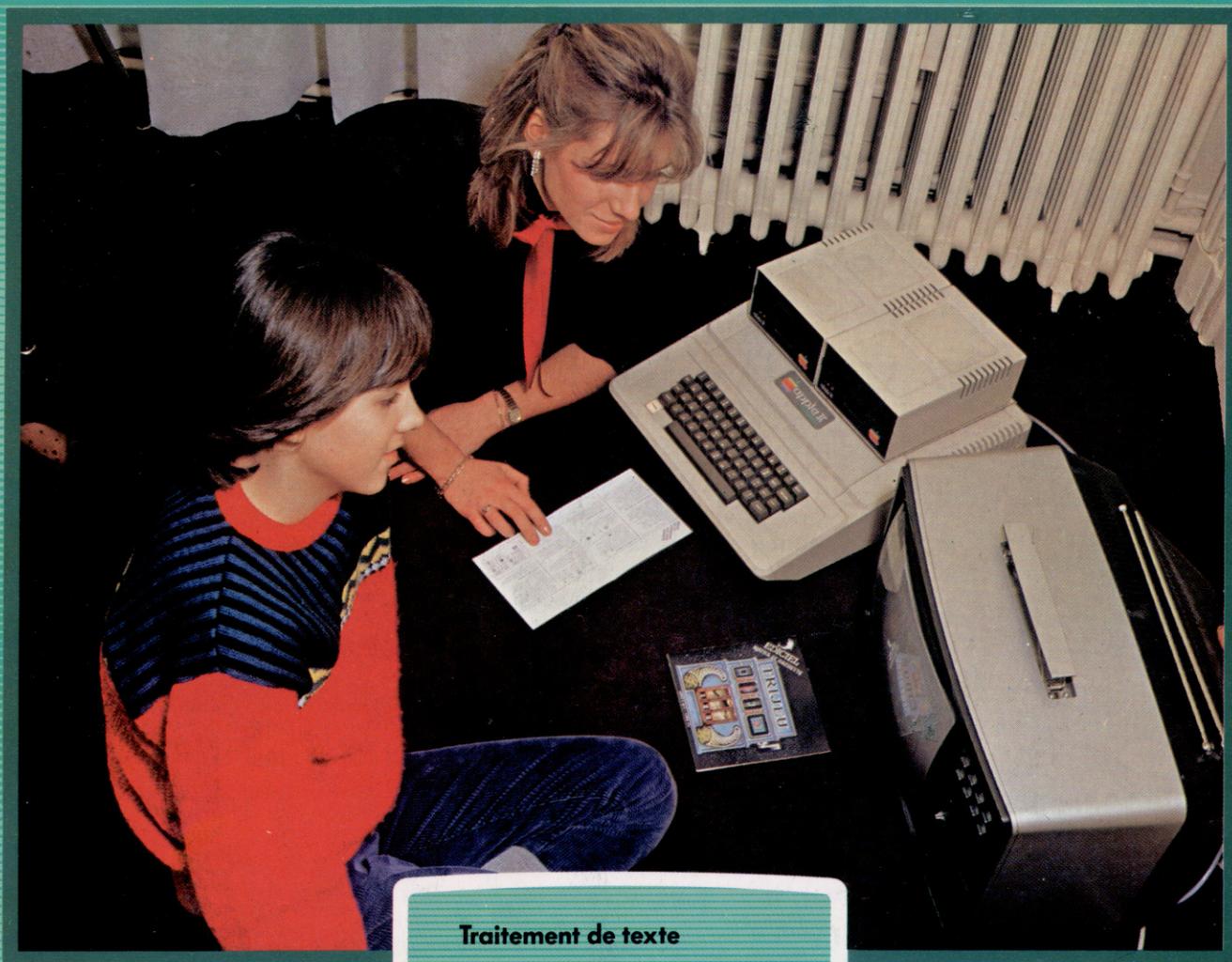


abc

N° 26

COURS
D'INFORMATIQUE
PRATIQUE
ET FAMILIALE

INFORMATIQUE



Traitement de texte

EAO

Choisissez votre moniteur

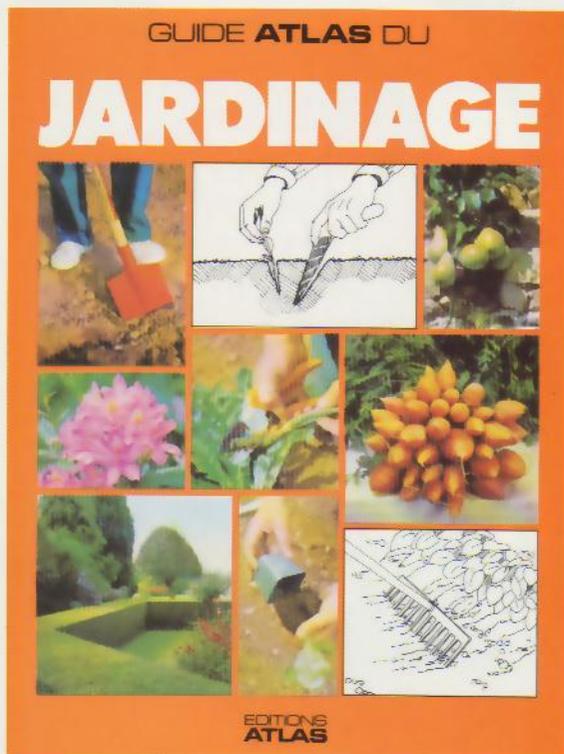
Mémoire organisée

EDITIONS
ATLAS

M6062-26-12F

85FB-3,80FS- $\$$ 1.95

Dans toutes les librairies



Guide Atlas du jardinage

Comment cultiver, entretenir et multiplier légumes, fleurs et arbres fruitiers?
 Comment organiser son jardin et quels végétaux choisir?
 Ce guide pratique répond à toutes ces questions et donne les conseils indispensables à tout jardinier, qu'il soit débutant ou chevronné. Des illustrations détaillant geste par geste les techniques de jardinage, ainsi que des encadrés donnant des tours de main contribueront à faire de vous un jardinier confirmé.

*Un volume relié,
 sous couverture illustrée.
 416 pages.
 1 200 dessins en noir et blanc.
 Format : 19,5 x 26,5 cm.*

Dans toutes les librairies



Motocross technique et compétition

Sport exaltant qui permet de goûter maintes satisfactions que ne procure pas la vie quotidienne, le motocross demande aussi une grande expérience technique et un pilotage sûr. Cet ouvrage documenté donne tous les conseils pratiques qu'il est indispensable de connaître avant de se lancer sur les pistes. Ses attrayantes photographies enthousiasmeront les futurs champions de motocross.

