Programmer
en
APL

NOUVEAU

Daniel-Jean David

Programmer
en
Assembleur

illustré avec le jeu d'instruction du Z-80

Alain Pinaud

Programmer
en
BASIC

Michel Plouin

Le **BASIC**
et ses
Fichiers

Tome 1
méthodes pratiques

Jacques Boisgontier

Le **BASIC**
et ses
Fichiers

Tome 2
programmes

NOUVEAU

Jacques Boisgontier

Programmer
en
FORTRAN

NOUVEAU

Daniel-Jean David

Programmer
en
LSE

Yves Noyelle
Stéphane Berche

Programmer
en
PASCAL

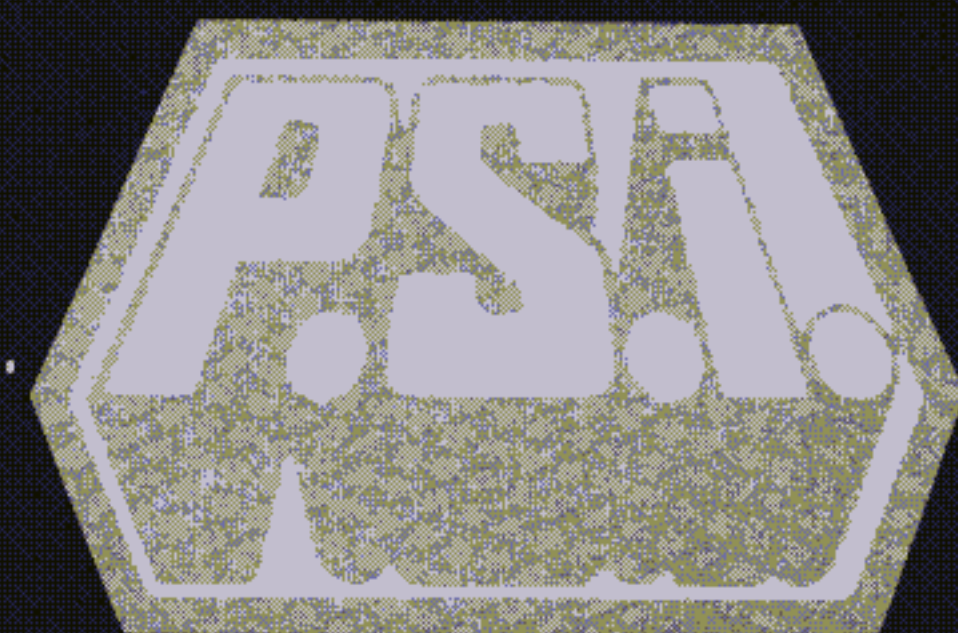
Daniel-Jean David
Jean-Luc Deschamps

Comment
Programmer

Jean-Claude Barbance

Comprendre
les
Microprocesseurs
et leurs circuits associés

Roland Dubois



Editions du P.S.I.
41-51, rue Jacquard
BP 86
77400 Lagny-s/Marne
Téléphone (6) 007.59.31

au Canada :
SCE Inc.
3449 rue Saint-Denis
Montréal Québec H2X3L1
Tél. : (514) 843.76.63

MATERIELS



La découverte
de
Applesoft
initiation au Basic de l'Apple

Dominique Schraen
Frédéric Lévy

La pratique de
l'APPLE II
Volume I - Basic Applesoft
Système Apple - Graphiques

Nicole Bréaud-Pouliquen

La pratique de
l'APPLE II
Volume II - périphériques
et gestion des fichiers

Nicole Bréaud-Pouliquen

La pratique
du **LX-500**
Volume I - présentation
apprentissage du Basic

Alain Séméteys
Francis Vasse

La pratique du
MZ.80 K
Volume I - le Basic - SP 5625

Jean-Pierre Lhoir

La découverte
du
P.E.T./C.B.M.

Daniel-Jean David

La pratique du
P.E.T./C.B.M.
Volume I - Périphériques et
Gestion des Fichiers

Daniel-Jean David

La découverte
du
PC-1211
NOUVEAU

Jean-Pierre Richard

La pratique du
TRS-80
Volume I - Architecture et Basic II

Pierre Giraud
Alain Pineud

La pratique du
TRS-80
Volume II - Compléments Basic II
Programmation du Z80

Pierre Giraud
Alain Pineud

La pratique du
TRS-80
Volume III - Fonctionnement du matériel
Schématique - composants

Pierre Giraud
Alain Pineud



Editions du P.S.I.
41-51, rue Jacquard
BP 86 - 77400 Lagny-s/Marne
Téléphone (6) 007.59.31

au Canada : SCE Inc.
3449 rue Saint-Denis
Montréal Québec H2X3L1
Tél : (514) 843.76.63

OP 4

Envoyer ce bon
accompagné
de votre règlement à
EDITIONS DU P.S.I.
41-51, rue Jacquard
BP 86 77400 Lagny-s/Marne
Tél. (6) 007.59.31.

BON DE COMMANDE

DESIGNATION	NOMBRE	PRIX
	TOTAL	

Les prix sont : taxes,
emballage et port compris.
(par avion : ajouter 5 FF par livre)

NOM _____ PRENOM _____
rue _____ N° _____
Code post. [][][][][] Ville _____

Votre bibliothèque informatique

Les ouvrages des éditions du P.S.I. sont répartis en quatre séries de difficulté croissante : - **Série verte** : initiation - **Série bleue** : perfectionnement - **Série rouge** : approfondissement - **Série noire** : maîtrise de la technique.

La découverte de l'Applesoft

par Dominique Schraen et Frédéric Lévy

Cet ouvrage d'initiation s'adresse aussi bien aux futurs utilisateurs de l'Apple voulant apprendre la programmation en Basic Applesoft, qu'à l'Appleophile chevronné sollicité par ses proches curieux de "voir un peu comment ça marche". D'approche progressive, il est illustré de nombreux exemples et exercices.

Série verte - 128 pages - 65 FF

La pratique de l'Apple II - Volume 1

par Nicole Bréaud-Pouliquen

Cet ouvrage présente les spécificités du Basic Applesoft à partir d'une description du matériel et du logiciel du système Apple. Les techniques de programmation, de composition et d'animation de dessins et graphiques colorés y sont expliquées à l'aide d'exemples illustratifs et d'exercices résolus.

Série bleue - 128 pages - 65 FF

La pratique de l'Apple II - Volume 2

par Nicole Bréaud-Pouliquen

Ce second volume de la pratique de l'Apple II est consacré au système d'exploitation disque, à la gestion des fichiers, à l'impression et aux imprimantes, à la carte horloge Appleclock. De nombreux exemples de programmes illustrent les fonctions et les commandes décrites.

Série rouge - 120 pages - 65 FF

La pratique de l'Apple II - Volume 3

par Nicole Bréaud-Pouliquen et Daniel-Jean David

Ce volume est une initiation à la programmation en langage machine 6502, dont le jeu d'instruction est expliqué et utilisé. L'assembleur symbolique et ses logiciels connexes y sont décrits. L'interaction avec le Basic et avec le système y sont étudiés.

Série noire - 176 pages - 75 FF

La pratique du LX 500

par Alain Séméteys et Francis Vasse

Cet ouvrage est une initiation au langage Basic et à sa programmation directement associée au LX 500 et à son logiciel. Sa démarche progressive est ponctuée d'exercices et d'exemples qui sont constitués de programmes s'enrichissant au fur et à mesure de la découverte du langage.

Série verte - 160 pages - 75 FF

La pratique du MZ-80K

par Jean-Pierre Lhoir

Cet ouvrage présente les caractéristiques du Basic SP 5025 de l'ordinateur Sharp MZ-80K. Il comprend plusieurs annexes : messages d'erreurs, code ASC II, réglages de l'écran et du haut-parleur ainsi qu'un répertoire des instructions Basic.

Série bleue - 128 pages - 65 FF

La découverte du PET/CBM

par Daniel-Jean David

Cet ouvrage d'initiation à l'utilisation de la gamme PET/CBM comprend, essentiellement, une introduction progressive au langage Basic. La découverte du langage est conduite en bâtissant des programmes de plus en plus élaborés, au fur et à mesure de l'introduction de notions nouvelles.

Série verte - 136 pages - 65 FF

La pratique du PET/CBM

Volume 1

par Daniel-Jean David

Cet ouvrage, qui suppose une bonne connaissance du Basic et des commandes du PET/CBM, ouvre les portes des applications faisant appel aux fichiers (cassettes, disquettes), à l'impression et au bus IEEE.

Il comporte également de nombreux exemples et exercices avec solution.

Série bleue - 136 pages - 65 FF

La pratique du PET/CBM

Volume 2

par Daniel-Jean David

Ce volume est une initiation à la programmation en langage machine 6502, dont le jeu d'instruction est expliqué et utilisé. L'assembleur symbolique, l'éditeur et le chargeur, y sont décrits. L'interaction avec le Basic et avec le système y sont étudiés.

Série rouge - 176 pages - 75 FF

La découverte du PC-1211

par Jean-Pierre Richard

Au fil des chapitres, cet ouvrage fera découvrir au non initié instructions et commandes, variables et mémoires, fonctions périphériques. Enrichi d'exercices d'applications et d'un index, ce manuel fournit à l'utilisateur tous les éléments de base nécessaires à la programmation en langage Basic du PC-1211 (ou TRS-pocket).

Série verte - 152 pages - 75 FF

La pratique du TRS-80 - Volume 1

par Pierre Giraud et Alain Pinaud

Cet ouvrage s'adresse aux amateurs de TRS-80 Modèles I et III déjà initiés. Il permettra aux utilisateurs de TRS-80 de tirer le meilleur parti de leur ordinateur et, en particulier, du langage BASIC niveau II. Il est complété par des exemples de programmes et des annexes pratiques.

Série bleue - 128 pages - 65 FF

La pratique du TRS-80 - Volume 2

par Pierre Giraud et Alain Pinaud

Ce volume s'adresse à ceux qui s'intéressent à la programmation en langage assembleur du Z-80. L'auteur aborde l'aspect mise au point de programmes écrits en langage assembleur et cite quelques outils. Il présente enfin des exemples de sous-programmes et décrit les entrées-sorties du TRS-80.

Série rouge - 220 pages - 85 FF

La pratique du TRS-80 - Volume 3

par Pierre Giraud et Alain Pinaud

Après le logiciel, le matériel, ce volume vous guide dans l'exploration systématique (55 schémas) du TRS-80 Modèle I. Pour mordus sans complexe en électronique dont une des armes familières est le fer à souder. Ce livre est, par ailleurs, une documentation originale et complète sur la structure d'un ordinateur.

Série noire - 128 pages - 75 FF

La découverte de la TI-57

par Xavier de la Tullaye

S'adressant aux débutants, cet ouvrage les conduira, dans un langage clair, de l'élémentaire 2 + 2 à des programmes perfectionnés. Après une étude fonctionnelle de la calculette, la programmation est expliquée progressivement, de la conception à la réalisation en s'appuyant sur de nombreux exemples.

Série verte - 144 pages - 65 FF

EDITIONS PHIPPS

The ZX-80 Pocket book

(in english)

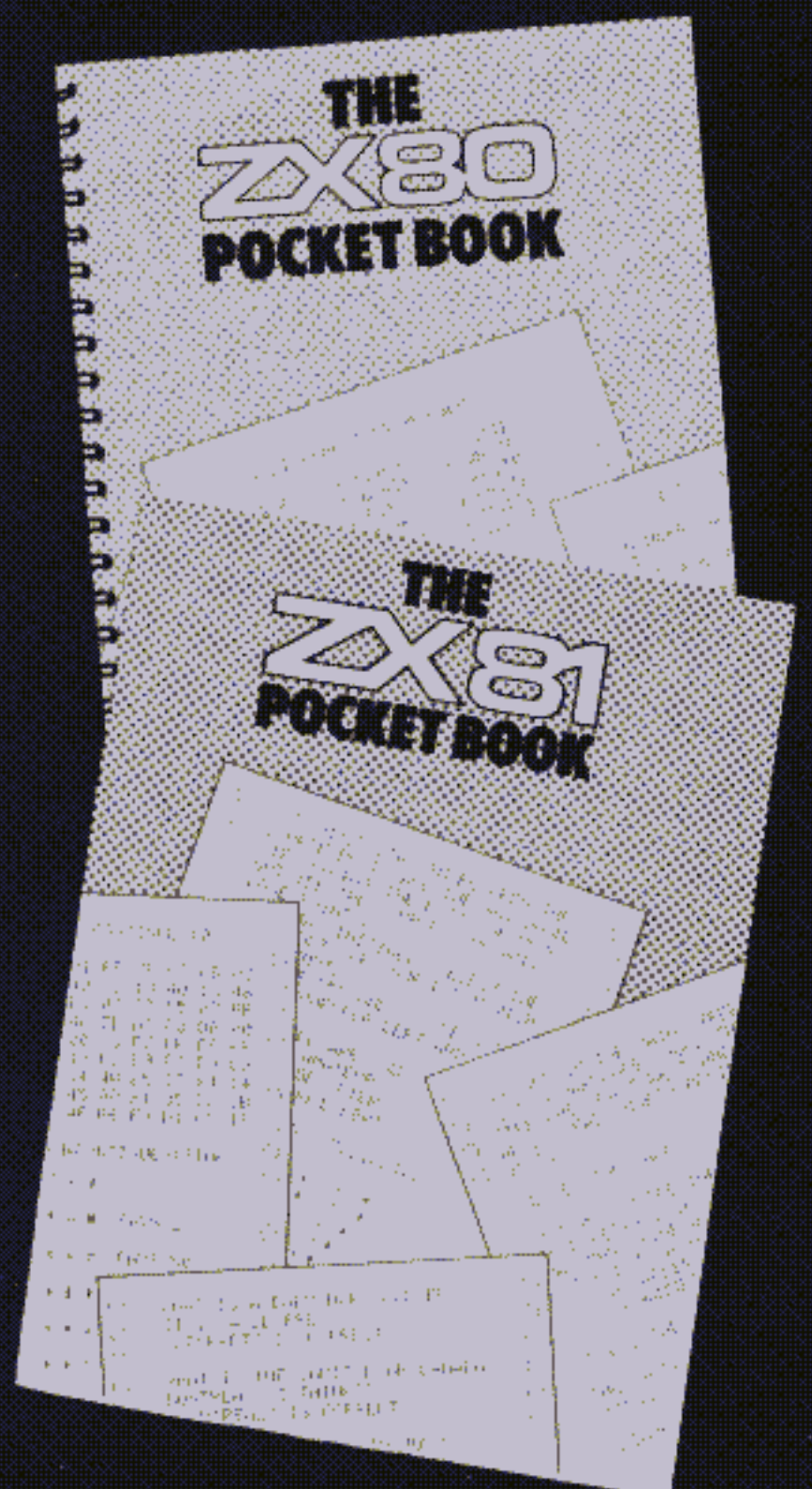
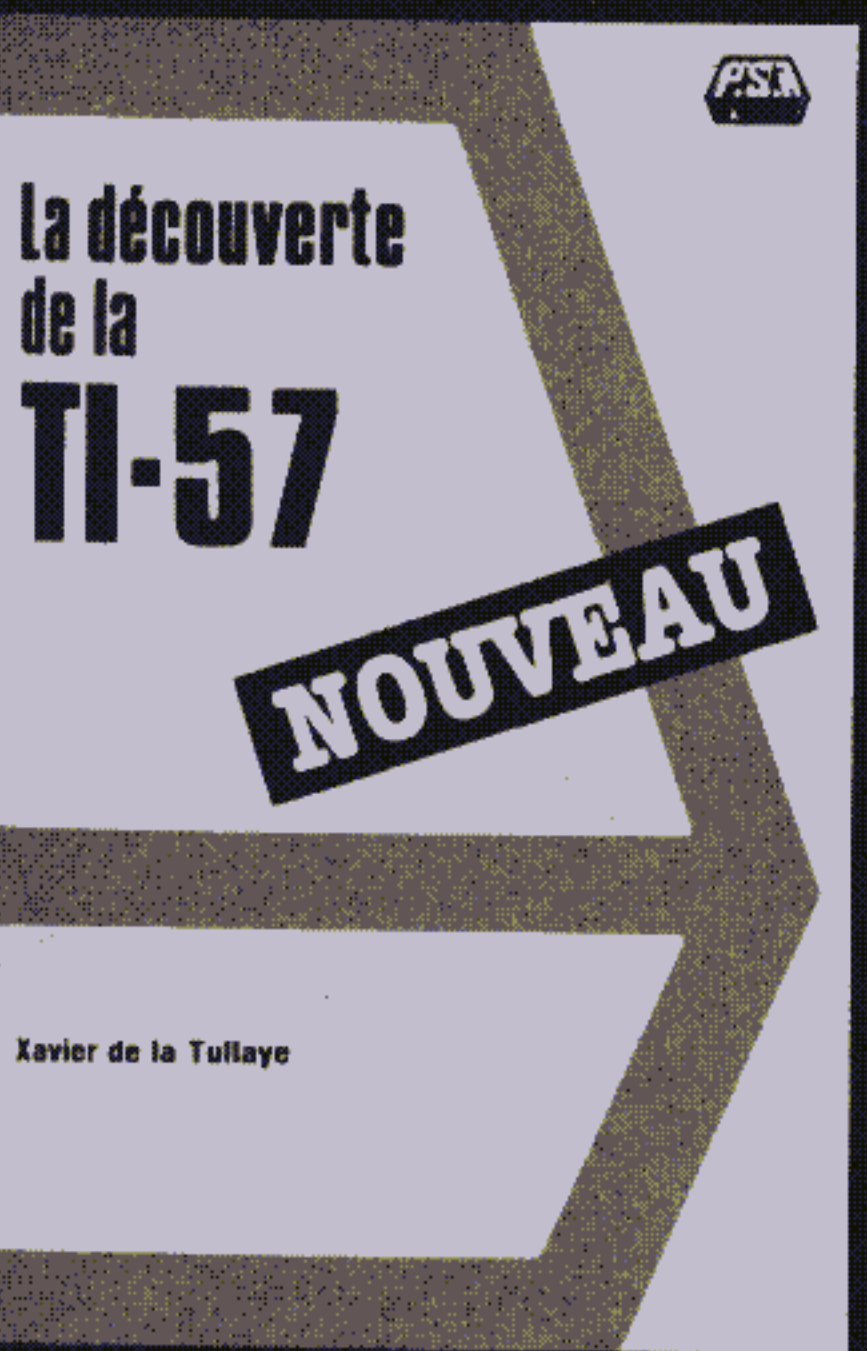
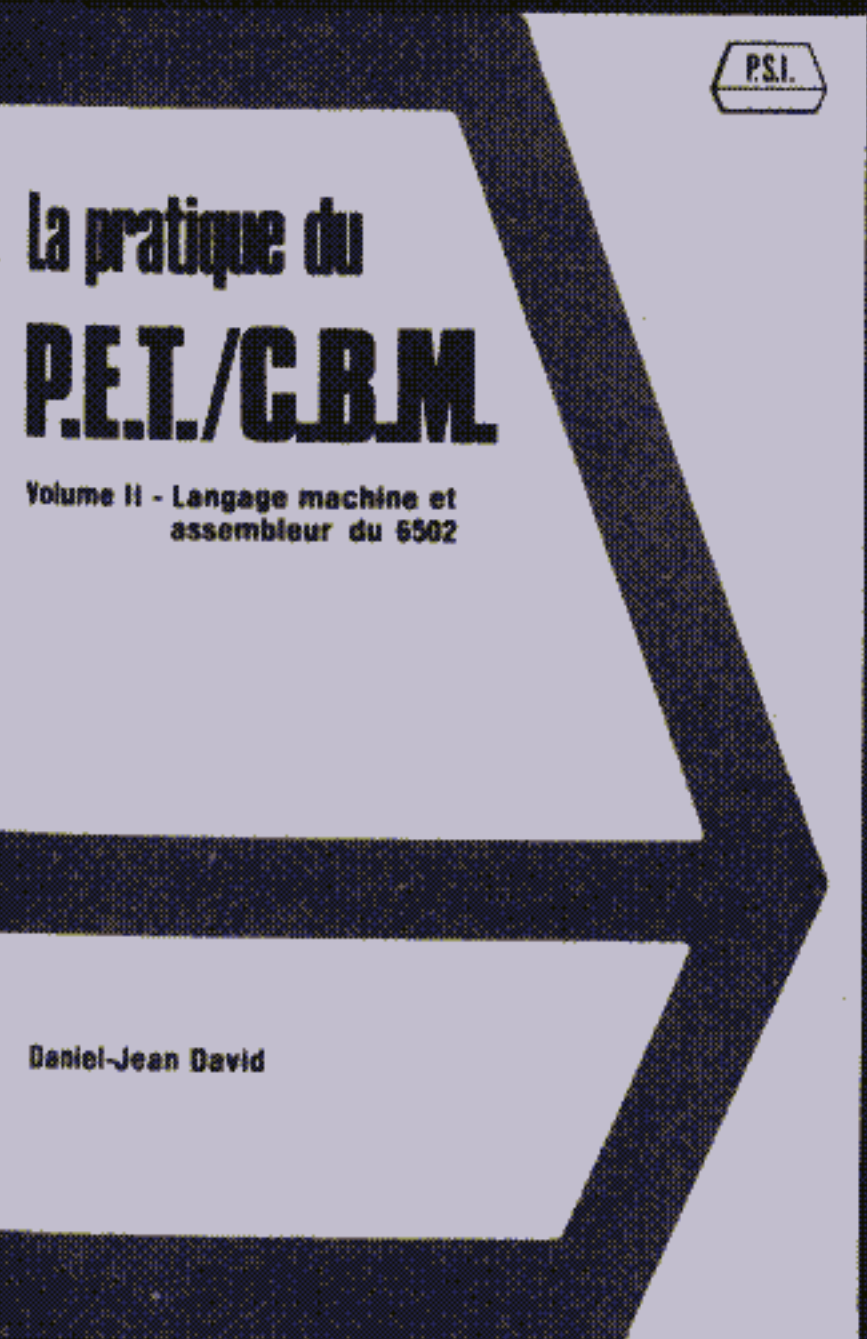
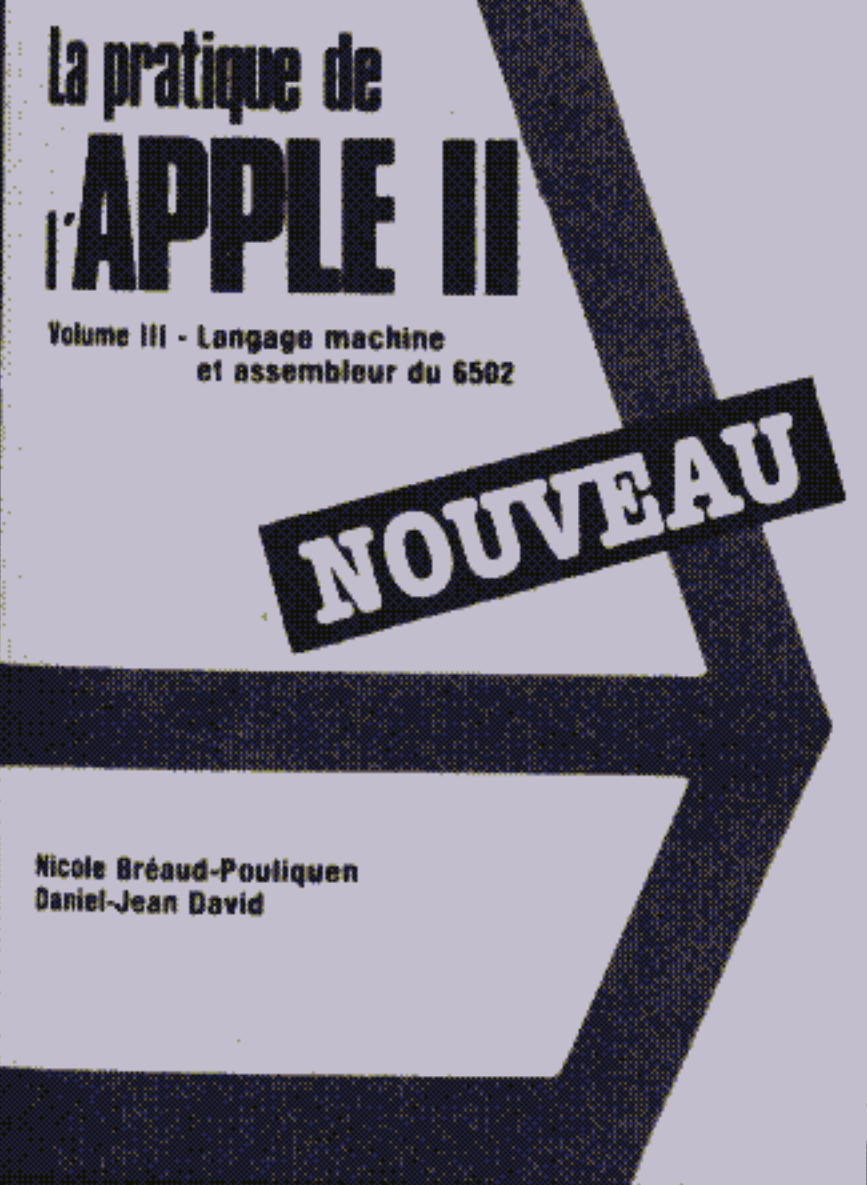
128 pages - 65 FF

The ZX-81 Pocket book

(in english)

Le Sinclair ZX-81 est désormais en vente en France. Les Editions du PSI commercialisent ce livre anglais qui constitue, au-delà de la documentation du fournisseur, une mine de conseils et de trucs pour mieux utiliser votre ZX-81.

136 pages - 65 FF



Evitons le torticolis

La surface des tables est généralement horizontale. Mais les micropoches, quand ils sont à plat sur une table, ne présentent pas souvent leur affichage sous le meilleur angle... Un petit pupitre incliné offre une solution très confortable.



■ Les fabricants d'ordinateurs dépensent des trésors d'ingéniosité pour mettre sur le marché des matériels de plus en plus étonnants. Mais il y a certains petits détails pratiques auxquels ils ne songent pas toujours. Or l'efficacité d'une machine est également fonction de son agrément. Les ergonomistes savent que tout ce qui facilite l'exécution d'une tâche diminue d'autant les risques d'erreurs.

Le petit bricolage auquel nous vous invitons vous permettra d'utiliser un PC-1211/TRS 80 P posé sur votre bureau dans une position optimale, c'est-à-dire oblique : vous pourrez tout à votre aise y pianoter d'une seule main et y lire les résultats sans vous pencher sans cesse sur les cristaux liquides. Il vous obéira en toute simplicité au doigt et à l'œil.

En effet, lorsque l'on regarde bien en face l'afficheur du micropoche, sa lisibilité est impeccable, c'est donc dans cette position qu'il convient de l'utiliser. Le support dont nous vous donnons le plan vous le présentera... comme sur un plateau.

La méthode utilisée permet de

construire un grand nombre d'objets de petites et de moyennes dimensions et elle a l'avantage de la simplicité. En ce qui concerne le support pour Sharp PC-1211, il est composé de 5 pièces dont l'assem-

blage et les cotes sont donnés figure 1 et 2.