*Un volume relié,
 couverture illustrée. 144 pages.
 151 photos en couleurs.
 62 photos en noir et blanc.
 15 dessins en couleurs et en noir et blanc.
 Format : 22,5 x 29 cm.*



Édité par ÉDITIONS ATLAS s.a., tour Maine-Montparnasse, 33, avenue du Maine, 75755 Paris Cedex 15. Tél. : 538-52-70.

Belgique : ÉDITIONS ATLEN s.a., Bruxelles.

Canada : ÉDITIONS ATLAS CANADA Ltée, Montréal Nord.

Suisse : FINABUCH s.a., ÉDITIONS TRANSALPINES, Mezzovico.

Réalisé par EDENA s.a., 29, boulevard Edgar-Quinet, 75014 Paris. Tél. : 320-15-01.

Direction éditoriale : J.-Fr. Gautier. Service technique et artistique : F. Givone et J.-Cl. Bernar. Iconographie : J. Pierre. Correction : B. Noël.

Publicité : Anne Cayla. Tél. : 202-09-80.

VENTE

Les numéros parus peuvent être obtenus chez les marchands de journaux ou, à défaut, chez les éditeurs, au prix en vigueur au moment de la commande. Ils resteront en principe disponibles pendant six mois après la parution du dernier fascicule de la série. (Pour toute commande par lettre, joindre à votre courrier le règlement, majoré de 10 % de frais de port.)