On commencera par tracer soigneusement le contour des pièces sur le côté mat du polystyrène en essayant bien entendu d'avoir le moins possible de chutes. Afin de conserver pour l'extérieur du volume la face brillante du plastique, il faudra inverser l'un des deux dessins de la pièce C. Sur la pièce A, on tracera l'emplacement du collage de D et sur C₁ et C₂ les deux autres emplacements (fig. 2). Après quoi on marquera au cutter (lame courbe) les sillons de découpe, puis on passera à la découpe proprement dite avant de poncer soigneusement chacune des tranches. On veillera à ce que les tranches soient perpendiculaires aux faces des pièces. Attention toutefois : en ponçant trop, on risque d'arrondir les tranches.

Le collage doit s'effectuer dans l'ordre suivant : D sur le verso A, puis B, et enfin les côtés C₁ et C₂. On aura tout intérêt à s'aider du plan de travail pour obtenir un parfait affleurement des tranches (fig. 6).

A chaque collage, il faut appuyer

Ce dont vous aurez besoin

Pour confectionner le pupitre, il vous faudra vous munir du matériel suivant :

- une feuille de polystyrène-choc en 2 mm d'épaisseur,
- un tube de colle Uhu Plastic ou similaire pour coller le polystyrène,
- un peu de trichloréthylène (un petit flacon),
- un pinceau,
- un cutter équipé d'une lame droite et d'une lame courbe,
- un triple décimètre, une équerre, un compas, un crayon et une gomme,
- une règle métallique (genre règle de cartonnier) avec, si possible, une semelle antidérapante,
- enfin, et c'est important, un courant d'air soigneusement organisé (fenêtre ouverte, etc.) : les vapeurs de trichloréthylène sont nocives, elles représentent un danger qu'il ne faut pas sous-estimer surtout si l'on renverse le flacon.

fortement pendant une trentaine de secondes. S'il le faut, pour colmater en quelque sorte une jointure imparfaite, on instillera un trait de trichloréthylène avec un pinceau fin, et l'on pressera à nouveau en vérifiant que la lumière ne passe plus entre les pièces assemblées.

Une fois le séchage terminé, il ne restera plus qu'à abattre légèrement au cutter (lame droite) les angles des tranches apparentes.

Si la réalisation du support est particulièrement simple, elle doit être effectuée avec un soin méticuleux. La solidité de l'assemblage en dépend. Il n'est pas inutile de rappeler brièvement les quelques précautions qui assureront le succès de la construction.

• **le tracé** : il ne diffère en rien des autres techniques d'assemblage. Il

faut qu'il soit précis et, par exemple, que les angles droits mesurent bien 90°... On utilisera donc une équerre et un crayon bien aiguisé.

• **le marquage au cutter** (fig. 3) : il s'agit de créer une ligne de rupture et non pas de découper. On utilisera la lame courbe à la façon d'un soc de charrue. Après un premier passage plus léger que les suivants, on recommencera trois ou quatre fois de manière à creuser le sillon.

• **la découpe** : la figure 4 montre comment procéder lorsque le tracé est assez loin du bord. Les deux index placés sous la ligne de rupture servent de point d'appui. Si le tracé est situé très près du bord, on utilisera une paire de pinces universelles. Mais si l'on doit employer la pièce ainsi dégagée, on revêtira les mors de la pince avec un tissu épais ou de l'ouate afin de ne pas marquer le plastique. Le bord d'une table peut également être utilisé comme point d'appui sous la ligne de rupture.

• **le ponçage des tranches** (fig. 5) : la découpe est extrêmement

Les différentes pièces qui composent le pupitre et leurs dimensions exactes.

Les cotes sont données en mm.
Les pièces désignées par la même lettre ont les mêmes dimensions.
Polystyrène choc de 2 mm d'épaisseur.

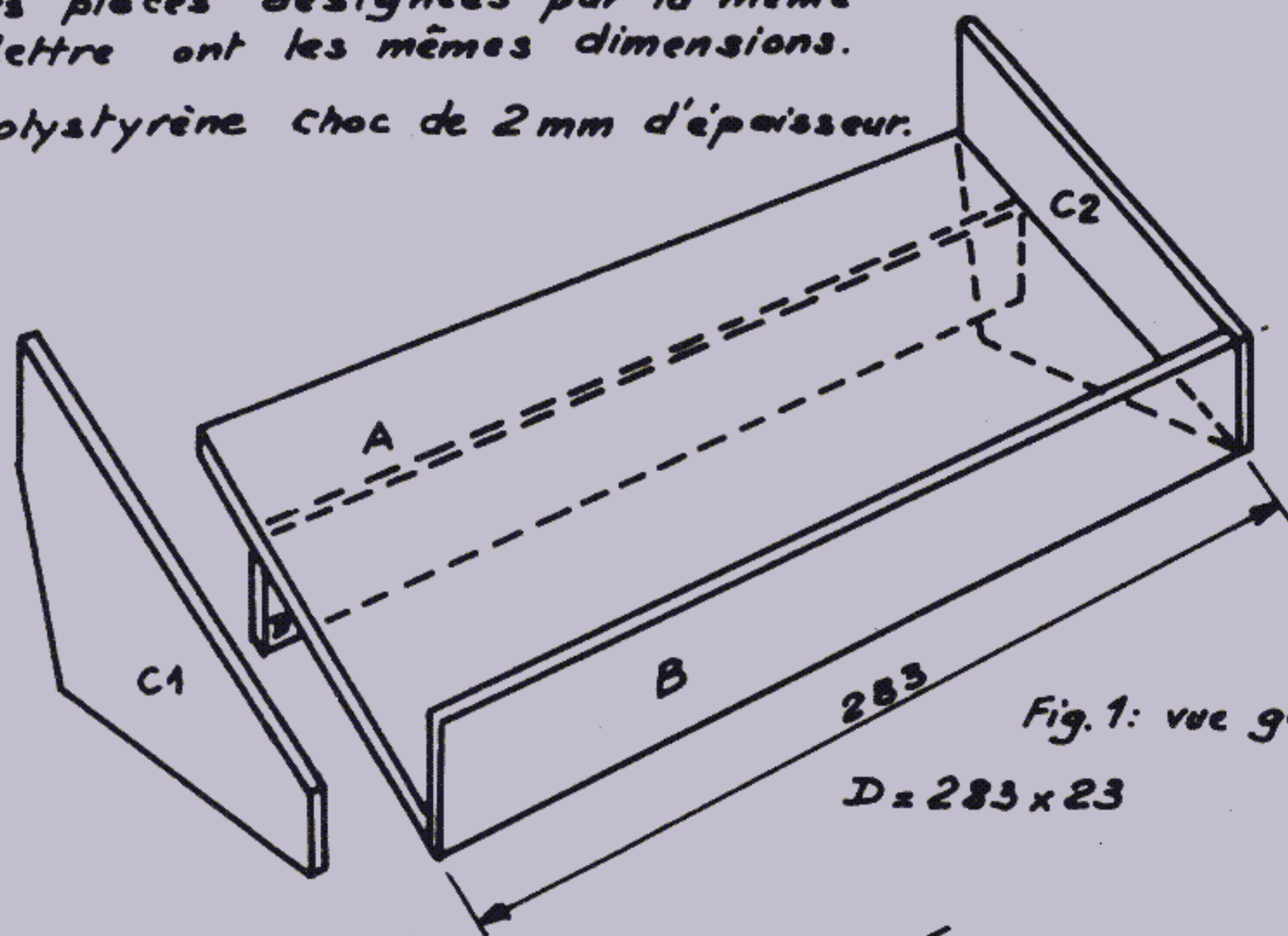


Fig. 1: vue générale.

D = 283 x 23

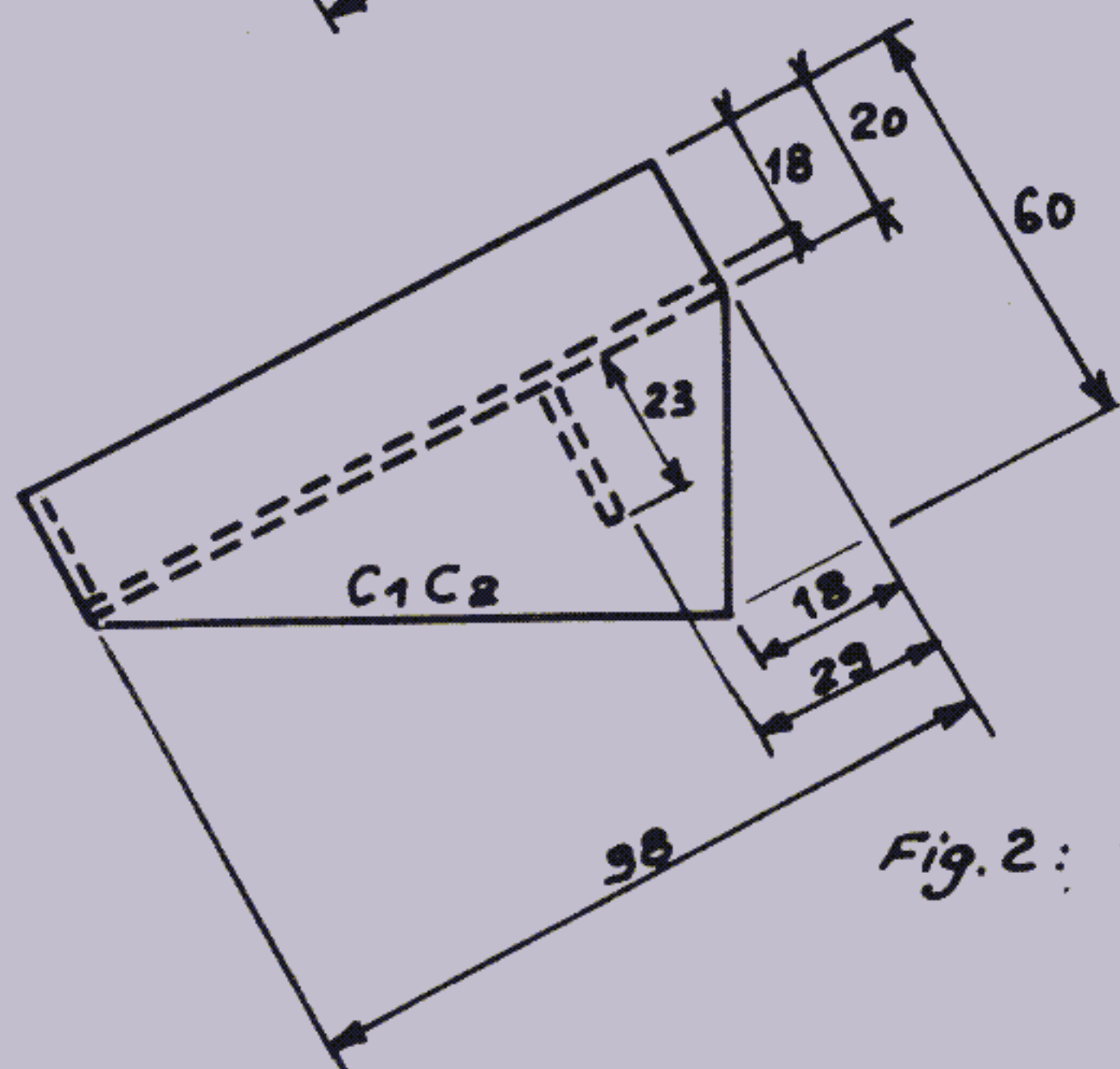
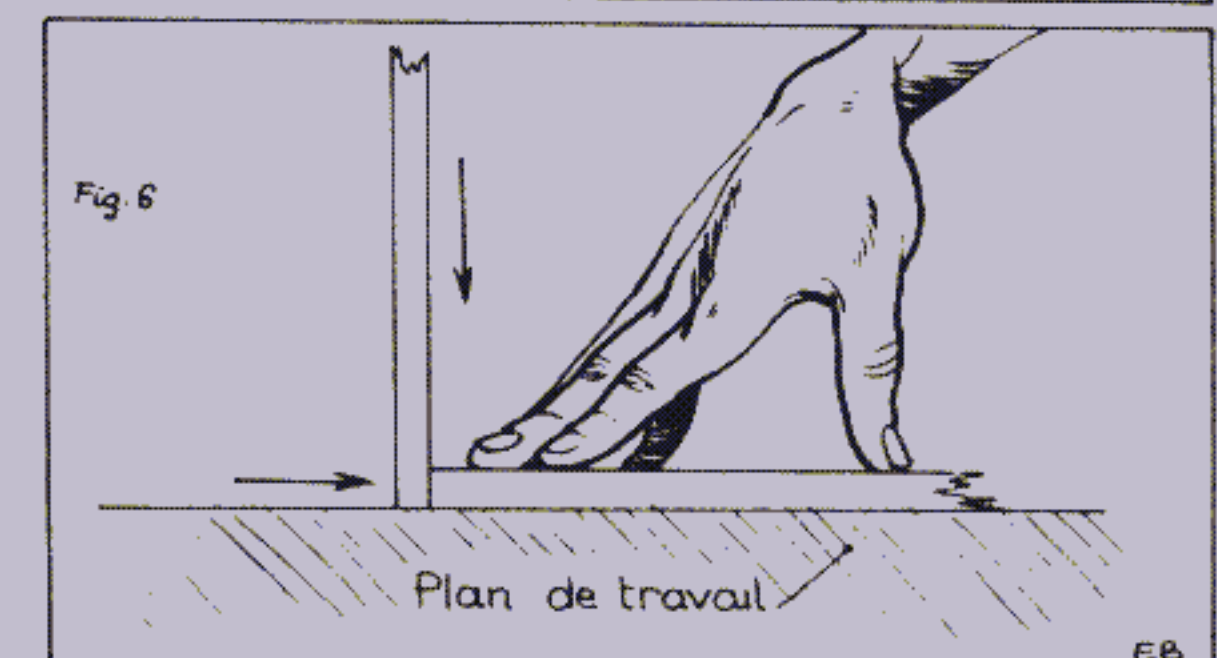
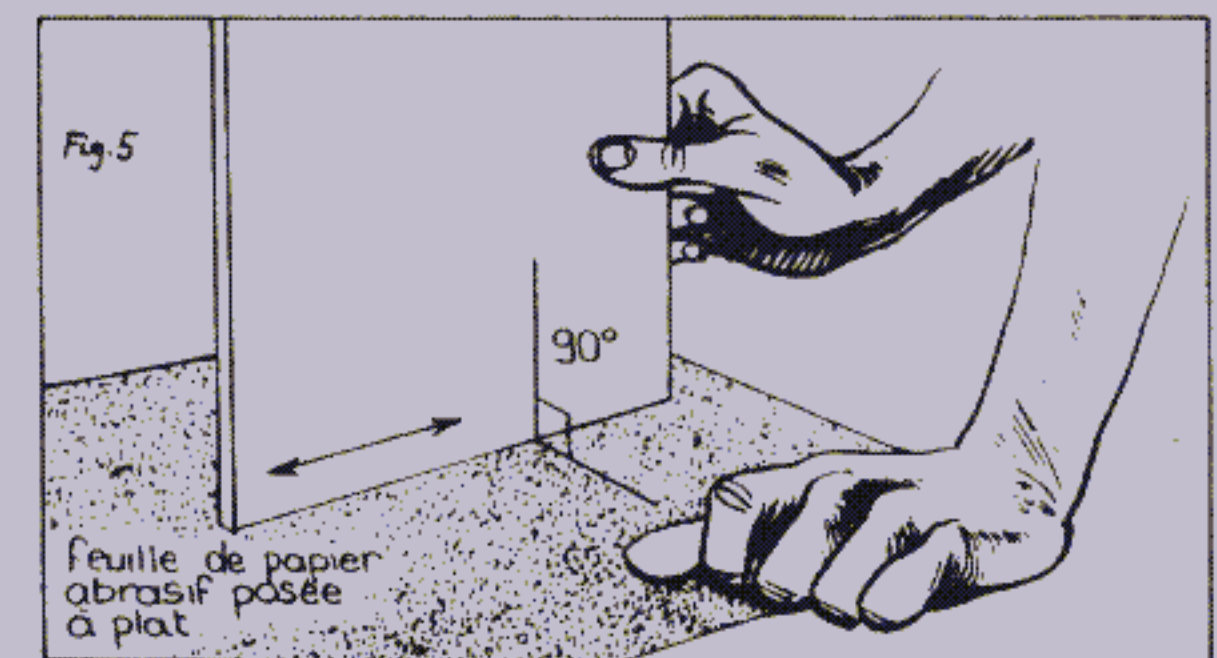
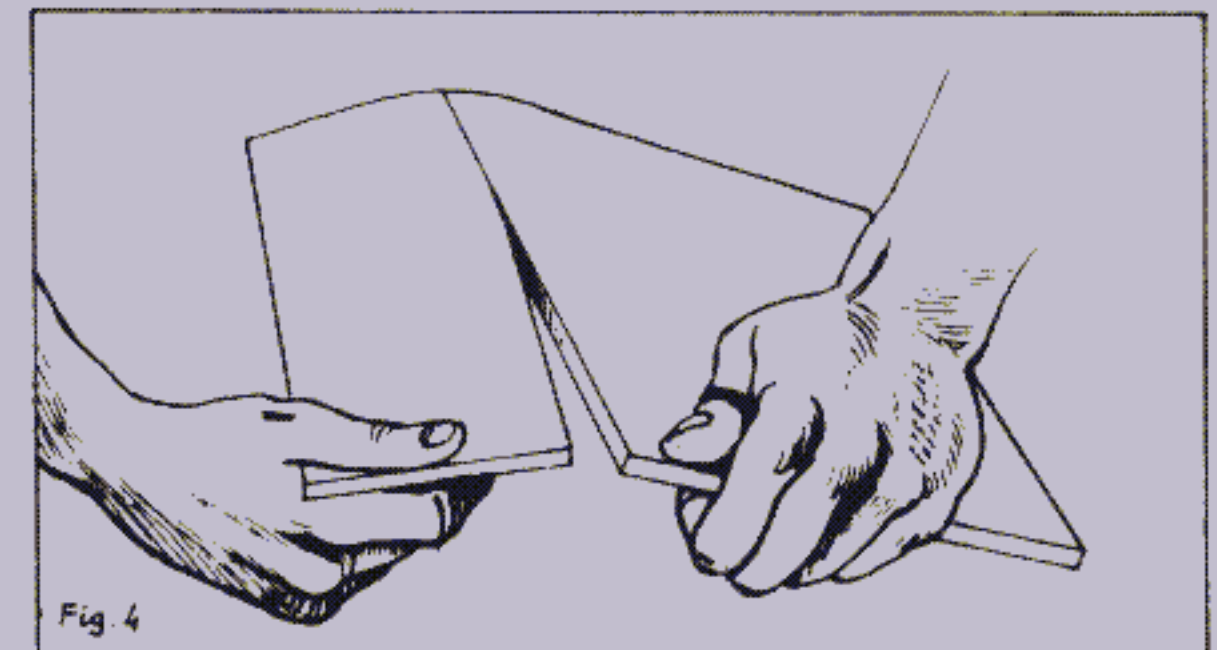
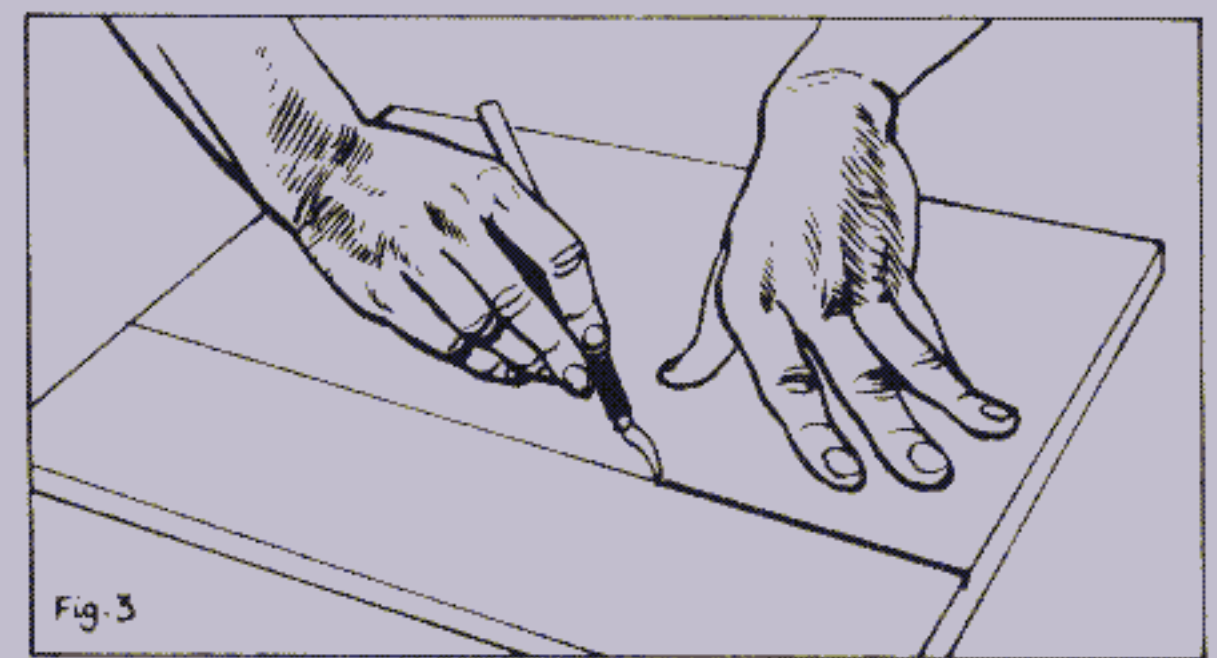


Fig. 2: vue en bout.

Ci-dessous, les principaux gestes que vous aurez à exécuter.

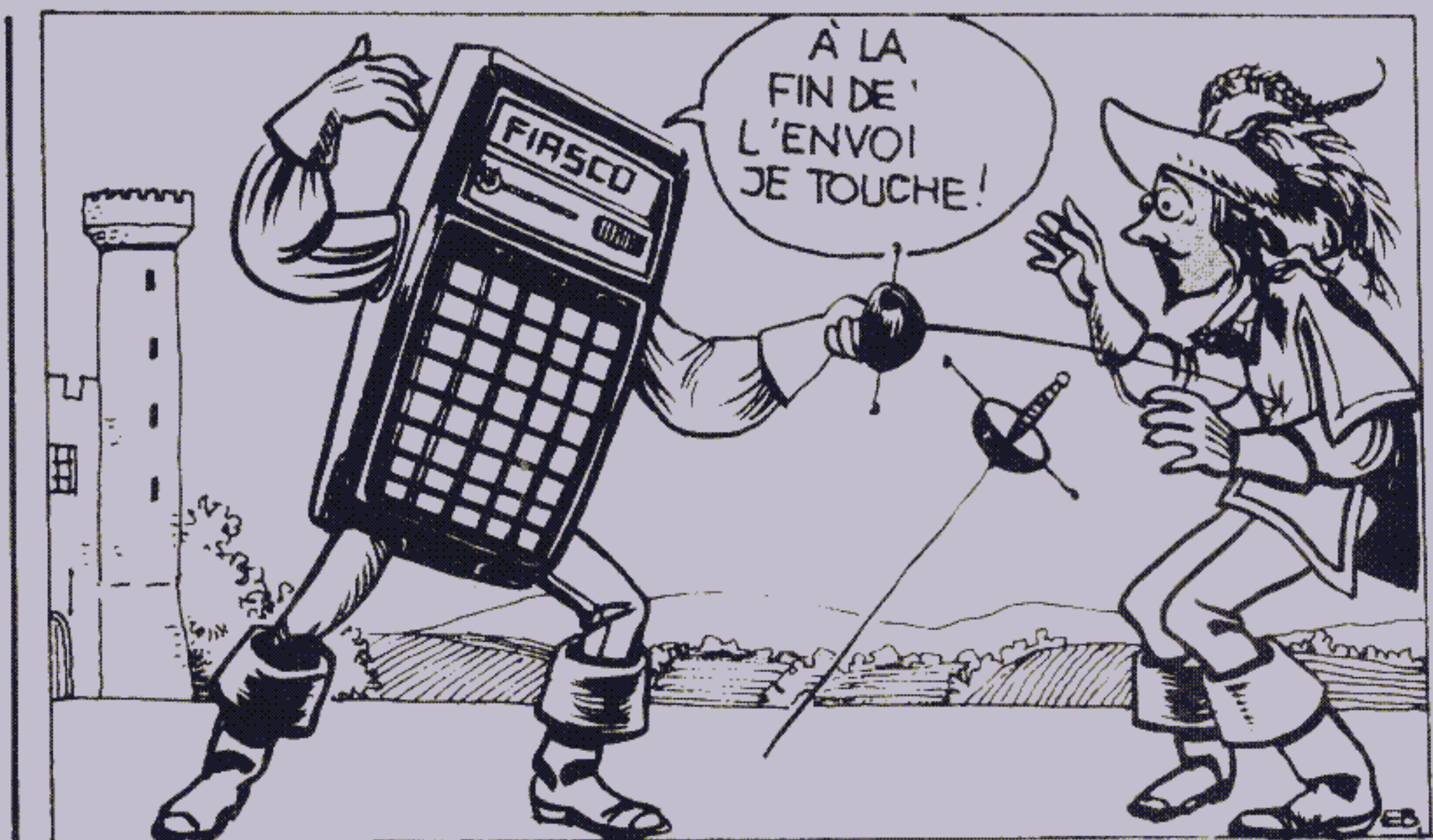


rapide, mais elle laisse un bourrelet et une tranche qui n'est pas vraiment plane. On commence donc par gratter le bourrelet avec la lame droite, puis on ponce la tranche sur une feuille de papier abrasif posée sur la table, sans insister : l'important est de veiller à la perpendicularité de la plaque.

• **le collage** : la colle spécifique ou le trichloréthylène appliqué au pinceau assurent une véritable soudure autogène du matériau. Le séchage définitif demande une à deux heures, mais une pression d'une deminute suffit avant de passer à l'étape suivante de l'assemblage.

Les dimensions indiquées valent pour le PC-1211 équipé de son imprimante CE-122, mais on peut les conserver pour le micropoche seul : il restera un petit espace libre où viendra s'insérer un bloc de papier dont l'utilité, en l'absence d'imprimante, n'est pas à démontrer.

□ Pierre Courbier



Comment faire un fiasco avec une TI 57

Voici un programme de quatorze pas au départ, onze à l'arrivée, qui vous permettra d'obtenir à l'affichage de votre micropoche un véritable fiasco !