Pour la France, s'adresser à ÉDITIONS ATLAS, tour Maine-Montparnasse, 33, avenue du Maine, 75755 Paris Cedex 15. Tél. : 538-52-70.

Pour les autres pays, s'adresser aux éditeurs indiqués ci-dessous.

SOUSCRIPTION

Les lecteurs désirant souscrire à l'ensemble de cet ouvrage peuvent s'adresser à :

France : DIFFUSION ATLAS, 3, rue de la Tave, 28110 Lucé. Tél. : (37) 35-40-23.

Belgique : ÉDITIONS ATLEN s.a., 55, avenue Huart-Hamoir, 1030 Bruxelles. Tél. : (02) 242-39-00. Banque Bruxelles-Lambert, compte n° 310-0018465-24 Bruxelles.

Canada : ÉDITIONS ATLAS CANADA Ltée, 11450 boulevard Albert-Hudon, Montréal Nord, H 1G 3J9.

Suisse : FINABUCH s.a., ÉDITIONS TRANSALPINES, zona industriale 6849 Mezzovico-Lugano. Tél. : (091) 95-27-44.

RELIEZ VOS FASCICULES

Des reliures mobiles permettant de relier 12 fascicules sont en vente chez votre marchand de journaux.

ATTENTION : ces reliures, présentées sans numérotation, sont valables indifféremment pour tous les volumes de votre collection. Vous les numéroterez vous-même à l'aide du décalque qui est fourni (avec les instructions nécessaires) dans chaque reliure.

En vente tous les vendredis. Volume III, n° 26.

ABC INFORMATIQUE est réalisé avec la collaboration de Trystan Mordrel (secrétariat de rédaction), Jean-Pierre Bourcier (coordination), Patrick Bazin, Jean-Paul Mourlon, Claire Rémy (traduction), Ghislaine Goullier (fabrication), Marie-Claire Jacquet (iconographie), Patrick Boman (correction).
 Crédit photographique, couverture : Jean Riby-Matra-Hachette.

Directeur de la publication : Paul Bernabeu. Imprimé en Italie par I.G.D.A., Officine Grafiche, Novara. Distribution en France : N.M.P.P. Tax. Dépôt légal : juin 1984. 29846. Dépôt légal en Belgique : D/84/2783/27.

© Orbis Publishing Ltd., London.
 © Editions Atlas, Paris, 1984.

A NOS LECTEURS

En achetant chaque semaine votre fascicule chez le même marchand de journaux, vous serez certain d'être immédiatement servi, en nous facilitant la précision de la distribution. Nous vous en remercions d'avance.

Les Éditions Atlas

EDITIONS ATLAS EDITIONS ATLAS EDITIONS ATLAS EDITIONS ATLAS



L'EAO arrive

L'ordinateur et l'enseignement font bon ménage. D'une part, l'informatique va progressivement être enseignée à l'école; d'autre part, l'enseignement commence à se servir des ordinateurs.

Entrés à l'école à titre expérimental dans cinquante-huit lycées pilotes, de 1971 à 1978, les micro-ordinateurs sont aujourd'hui, en France, au nombre de douze mille — essentiellement dans les lycées, c'est-à-dire réservés aux classes de seconde, première et terminale. Peu de collèges (classes de sixième à troisième) en bénéficient et il n'en existe pratiquement pas dans le primaire.

Mais, depuis le printemps 1983, le ministère de l'Éducation nationale a un nouveau projet. « Cent mille micros » devront être installés dans les écoles d'ici à 1988, projet assorti de la formation d'autant d'enseignants.

L'EAO (enseignement assisté par ordinateur) en France n'en est encore qu'au stade de l'expérimentation et des recherches, et les cent mille ordinateurs seront loin de satisfaire les besoins des écoliers qui fréquentent quelque soixante-treize mille établissements scolaires.