■ En reprenant les indications fournies dans l'Op n° 2, je suis parvenu à construire un petit programme qui permet de faire afficher à la TI 57 tous les mots de moins de neuf lettres qui sont composés des lettres a, b, c, d, e et f. A ces lettres s'ajoutent d'ailleurs le chiffre 0 pour la lettre o, le chiffre 5 pour la lettre s et le chiffre 1 pour la lettre i ou, éventuellement le l minuscule. Dans certains mots enfin, le chiffre 8 peut

Programme n° 1

```
LRN
00 2nd Exc SST
01 2nd Lbl 0
02 0
03 R/S
04 7
05 1
06 R/S
07 2
08 5
09 R/S
10 4
11 0
12 R/S
13 RST
```

LRN

très bien faire office de B majuscule et le 6 peut faire un g acceptable (dans *aigle* par exemple). Ce qui nous fait en tout une bonne dizaine de lettres...

On commence par entrer le programme n° 1, puis on procède de la sorte :

- RST suivi de R/S et l'afficheur indique 0,
- LRN nous conduit au pas 04 dont le code est 07 0 ; on introduit alors au clavier INV 2nd 7 (un BST nous apprendrait que le pas 04 contient maintenant le code 07, c'est-à-dire celui de la lettre F, mais pour la bonne marche de la suite, mieux vaut ne pas aller y regarder) ; on quitte donc le mode de programmation (LRN et l'on presse 4 fois sur R/S jusqu'à réobtenir 0 à l'affichage ;
- LRN nous replace une nouvelle fois au pas 04 dont le code est redevenu 07 0 ; on introduit encore INV 2nd 7, on quitte le mode de programmation grâce à LRN et l'on presse 5 fois sur R/S jusqu'à obtenir l'affichage de FI : ce sont les deux premières lettres du mot qui nous intéresse...
- cette fois-ci, LRN provoque l'affichage de 07 02 0 ; on tape alors INV 2nd 2, puis LRN et une série de R/S (5 au total), jusqu'à l'apparition de 51A5 ;
- une autre pression sur LRN et nous découvrons le pas 10 : 10 04 0 ce à quoi l'on répond par INV 2nd 4, LRN, et une nouvelle série de R/S

Programme n° 2

```
00 38 (2nd Exc, SST)
01 86 0 2nd Lbl 0
02 00 0
03 15 équivalent de F
04 01 I
05 10 équivalent de A
06 05 S
07 12 équivalent de C
08 00 O
09 81 R/S
10 71 RST (facultatif)
```

(5 encore) jusqu'à l'affichage de 5205C0 ;

- on se replace alors au début du programme grâce à RST et LRN, et l'on avance avec trois SST jusqu'au pas 03 que l'on supprime au moyen de 2nd Del ; on se rend ensuite au pas 5 (2 pressions sur SST) que l'on supprime aussi ; même procédure pour détruire le pas 07, et un dernier appui sur la touche LRN avant d'exécuter (RST et R/S) le nouveau programme qui se trouve dans la TI 57 (pgr. n° 2). La réponse ne se fait pas attendre, et on ne vous l'envoie pas dire : "FIASCO".

Ce programme n'utilise en fait que dix pas : le dernier — RST — permet seulement de reprendre au début à chaque nouvelle pression sur R/S. Je suggère de l'incorporer à d'autres programmes (je pense aux programmes de jeu) : il permettrait d'indiquer aux joueurs, et en toutes lettres, qu'ils ont perdu (ce sont des choses qui arrivent...).

□ Claude Lacour

Concours de saut sur TI 58/59 étiquettes ou adressage numérique ?

En programmation, les chemins les plus courts sont toujours les meilleurs.

Ce sont donc eux qu'il faut emprunter pour faire des sauts à l'intérieur d'un programme.

■ Les possesseurs de TI 58/59 connaissent bien les deux possibilités de branchement à une adresse donnée : on peut renvoyer à une étiquette (GTO INV par exemple) ou à une adresse numérique (GTO 150).

La difficulté, quand on a deux options comme celles-ci, ce n'est pas tellement d'en choisir une, mais plutôt de les mélanger dans la meilleure proportion et de le faire à bon escient. Il n'est donc pas question de chercher à savoir si l'un des deux adressages est meilleur que l'autre ; l'important est de connaître leurs avantages et leurs inconvénients respectifs pour en tirer parti dans la construction des programmes.

Voyons d'abord ce qui se passe lors d'un saut effectué à la suite d'un test ($x=t$; $x \geq t$), d'une boucle (Dsz), ou d'un branchement inconditionnel (GTO). Lorsque l'adresse est une étiquette, le pointeur la recherche à partir de la ligne 000. Cette recherche n'est pas instantanée : il faut compter environ une

seconde pour une étiquette placée à la ligne 450.

Si l'adresse est numérique au contraire, le branchement sera presque immédiat, d'où un gain de temps non négligeable.

L'adressage numérique a donc un net avantage en ce qui concerne la vitesse d'exécution, ce qui laisse penser, a priori, qu'un programme *dans sa version définitive* devra utiliser ce type d'adressage.

Je viens d'exposer l'argument le plus important en faveur de l'adressage numérique. L'étiquetage a, bien entendu, lui aussi un point fort qui le rend très précieux au moment de la construction d'un programme. Pour renvoyer à ce numéro de ligne, en effet, il faut le connaître, et ce n'est pas toujours possible quand cette ligne se situe en aval du branchement. Si nous décidons, à la ligne 100, d'un branchement vers un bloc qui commence aux alentours de la ligne 150, nous devons attendre d'avoir écrit cette ligne pour en connaître exactement le numéro.

L'étiquetage nous affranchit de cette gymnastique en permettant un branchement à une étiquette sans avoir à nous soucier de sa position dans le programme. De plus, les modifications nécessitant l'insertion ou la destruction de lignes changent la numérotation de toutes celles qui sont en aval de la modification. On entrevoit aisément le travail à accomplir pour changer en conséquence toutes les adresses de saut si l'on utilise l'adressage numérique.

Il est donc souhaitable, sinon

indispensable, d'employer les étiquettes lors de la construction et la mise au point d'un programme.

Une fois le programme terminé, et dûment étiqueté, il semblerait logique de remplacer les étiquettes par des adresses numériques. Dans la pratique, je vous propose un compromis : plutôt que de conserver un seul programme, unique et définitif, gardez d'abord une version du programme étiqueté (liste ou carte magnétique), et une version utilisant à votre gré et de façon mixte les deux types d'adressage.

- La version étiquetée se révèle utile dans plusieurs cas : le programme peut avoir à subir des modifications, améliorations ou ajouts ultérieurs. On comprend que ces transformations réclameront beaucoup de travail sur un programme "rigidifié" par l'adressage numérique. La compréhension du programme par un tiers est grandement facilitée par la présence de blocs bien apparents grâce à la déclaration "Lbl".

- La version définitive, c'est-à-dire celle qui sera utilisée en pratique, ne sera pas nécessairement "numérique" du début jusqu'à la fin. Un programme dont l'exécution demande trois secondes en version étiquetée ne gagnera pas beaucoup à être transformé.

Plus précisément, on connaît les "dévoreurs de temps" dans un programme : ce sont les boucles ; on aura donc souvent intérêt à les "numériser".

Un dernier point impose parfois l'un ou l'autre des deux adressages :

Premier temps de la manœuvre : on note + 1 tous les branchements vers des étiquettes et - 2 toutes les étiquettes.

```

000 76 LBL
001 11 A
002 05 5
003 66 PAU
004 42 STD
005 01 01
006 61 GTO
007 22 INV +1
008 76 LBL
-2 → 009 24 CE
010 01 1
011 00 0
012 42 STD
013 01 01
014 76 LBL
-2 → 015 22 INV
016 43 RCL
017 01 01
018 66 PAU
019 97 DSZ
020 01 01
021 22 INV +1
022 61 GTO
023 24 CE +1

001 11 A
010 009 24 CE
016 015 22 INV

```

c'est la limitation en lignes de programme. Phénomène peu connu sur TI 59, il se rencontre souvent sur TI 58, et il s'en faut parfois d'un rien pour qu'un programme rentre ou ne rentre pas dans la mémoire.

Faisons nos comptes :

- la déclaration d'étiquette " coûte " deux lignes (Lbl INV),
- le branchement à une étiquette coûte deux lignes (GTO INV),
- le branchement à une adresse numérique coûte trois lignes (GTO 01 50 ; pour GTO 150).

On gagne donc à ce qu'une adresse vers laquelle il n'existe qu'un seul branchement soit numérique (pas de déclaration, saut : trois lignes, coût total : trois lignes).

En revanche, si l'on a un branchement fréquent vers une même adresse, on aura intérêt à utiliser une étiquette. En effet, si la déclaration coûte deux lignes, chaque branchement économise une ligne par rapport à l'adressage numérique. Il

suffit alors de placer les étiquettes concernées en début de programme pour limiter le temps de recherche.

Pour conclure, voici une méthode rapide et rationnelle pour " numériser " un programme étiqueté. Le programme utilisé comme exemple ne trouve pas une application évidente (je me demande moi-même à quoi il peut servir), mais le but en est essentiellement pédagogique.

Procédé :

- écrire la liste du programme avec les numéros de lignes (c'est ici, entre autres, que l'imprimante peut rendre des services) ;
- repérer quelles sont les déclarations d'étiquettes qui devront être " numérisées " et placer une flèche à gauche devant la ligne qui suit la déclaration (début du bloc proprement dit) ; les utilisateurs de PC 100 exécuteront 2nd Op 08 qui donne la liste des étiquettes et ils ajouteront 1 à chaque numéro de ligne ;
- placer une flèche à droite au

```

000 76 LBL
001 11 A
002 05 5
003 66 PAU
004 42 STD
005 01 01
006 61 GTO
007 22 INV +1
008 76 LBL
-2 → 009 24 CE Total:
010 01 1 -1
011 00 0
012 42 STD
013 01 01
014 76 LBL
-2 → 015 22 INV Total:
016 43 RCL -3
017 01 01
018 66 PAU
019 97 DSZ
020 01 01 +1
021 22 INV +1
022 61 GTO +1
023 24 CE +1

001 11 A
009 010 24 CE
013 016 22 INV

```

Deuxième étape : on fait le décompte des pas en plus et en moins.

niveau de chaque branchement devant être " numérisé " ;

- au-dessus de chaque flèche correspondant à une déclaration, écrire " -2 " ; au-dessus de chaque flèche correspondant à un branchement, écrire " +1 " (voir liste 1).

Dès lors, le travail est simple. En partant du début du programme, faire la somme des +1 et des -2. Arrivé à une déclaration d'étiquette, faire le total et l'ajouter au numéro correspondant à cette étiquette. Continuer ainsi jusqu'à la fin, et l'on a alors toutes les adresses numériques des étiquettes devant disparaître (voir liste 2).

```

000 76 LBL
001 11 A
002 05 5
003 66 PAU
004 42 STD
005 01 01
006 61 GTO
007 00 00
008 13 13
009 01 1
010 00 0
011 42 STD
012 01 01
013 43 RCL
014 01 01
015 66 PAU
016 97 DSZ
017 01 01
018 00 00
019 13 13
020 61 GTO
021 00 00
022 09 09

```

Le programme définitif " figé " avec ses adressages numériques.

La fin du travail consiste à détruire les déclarations d'étiquettes (Delete), et à remplacer chaque saut vers une étiquette par un branchement à l'adresse numérique correspondante, sans oublier d'insérer une ligne à chaque branchement (voir liste 3).

Voilà, les explications paraissent peut-être un peu longues, mais à l'usage cette méthode se révèle facile à comprendre et très pratique.

□ Antoine Jennet

Un pot commun pour toutes les machines

Conversions des unités biologiques

Dans le deuxième numéro de *l'Op*, on a pu lire un article de Constantin Missirliu où figurait un programme très pratique de conversions des unités biologiques pour TI 57. Jusqu'à une date récente, le résultat de certaines analyses était exprimé sous forme de poids par volume. Cependant, on utilise de plus en plus les nouvelles unités S.I. qui expriment les résultats en micromoles par litre. La coexistence de deux systèmes de mesure différents contraint médecins et biologistes à des conversions fastidieuses dont les micropoches peuvent très bien se charger.

constituant à convertir au moyen d'un *menu* où défilent, en rouleau, les différents éléments que le programme peut convertir : urée, créatinine, bilirubine, glucose, etc.

On lance l'exécution du programme en demandant XEQ « CO » (comme conversion) et le menu affiche l'un après l'autre les différents composants disponibles. Pour indiquer son choix, on doit seulement appuyer sur l'une quelconque des touches numériques lorsque le composant attendu est affiché.

Le micropoche indique alors l'indicateur du composant et la valeur de la constante de conversion. Cela permet de vérifier qu'il n'y a pas d'erreur à cette étape du calcul. On introduit alors la quantité à convertir : q, R/S, et l'on indique dans quel sens va s'effectuer la conversion :

- 0 correspond à une conversion g/l en U.I.
- 1 correspond à une conversion U.I. en g/l et l'on presse sur R/S.

Pour une nouvelle quantité du même produit, il suffit d'appuyer sur A. Si l'on veut utiliser le programme pour un autre produit, on repart à zéro en demandant XEQ « CO ».

□ Pascal Leclerc

Sur HP 41 C

■ Les modifications qui ont été apportées au programme initial vont dans le sens d'une plus grande facilité d'utilisation. La mémoire constante et le caractère alphanumérique du micropoche le permettaient, on ne s'en est donc pas privé.

C'est ainsi que le programme permet de traiter 15 des constituants les plus fréquents et qu'il n'est plus nécessaire d'introduire à chaque initialisation les différents facteurs de conversions. La mémoire de l'ordinateur n'étant pas volatile, ils sont conservés avec le programme.

Afin de limiter le plus possible les risques d'erreurs, un test de décision permet de déterminer le type de conversion souhaitée. D'autre part, pour le confort de l'utilisateur, on a fait en sorte qu'il puisse choisir le

```
42+LBL b
CLA ARCL 02 "F "
ARCL X AVIEW RTN
```

```
49+LBL 01
5.95 "URIC" RTN
```

```
53+LBL 02
1.71 "BILI" RTN
```

```
57+LBL 03
2.52 "C.S." RTN
```

```
61+LBL 04
.025 "Ca++" RTN
```

```
65+LBL 05
8.85 "CREA" RTN
```

```
69+LBL 06
.179 "FER" RTN
```

```
73+LBL 07
5.56 "GLU" RTN
```

```
77+LBL 08
.62 "H.b" RTN
```

```
81+LBL 09
.32 "PHO" RTN
```

```
85+LBL 10
1.14 "T.G." RTN
```

```
89+LBL 11
16.6 "UREE" RTN
```

```
93+LBL 12
1.45 "ALBU" RTN
```

```
97+LBL 13
2.94 "FIBRI" RTN
```

```
101+LBL 14
.5 "MAG" RTN
```

```
105+LBL 15
.133 "Pa02/PC02" RTN
```

```
109+LBL 16
TONE 9 SF 00 .END.
```

```
01+LBL "CO"
"MENU" AVIEW PSE CLST
CF 22
```

```
07+LBL 99
RDN ISG X DEG FS?C 00
GTO "CO" XEQ IND X
AVIEW PSE FC?C 22
GTO 99 RDN STO 00
ASTO 02
```

```
21+LBL A
CLA ARCL 02 "F CST="
FIX 2 ARCL 00 PROMPT
STO 01 TONE 0 STOP
X>0? XEQ B RCL 00
ST* 01 RCL 01 GTO b
```

```
37+LBL B
RCL 00 1/X STO 00 RTN
```

Un pot commun pour toutes les machines

Sur PC-1211 TRS 80 pocket

■ Cette version du programme de C. Missirliu dispose elle aussi d'un menu qui simplifie grandement le choix de la séquence mais qui peut être supprimé si on le désire. Ce sont les lignes 10 à 17 qui organisent ce menu dont le principe est très simple puisque l'instruction INPUT du BASIC Sharp fait sauter à la ligne suivante si l'on n'introduit pas de valeur : le reste de la ligne n'est pas exécuté et l'ancienne valeur de la variable, A dans notre exemple, est conservée.

```

10: "V" PAUSE "CH
    OIX : " :
    INPUT " BILI
    RUBINE " : A: B
    =1.71: END
11: INPUT " CHOL
    ESTEROL " : A:
    B=2.58: END
12: INPUT " GLUC
    OSE " : A: B=5.
    56: END
13: INPUT " UREE
    " : A: B=16.6:
    END
14: INPUT " CREA
    TININE " : A: B
    =8.85: END
15: INPUT " HEMO
    GLOBINE " : A:
    B=.62: END
16: INPUT " CALC
    IUM " : A: B=.0
    25: END
17: GOTO "V"
20: "C" AREAD A: A
    =1/(1/A)*B)
    : PRINT "UNIT
    E CLASSIQUE:
    " : A: END
30: "S" AREAD A: A
    =A*B: PRINT "
    UNITE S. I. :
    " : A: END

```

Restent disponibles 1 129 pas de programme pour allonger la liste des constituants que vous désirez convertir. Ce n'est donc pas la place qui vous manquera.

La procédure à suivre est la suivante :

- 1 Introduire le programme et passer en mode DEF ;
- 2 Choix de la séquence : faire SHIFT V (valeur) et le nom des substances apparaît sur l'afficheur en toutes lettres ; s'il ne s'agit pas de la substance recherchée, il suffit de presser sur ENTER pour passer à la substance suivante. Lorsque l'on est rendu à la rubrique recherchée (hémoglobine ou calcium par exemple), on presse sur une touche numérique puis sur ENTER. La machine alors rend la main.
- 3 A ce moment-là, le facteur de conversion a été introduit dans la variable B. On affiche la valeur connue et pour obtenir celle qui est recherchée, on a le choix entre deux options :
 - S'il s'agit de convertir en unité classique, on demande SHIFT C (C comme classique) ;
 - S'il s'agit de convertir en unité S.I., on demande SHIFT S (S comme S.I.).

Un petit exemple d'exécution pour terminer. Si l'on désire connaître ce que représente 1,96 gr de cholestérol en unité S.I., on procédera de la sorte :
 SHIFT V : apparaît « choix », puis « Bilirubine » ;
 ENTER : apparaît « Cholestérol » ;
 1 ENTER : le micropoche rend la main ;
 1.96 : affichage de la valeur connue ;
 SHIFT S : affichage de « 5,0568 » qui est la traduction en unité S.I. de la valeur que l'on connaissait en unité classique (1,96 gr).

□ Luc Bureller

Des erreurs de transcription peuvent toujours se glisser entre le programme original et sa publication. S'agissant de logiciels qui peuvent servir à des fins médicales, la plus grande prudence s'impose. Avant d'utiliser l'un de ces programmes, chacun vérifiera donc scrupuleusement que les conversions obtenues sont bien celles qui conviennent.

NDLR

La TI 58/59 au grand maximum

■ La page 60 du n° 3 de *l'Op* montrait une méthode originale (pour tous ceux qui ne la connaissaient pas) permettant de trouver à l'aide d'un micropoche le maximum d'une fonction sur un intervalle donné (voir encadré).

Lecture de l'article, et bien entendu je cherche la liste me permettant de rassasier ma TI 59... Chou blanc ! J'en conclus aussitôt que ma machine a été placée hors-concours. Il y a bien des comparaisons entre HP et Sharp, mais pas de TI. Réparons incontinent cet oubli,

Fig. 1

$$F(X) = \frac{1 + 2X - X^2}{(1 - X)^2}$$

ETUDE DE MAX (F) = M
SUR (2.3, 2.6)
M = F (T)

THEORIQUE :
 $T = 1 + \sqrt{2}$
 2.414213562
 $M = 4 - 2\sqrt{2}$
 1.171572875

10 BOUCLES
 2.414257813 T
 1.17157285 M
 ERREUR :
 - 4.4250127-05 e (T)
 2.5734-08 e (M)

18 BOUCLES
 2.414214897 T
 1.171572875 M
 ERREUR :
 - 1.334781-06 e (T)
 1.2-11 e (M)

Comment calculer le maximum d'une fonction sur un intervalle donné ? Ce problème peut se révéler assez simple comme l'indique l'exemple ci-dessus.

```

000 76 LBL      044 42 STO      088 54 )
001 11 A       045 05 05      089 55 ÷
002 42 STO     046 43 RCL      090 02 2
003 09 09     047 02 02      091 95 =
004 91 R/S    048 71 SBR      092 91 R/S
005 42 STO     049 01 01      093 61 GTD
006 04 04     050 10 10      094 01 01
007 91 R/S    051 42 STO      095 10 10
008 42 STO     052 06 06      096 43 RCL
009 00 00     053 32 X/T      097 02 02
010 43 RCL    054 43 RCL      098 42 STO
011 00 00     055 05 05      099 04 04
012 85 +      056 77 GE       100 61 GTD
013 43 RCL    057 00 00      101 00 00
014 04 04     058 96 96      102 79 79
015 54 )      059 43 RCL      103 43 RCL
016 55 ÷      060 03 03      104 02 02
017 02 2      061 71 SBR      105 42 STO
018 54 )      062 01 01      106 00 00
019 42 STO    063 10 10      107 61 GTD
020 02 02     064 32 X/T      108 00 00
021 85 +      065 43 RCL      109 79 79
022 43 RCL    066 05 05      110 42 STO
023 04 04     067 22 INV      111 07 07
024 54 )      068 77 GE       112 75 -
025 55 ÷      069 01 01      113 06 6
026 02 2      070 03 03      114 54 )
027 54 )      071 43 RCL      115 65 ×
028 42 STO    072 01 01      116 43 RCL
029 03 03     073 42 STO      117 07 07
030 43 RCL    074 00 00      118 54 )
031 00 00     075 43 RCL      119 85 +
032 85 +      076 03 03      120 07 7
033 43 RCL    077 42 STO      121 54 )
034 02 02     078 04 04      122 55 ÷
035 54 )      079 97 DSZ      123 53 (
036 55 ÷      080 09 09      124 01 1
037 02 2      081 00 00      125 75 -
038 54 )      082 10 10      126 43 RCL
039 42 STO    083 43 RCL      127 07 07
040 01 01     084 00 00      128 54 )
041 71 SBR    085 85 +        129 54 )
042 01 01     086 43 RCL      130 92 RTN
043 10 10     087 04 04      131 00 0

```

me suis-je dis, et voici donc un programme pour Texans dont on peut comparer les performances avec celles de HP et de Sharp : il est plus précis que le premier et plus rapide que le second.

Pour 10 boucles, l'exécution demande environ 50 secondes. Pour 18 boucles, c'est-à-dire avec une bonne précision, il faut attendre un peu moins de 100 secondes. Les

résultats parlent d'eux-mêmes, si je puis dire, à la figure n° 1.

C'est à partir de la ligne 110 qu'il faut placer la séquence de programme correspondant à la fonction à étudier. Dans notre exemple, c'est à cette ligne qu'apparaît la fonction

$$f(x) = \frac{1+2x-x^2}{(1-x)^2}$$

Vous pourrez utiliser les registres

De quoi s'agit-il ?

Connaissant l'expression mathématique d'une fonction, et sachant que celle-ci possède un maximum M sur un intervalle [a, b] donné, on cherche à déterminer une valeur approchée de ce maximum et de T, valeur de l'abscisse pour laquelle on a M = f(T).

On impose au programme, avant d'en lancer l'exécution, un certain nombre de boucles n. Plus n est grand et meilleure est la précision, mais le temps d'exécution augmente en conséquence.

R 07, R 08... etc. si besoin est. Ne pas oublier de terminer cette séquence par un RTN (touches INV SBR au clavier).

Pour obtenir T et M sur [a, b], introduire :

- le nombre de boucles désirées x, et appuyer sur A,
- la borne inférieure a, et appuyer sur R/S.

Après exécution, la calculatrice affiche T. Pour obtenir M, appuyer sur x = t.

□ André Turlure

Pour lire les listes de la PC 100

Les programmes que nous publions pour TI 58, 58C ou 59 sont presque toujours listés sur l'imprimante PC 100. Il arrive parfois que l'on ait du mal à s'y retrouver car les codes qui sont inscrits en clair par la PC 100 ne sont pas tous identiques à ceux qui sont gravés sur le clavier du micropoche.

Voici un petit tableau d'équivalence où figurent les instructions qui prêtent le plus à confusion.

Listé par la PC 100	Code num.	Séquences de touches
LNX	23	2nd Inx
PAU	66	2nd Pause
EQ	67	2nd x = t
GE	77	2nd x ≥ t
STF	86	2nd St flg
IFF	87	2nd If flg
LST	90	2nd List
RTN	92	INV SBR
PD*	64	2nd Prd 2nd Ind
ST*	72	STO 2nd Ind
RC*	73	RCL 2nd Ind
SM*	74	SUM 2nd Ind

D'une façon générale, l'astérisque dans une liste sortie de la PC 100 remplace la séquence 2nd Ind.

□ l'Op

Un pot commun

pour toutes

les machines

000	32	XIT	059	69	OP	118	32	XIT	177	47	CMS
001	25	CLR	060	20	20	119	01	1	178	86	STP
002	77	GE	061	73	RC*	120	02	2	179	01	01
003	02	02	062	00	00	121	03	3	180	42	STD
004	14	14	063	44	SUM	122	04	4	181	11	11
005	01	1	064	14	14	123	05	5	182	01	1
006	00	0	065	43	RCL	124	06	6	183	00	0
007	22	INV	066	00	00	125	07	7	184	42	STD
008	77	GE	067	67	EQ	126	08	8	185	12	12
009	02	02	068	00	00	127	09	9	186	11	A
010	14	14	069	77	77	128	00	0	187	43	RCL
011	25	CLR	070	01	1	129	87	IFF	188	11	11
012	42	STD	071	00	0	130	01	01	189	23	LNK
013	00	00	072	49	PRD	131	01	01	190	33	X*
014	42	STD	073	14	14	132	36	36	191	42	STD
015	14	14	074	61	GTD	133	42	STD	192	11	11
016	32	XIT	075	00	00	134	18	18	193	65	*
017	42	STD	076	59	59	135	32	XIT	194	01	1
018	15	15	077	01	1	136	42	STD	195	09	9
019	42	STD	078	44	SUM	137	13	13	196	09	9
020	16	16	079	17	17	138	42	STD	197	09	9
021	73	RC*	080	43	RCL	139	14	14	198	01	1
022	15	15	081	18	18	140	01	1	199	95	=
023	48	EXC	082	32	XIT	141	00	0	200	22	INV
024	01	01	083	43	RCL	142	42	STD	201	59	INT
025	72	ST*	084	14	14	143	00	00	202	65	*
026	15	15	085	67	EQ	144	01	1	203	01	1
027	01	1	086	02	02	145	00	0	204	00	0
028	22	INV	087	20	20	146	22	INV	205	85	+
029	44	SUM	088	25	CLR	147	49	PRD	206	01	1
030	16	16	089	67	EQ	148	14	14	207	95	=
031	32	XIT	090	01	01	149	43	RCL	208	59	INT
032	43	RCL	091	14	14	150	14	14	209	97	DSZ
033	16	16	092	76	LBL	151	22	INV	210	12	12
034	22	INV	093	15	E	152	59	INT	211	00	00
035	77	GE	094	43	RCL	153	22	INV	212	17	17
036	01	01	095	17	17	154	44	SUM	213	81	RST
037	01	01	096	66	PAU	155	14	14	214	25	CLR
038	01	1	097	43	RCL	156	52	EE	215	35	1/X
039	00	0	098	14	14	157	01	1	216	32	XIT
040	32	XIT	099	91	R/S	158	72	ST*	217	91	R/S
041	43	RCL	100	81	RST	159	00	00	218	61	GTD
042	15	15	101	01	1	160	25	CLR	219	15	E
043	67	EQ	102	00	0	161	97	DSZ	220	32	XIT
044	01	01	103	42	STD	162	00	00	221	43	RCL
045	08	08	104	16	16	163	01	01	222	17	17
046	01	1	105	61	GTD	164	44	44	223	32	XIT
047	44	SUM	106	00	00	165	42	STD	224	66	PAU
048	15	15	107	38	38	166	17	17	225	61	GTD
049	73	RC*	108	01	1	167	43	RCL	226	02	02
050	16	16	109	42	STD	168	13	13	227	23	23
051	63	EX*	110	15	15	169	92	RTH	228	76	LBL
052	15	15	111	61	GTD	170	81	RST	229	14	D
053	72	ST*	112	00	00	171	76	LBL	230	43	RCL
054	16	16	113	49	49	172	13	D	231	13	13
055	87	IFF	114	43	RCL	173	43	RCL	232	61	GTD
056	01	01	115	14	14	174	11	11	233	11	A
057	01	01	116	76	LBL	175	76	LBL	234	00	0
058	87	87	117	11	A	176	12	B	235	00	0

Le Kibur sur TI 58/59

■ La possibilité de kiburer (l'Op n° 3, page 42) n'est pas réservée aux seuls alphanumériques de la HP 41 C. Les utilisateurs de TI 58 et TI 59 peuvent désormais aiguïser leur sagacité grâce à cette distraction tout en finesse. Une seule variante par rapport au jeu initial, mais elle est mineure : ce sont uniquement des chiffres que l'on permute. (Quand les petites Texas cesseront-elles enfin d'être analphabètes ?)