Actuellement, à part l'option informatique qui offre trois heures hebdomadaires d'initiation au maniement de l'ordinateur et quelques clubs animés par des professeurs volontaires, l'essentiel de l'activité des lycéens sur ordinateur consiste en travaux dirigés appliqués aux matières étudiées : mathématiques, français, géographie, langues, etc. L'élève travaille avec l'ordinateur sans avoir besoin d'apprendre à s'en servir. Il lui suffit de mettre en place une cassette ou une disquette, d'appuyer sur des touches de commande (RUN ou VALID). Sur l'écran de l'ordinateur s'affichent alors des questions auxquelles il lui faut répondre, ou bien des phrases incomplètes dont il devra remplir les « trous ». Chaque élève peut ainsi avancer à son rythme, ne passant à la question suivante que lorsqu'il a assimilé ce qui précède. S'il ne sait pas répondre, ou lorsqu'il a fourni une réponse inexacte, l'ordinateur lui donne le bon résultat sans jamais perdre patience. Le professeur n'en est pas pour autant devenu inutile, mais les rapports entre maître et élèves se font sur un autre plan : les professeurs deviennent conseillers et animateurs.

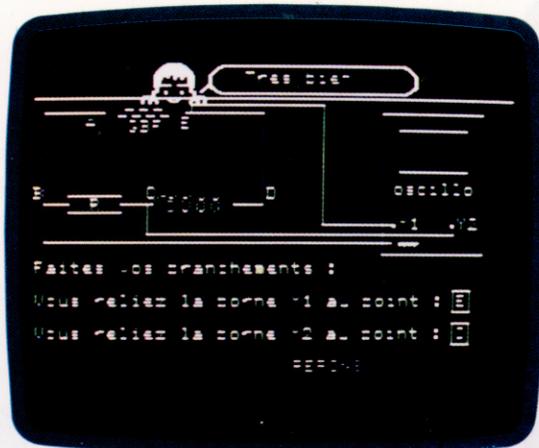
Plusieurs éditeurs français publient des logiciels d'enseignement, également appelés « didac-

Un bon départ

Beaucoup de propriétaires d'ordinateurs domestiques achètent des logiciels éducatifs en pensant à l'avenir de leurs enfants. Pour les plus jeunes, une grande variété de

programmes est disponible afin de les initier au processus d'apprentissage. Les élèves préparant le baccalauréat disposent également de beaucoup de logiciels de révision dans toutes les matières.



Physique


ticiels ». Ceux-ci fonctionnent sur divers matériels. L'Apple II, le TO7 de Thomson et l'Atari, entre autres, disposent d'une ample bibliothèque de didacticiels.

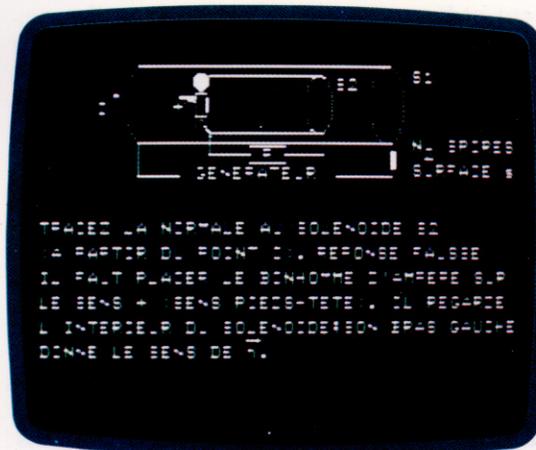
Cependant, les plus nombreux sont ceux qui sont adaptés aux micro-ordinateurs Micral de R2E, au Logabax et au SIL'Z de Léonard; et n'oublions pas les nombreux programmes d'enseignement qui peuvent être obtenus à partir d'un terminal Minitel, réalisés par Didao. Ces logiciels sont souvent accompagnés d'un manuel qui explique le sujet et permet de préparer les exercices avant de les exécuter soit même sur micro-ordinateur ou sur terminal que l'on possède chez soi.

Pour les plus jeunes élèves, les logiciels d'enseignement se présentent généralement sous la forme de jeux au graphisme attrayant : de petits personnages ou animaux invitent les écoliers à additionner des nombres ou à conjuguer des verbes. Pour eux, la manipulation du clavier est généralement remplacée par l'utilisation de « manches à balai » ou de crayons optiques qui ne nécessitent pas de connaissances préalables des chiffres et des lettres.