L'application « à la lettre » des règles du jeu l'aurait réservé aux seuls possesseurs de l'imprimante PC-100. Va donc pour les chiffres. Remarquez toutefois que le jeu est bien resté le même : il n'a rien perdu ni de son attrait ni de son originalité.

_____ Chaque chiffre _____
_____ à sa place _____

Au début de chaque partie, le micropoche affiche les dix chiffres 1, 2, 3, 4, 5, 6, 7, 8, 9 et 0 dans un ordre quelconque, par exemple 1254768093. Le but du jeu est de rétablir l'ordre « naturel » : 1234567890.

Chacun des chiffres est repéré grâce à sa position dans la suite par un nombre compris entre 1 et 10. Dans l'exemple que nous avons choisi, le chiffre 5 est en position 3.

Jouer une position p, c'est (soyez très attentif !) dans une première phase permuer le chiffre en position p avec celui qui se trouve en position 1. En conservant notre exemple, jouer 3 conduit à permuer le chiffre en position 3 (ici le 5) avec celui qui se trouve au début de la suite (en l'occurrence le 1) ; ce qui nous donne alors 5214768093. Mais ce n'est pas fini : dans un deuxième temps, on opère une inversion entre les deux chiffres situés de part et d'autre de la position p initialement choisie. Dans notre cas, cette inversion porte entre le 2 et le 4. En

jouant la position 3, on aboutit donc finalement à 5412768093.

Dans le cas où l'on jouerait l'une des deux positions extrêmes (le 1 ou le 10), le phénomène d'inversion se produit tout de même : tout se passe alors comme si la suite de chiffres se « mordait la queue ». Autrement dit, le chiffre placé à droite de celui qui se trouve en position 10 est en réalité celui qui est en position 1. De la même façon, à la gauche de la position 1 se trouve la position 10.

Peut-on gagner en moins de coups ?

Si l'on reprend notre exemple (1254768093) et que l'on joue 10, on a d'abord inversion entre le 3 et le 1 (3254768091), puis échange entre le 3 — position n° 1 à droite de la

Exemple de partie

1254768093.
9.
9254768310.
7.
8254739610.
8.
6254731890.
6.
3254167890.
4.
4213567890.
2.
1423567890.
3.
2314567890.
2.
1234567890.

position n° 10 — et le 9 — position n° 9 à gauche de la position n° 10. Le résultat final est alors 9254768031.

Voilà, je vous ai tout dit. A vous de jouer maintenant : remettez bon ordre dans les dix chiffres.

□ Jean Simoni

Mode d'emploi du programme

Combinaison de départ manuelle :

- 1, inscrire au clavier une combinaison de départ puis presser sur A,
- 2, choisir une position de 1 à 10, puis presser sur R/S. Après quelques secondes apparaît furtivement à l'affichage le score (nombre de coups déjà joués) puis la nouvelle combinaison. Dès que l'on atteint la combinaison gagnante, elle s'affiche alternativement avec le score.

Combinaison de départ générée aléatoirement :

Entrer un nombre-source quelconque puis presser sur B. Après quelques secondes, la combinaison de départ apparaît. Continuer alors le jeu comme en 2. Les fois suivantes, il ne sera pas utile de réintroduire un nombre-source : on pressera seulement sur C. Pour obtenir de nouveau la combinaison de départ de la partie précédente, appuyer sur D, puis reprendre en 2.

Remarque : Quand le 0 occupe la position 1, il n'apparaît pas à l'affichage.

ABONNEZ-VOUS A

L'Ordinateur de poche

C'est la seule revue française exclusivement dédiée à l'informatique de poche.

Si vous possédez déjà un ordinateur de poche ou une calculatrice programmable, vous trouverez dans L'ORDINATEUR DE POCHE

des tas d'astuces qui vous permettront de tirer un meilleur parti de votre machine.

Si vous envisagez d'en acheter une, L'ORDINATEUR DE POCHE sera pour vous un guide de choix irremplaçable.

N'hésitez pas à investir 45 Francs^(*) pour une meilleure information.

Retournez aujourd'hui même le bulletin d'abonnement ci-dessous.

Bulletin à retourner à
L'ORDINATEUR DE POCHE Service Abonnements
41 rue de la Grange aux Belles 75483 Paris Cedex 10

Nom _____

Prénom _____

Adresse _____

Pays _____ Code postal _____

Ville _____

Veuillez m'abonner pour un an (4 numéros)
à L'ORDINATEUR DE POCHE,
ci-joint mon règlement de 45 FF.

(*Etranger : 60 FF, Belgique : 320 FB, Suisse : 18 FS).

(Tarif par avion : Afrique, Moyen-Orient : 110 FF ; autres pays : 85 FF)

ORDINATEURS SHARP



PC 1211 AVEC IMPRIMANTE	2 000 F
PC 1211 AVEC IMPRIMANTE ET MAGNETOPHONE	2 400 F
MZ 80 K	6 600 F
MZ 80 B	11 000 F

CREDIT GRATUIT 3 MOIS

MICROEDITIONS INFORMATIQUES
116, RUE DE PARIS - 93100 - MONTREUIL
TEL. : 857.96.33 - METRO ROBESPIERRE

A cette époque-là, très peu de personnes se doutaient de ce que l'on allait pouvoir mettre dans sa poche.

C'était en 1975.

Rétrospectivement, on s'aperçoit que la HP 25 préfigurait certaines des machines actuelles.

■ La HP 25 est, à ma connaissance, la première calculatrice programmable de poche commercialisée par Hewlett-Packard ; elle date de 1975. A la différence du Compucorp, présenté dans le numéro précédent, il s'agit réellement d'une machine de poche : elle mesurait (elle mesure encore, puisque la mienne est toujours en état de marche) $129 \times 67 \times 29$ mm pour un poids de 170 g, batterie comprise. Elle était livrée avec sa housse, son chargeur de batterie et deux manuels : manuel d'utilisation et manuel d'application. La mémoire était bien entendu volatile, mais dès 1976 apparaît une HP 25C à mémoire constante.

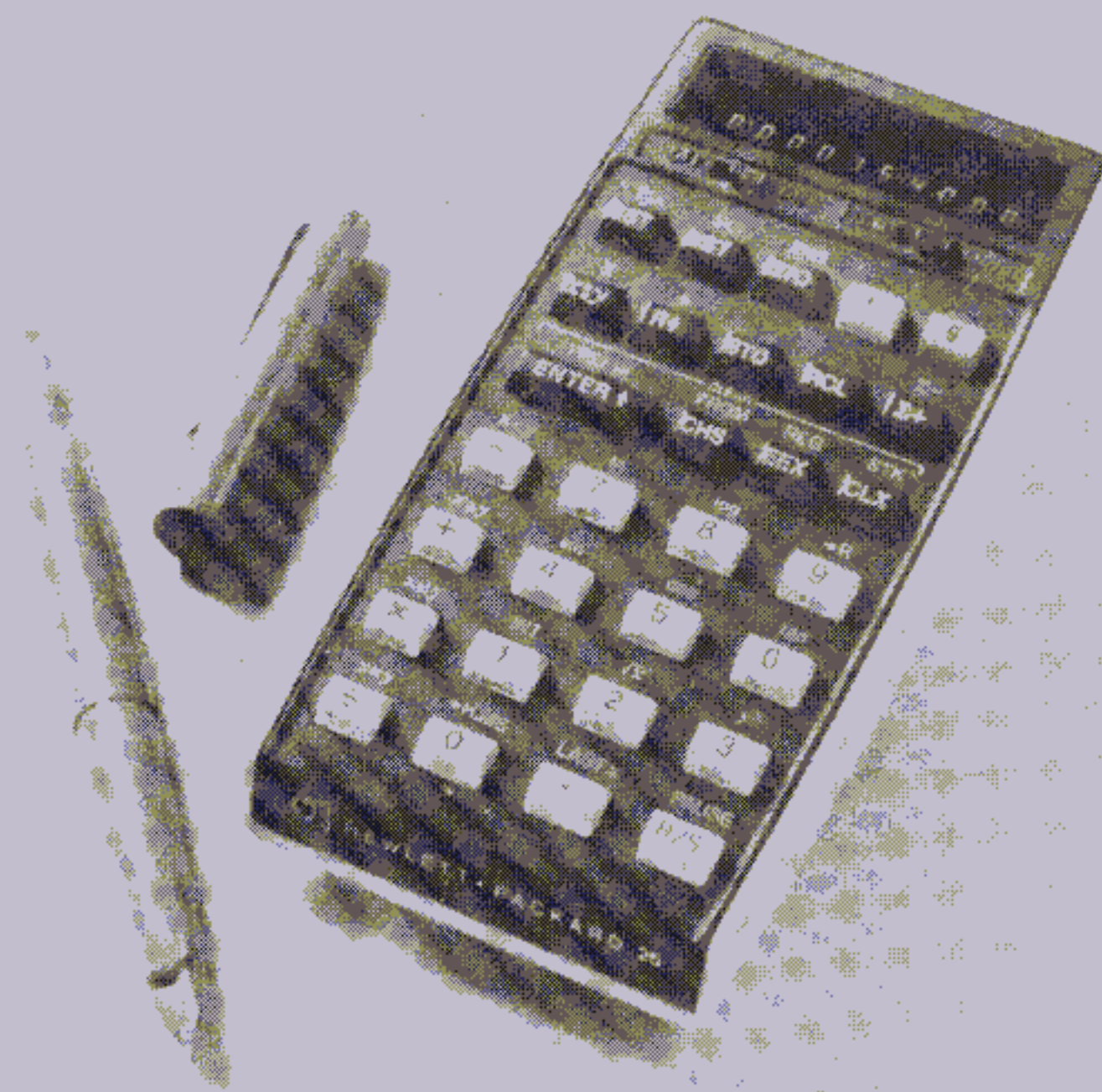
Physiquement, la machine se présente dans un petit boîtier beige dont la façade est noire. L'affichage s'effectue grâce à des diodes électroluminescentes (DEL) rouges. On peut obtenir jusqu'à 10 chiffres en notation fixe, 8 chiffres de mantisse et 2 pour l'exposant en notation scientifique ou ingénieur.

Les calculs, les résultats et les stockages utilisent dix chiffres. Le nombre π est également mémorisé sous forme de dix chiffres. Il y a huit registres de mémoire auxquels on accède grâce à RCLn et STOn avec possibilité d'opérations sur ces registres (STO+n, etc.).

Arithmétique. Le clavier est standard et les opérateurs arithmétiques sont alignés dans la colonne de gauche. La machine utilise déjà la notation polonaise inverse avec une pile à 4 niveaux et un registre Last x qui recueille le dernier nombre affiché avant l'exécution d'une opération ou d'une fonction. On trouve déjà les différentes touches de services du genre $x \Rightarrow y$, ENTER ↑, etc.

Fonctions. A l'exception de $\Sigma+$, toutes les fonctions sont préfixées par f (touche jaune) ou g (touche bleue). On trouve, pour l'essentiel :

- $1/x$, \sqrt{x} , X^2 , y^x , INT, FRAC, ABS, %
- \sin , \cos , \tan , \sin^{-1} , \cos^{-1} , \tan^{-1} , R→P, P→R dans les 3 modes trigo
- Ln, LOG, e^x , 10^x



- conversions sexagésimal \Rightarrow décimal
- statistiques : $\Sigma+$, $\Sigma-$, \bar{x} , s (variance). Des 6 sommations usuelles (Σx , Σx^2 , Σy , Σy^2 , Σxy , n), 5 seulement sont réalisées, en raison probablement du faible nombre de mémoires ; Σy^2 , qui manque, doit être programmé si l'on en a besoin, de même que \bar{y} et S_y .

Programmation

- Permutateur RUN/PRGM, touche R/S, touche NOP (no operation) PAUSE ;
- 8 tests de comparaison $x < y$, $x \geq y$, $x \neq y$, $x = y$, $x < 0$, $x \geq 0$, $x \neq 0$, $x = 0$;
- branchement par GTO avec adressage absolu (n° de ligne) — d'où la présence d'une touche NOP —, GTO 00 renvoie le pointeur au pas 00 et stoppe le programme ;
- 49 pas de programme (7 registres de 7 octets) avec une instruction complète par pas (STO+3, GTO 16, par exemple), un pas par chiffre ou assimilé introduit ;
- ni étiquettes, ni adressage relatif, donc ni sous-programme, ni compteur de boucle, ni drapeau, ni possibilité d'insertion d'une instruction nouvelle ou oubliée ;
- touches SST et BST pour la mise au point, la revue des programmes ou l'exécution pas à pas en mode RUN.

Manuel d'utilisation : 118 pages, format livre de poche.

Manuel d'applications. Comportant 160 pages, il présente 54 programmes relatifs à diverses disciplines (algèbre, finances, jeux, navigation, calcul numérique, statistiques, topographie et géométrie). De caractère pratique, ces programmes sont bien introduits — formules, principes — mais la programmation laisse souvent à désirer : citons en particulier la navigation loxodromique

et de nombreux calculs statistiques. Autre exemple : l'ajustement de courbe ; les ajustements de données se font sur plusieurs types de courbes dont 4 sont classiques : la régression linéaire ($y = a + bx$), la fonction logarithmique ($y = a + b \ln x$), la fonction exponentielle ($y = a \cdot e^{bx}$) et la fonction puissance ($y = a \cdot x^b$) ; on peut avoir à trouver quel type de courbe s'ajuste le mieux à un ensemble de données : si l'on essaye les 4 ajustements à la suite, il faudra entrer 44 pas, puis 45, puis 46, puis 47 pas de programme, les valeurs estimées $\hat{y}(x)$ étant calculées "à la main" ; ces 4 programmes peuvent en fait être réduits à un seul de 40 pas (+ 6 pas pour calculer $\hat{y}(x)$).

Six ans après... Toujours alerte, toujours aimée. Toujours aimée pour de nombreuses raisons : pour sa *fiabilité* tout d'abord — seule dépense, une nouvelle batterie en 1981 —, pour son *endurance* — je l'ai arrêtée après plus de 3 jours de fonctionnement continu —, et parce qu'elle m'a beaucoup appris.

Malgré ses capacités réduites en pas et en mémoires, elle autorise la programmation de problèmes assez complexes, une part de ses facultés provenant des facilités offertes par la pile opérationnelle : des réussites, mais aussi des échecs pour le moins aussi enrichissants.

Des faiblesses, la HP 25 en avait aussi :

- une batterie un peu "courte" surtout avec la mémoire volatile ; heureusement, le chargeur pouvait se brancher et se débrancher sans perturber le fonctionnement ;
- une fonction 10^x un peu insuffisante : $8 \cdot 10^x$, par exemple, ne donne pas 10^8 , ce qui est gênant pour un certain nombre de problèmes d'arithmétique en particulier (y^x souffre également du même mal). Cette insuffisance subsistera d'ailleurs sur la HP 25C.

Un peu d'histoire pour terminer. La HP 25 est apparue en juillet 1975 ; la HP 25C, à mémoire continue, fut proposée en juin 1976. Mais dès juillet 1976 elles avaient la concurrence de la HP 29C, puis de la HP 19C (avec imprimante) en septembre 1977.

Je laisse à un utilisateur le soin de présenter la HP 29 (ou 19)C, ne m'étant pas payé le luxe de m'offrir toutes les productions HP ! Quelle qu'en fut mon envie.

□ Jean Thiberge

Pour mieux choisir "votre" ordinateur et pour mieux l'utiliser.



Lisez

L'ORDINATEUR INDIVIDUEL

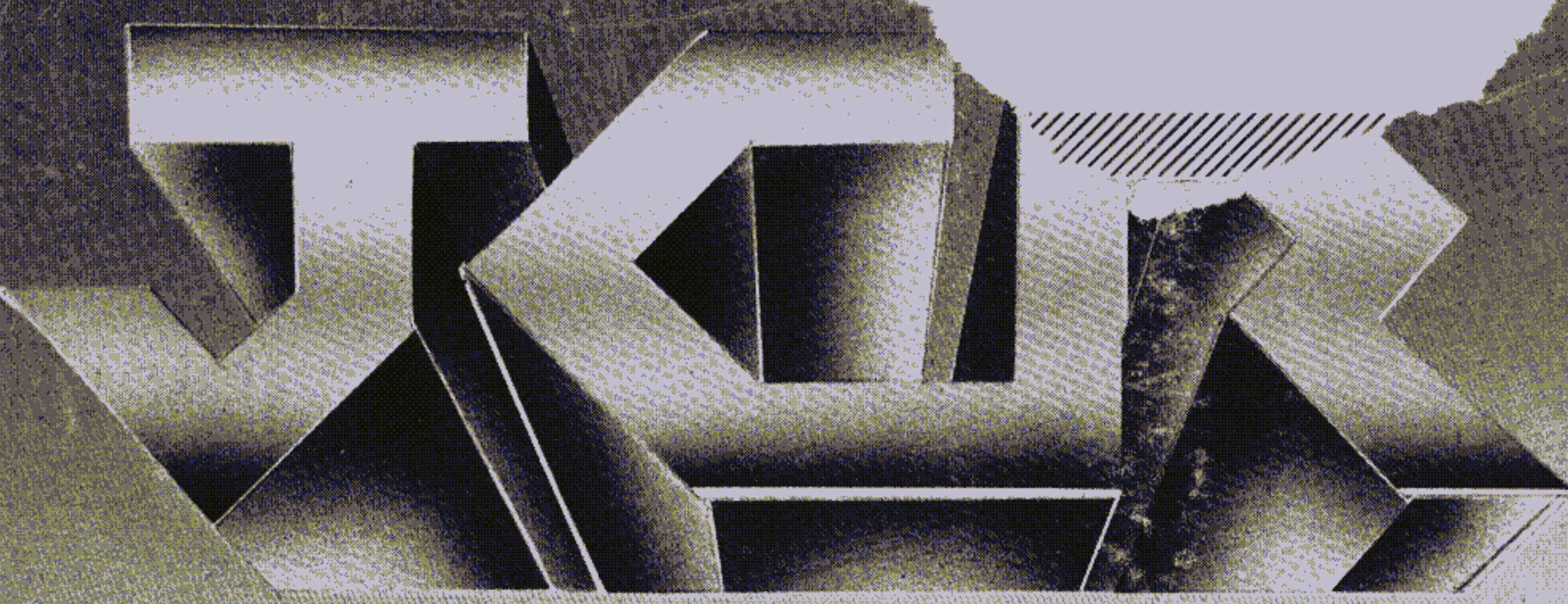
Vous y trouverez :

L'actualité et les tendances de l'informatique individuelle • des galops et des bancs d'essai des principaux matériels • des panoramas et des tests comparatifs • le point des grandes manifestations internationales • des articles d'initiation • des synthèses • des programmes • des interviews "exemplaires" • des conseils • des idées • des astuces.

L'ORDINATEUR
INDIVIDUEL

chez votre marchand de journaux

41 rue de la Grange aux Belles - 75010 Paris



SHARP PC 1211

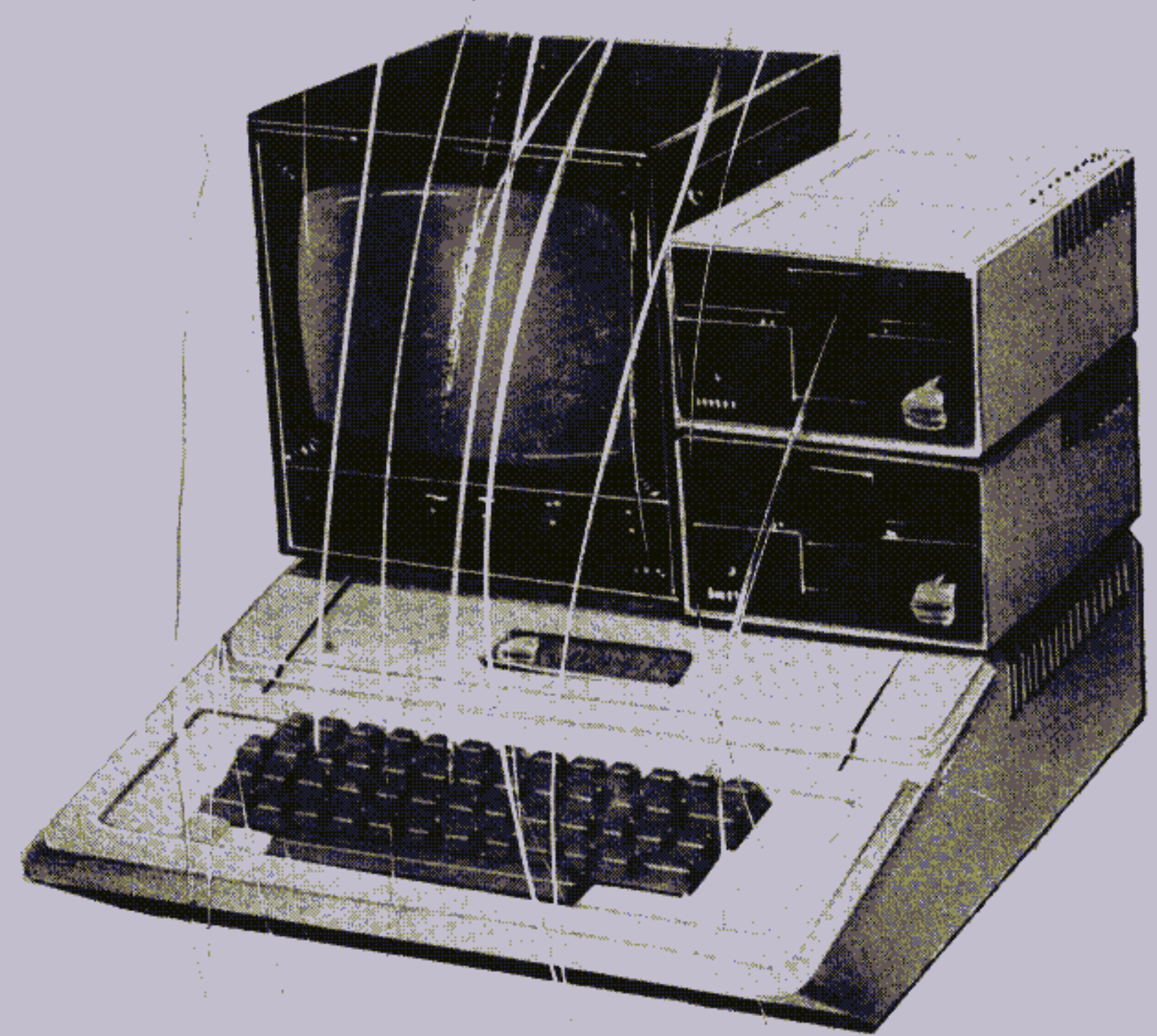
PROMO SHARP PC 1225
SHARP PC 1211 + CE 122
1900 F TTC

PROMO VIDE GENIE
VIDEO GENIE FG 3003
+ 10 K7 VIDEOS
4150 F TTC

PROMO VICTOR LAMBDA
VICTOR 16 K + PERITEL
2 MANÈTES DE JEUX
4 PROGRAMMES JEUX
3500 F TTC

	PRIX F.T.T.C.		PRIX F.T.T.C.
APPLE II + 16 K	8 500	CARTE APPLESOFT	1 450
APPLE II + 32 K	8 750	CARTE PASCAL	3 240
APPLE II + 48 K	8 990	CARTE LANGAGE	1 580
APPLE III 128 K	26 000	CARTE MICROSOFT Z80	2 750
DISK II + CONTROLEUR	4 720	TABLETTE GRAPHIQUE	5 200
DISK II	3 375	CARTE IEEE 488	3 100
DISK III SUPPL	4 400	CARTE PROTOTYPE	160
DISK 8" 2*256 K	19 150	CARTE PROTO APPLE III	390
DISK 8" 2*512 K	24 100	CARTE RECON.PAROLE	1 450
DISQUE DUR 10 MGA	41 500	CARTE MICROMUSIC	1 320
MODULATEUR N/B KIT	220	CARTE APPLECLOCK	2 160
CARTE COULEUR SECAM	1 050	CARTE TIMER CCS	1 150
CARTE COULEUR R.V.B.	920	CARTE VIDEOTHERM 80 C	2 970
INTERFACE //	1 350	CARTE SMARTEM 80 C	2 180
INTERFACE // APPLE III	1 600	SYLENTYPE II	2 800
INTERFACE V24.RS 232	1 450	KIT SYLENTYPE II → III	270
CARTE INTEGER	1 450		

Promotion exceptionnelle - Quantité limitée
Apple II 48K + Disk II avec Contrôleur
+ Moniteur Vert 9" + 10 Disquettes
13 950 F.T.T.C.



- * CREDIT - LEASING
- * STOCK IMPORTANT
- * CHOIX EXCEPTIONNEL
- * GARANTIE 1 AN P et M-O.
- * PRIX SPECIAUX PAR QUANTITE
- * DETAXE A L'EXPORTATION
- * EXPEDITION DANS TOUTE LA FRANCE

J.C.R. Electronique - 58, rue Notre-Dame-de-Lorette - 75009 Paris - Tél. (1) 282.19.80

Expedition dans toute la France - Matériel garanti 1 an pièces et m.o. - Ouvert du mardi au samedi de 10 h à 13 h et de 14 h à 19 h
En raison des fluctuations monétaires des prix sont susceptibles d'être modifiés sans préavis. Nous consulter pour confirmation.