De même, la pratique du LOGO ne requiert de la part de l'enfant aucune connaissance particulière. Dès la maternelle, il pourra s'initier au maniement de la tortue à l'aide d'instructions abrégées comme AV pour AVANCE et DR pour DROITE. L'apprentissage du langage et de la programmation en LOGO constitue un éveil au raisonnement et à la logique, ainsi qu'une introduction à des notions mathématiques (comme les angles, les nombres, les lignes droites ou courbes), ou physiques (il est possible de simuler, à l'aide du LOGO, la trajectoire d'un objet en chute libre, par exemple).

L'apprentissage des langues étrangères est l'un des domaines d'application privilégiés de l'EAO. Les premières tentatives d'enseignement mécanisé furent les « laboratoires de langues » des années 1960. Chaque étudiant disposait d'un magnétophone avec un texte préenregistré, sur lequel il pouvait travailler à son propre rythme, sous la surveillance d'un professeur qui avait accès à tous les postes de ses étudiants pour intervenir.

Dans l'EAO, cette intervention est effectuée par l'ordinateur lui-même qui corrige immédiatement les erreurs de l'élève, ou bien lui laisse une autre possibilité de réponse. Un bon logiciel d'enseignement est, en quelque sorte, un système expert : c'est-à-dire qu'il doit pouvoir répondre à toutes les défaillances de l'élève, donner toujours des résultats exacts et, en outre, permettre à l'élève d'avancer dans la matière concernée de manière simple et natu-

Physique


relle. C'est pourquoi, ici, l'ordinateur doit être aussi facile à utiliser que possible et ne doit pas exiger des compétences en informatique ou en programmation.

Les logiciels d'enseignement ne sont pas nécessairement destinés à l'utilisation dans des locaux scolaires. Au contraire, même si les élèves disposaient d'un nombre suffisant d'ordinateurs dans leurs classes, la présence d'un micro-ordinateur domestique serait encore profitable pour approfondir à la maison les notions apprises à l'école. Il est significatif que l'une des premières raisons invoquées pour l'achat d'un micro-ordinateur familial soit l'éducation.

En ce qui concerne le contenu proprement dit des logiciels, il faut faire une distinction entre les scientifiques et les littéraires. Pour les premiers, où l'on traite de faits connus et quantifiables, les illustrations sont généralement nombreuses et attrayantes, même sur les micro-ordinateurs possédant de faibles possibilités de résolution graphique. Pour les logiciels littéraires, en revanche, ce sont essentiellement des textes qui sont affichés.

Outre cette distinction, on peut classer les didacticiels en trois catégories : les logiciels d'enseignement à proprement parler, dont le but est d'expliquer des faits et de faciliter leur mémorisation; les logiciels de révision, qui fonctionnent suivant le principe des questions-réponses; et enfin les logiciels de simulation expérimentale, qui impliquent des techniques de programmation bien plus complexes que les deux premières catégories, puisque le programme est censé reproduire mathématiquement l'interaction de forces physiques.

Considérons à présent les quatre principaux groupes d'âge auxquels sont destinés les logiciels éducatifs, afin d'en déduire les spécificités.

Maternelle

La plupart des logiciels destinés aux plus jeunes enfants ont trait à la reconnaissance des formes et à l'orientation. C'est ainsi que l'on trouve des jeux d'orientation gauche-droite, et une « ronde des chiffres » qui familiarise l'enfant avec la forme des chiffres et leur rapport avec l'idée de grandeur. Les plus populaires sont les programmes pour apprendre à lire l'heure, à compter, et pour s'exercer à la lecture globale à l'aide de phrases simples dont les mots, mêlés de façon aléatoire, doivent être remis en ordre afin de reconstituer la phrase. Tous ces programmes se présentent sous la forme de jeux, qui aident l'enfant à soutenir son attention et éveillent son intérêt.

Mathématiques



De six à dix ans

Si cette classe d'âge demande encore des programmes plus ludiques que sérieux, l'approche fait appel à plus de discipline, et reproduit les expériences de l'enfant pendant la classe.

Une grande variété de programmes de mathématiques et de français sont proposés aux jeunes écoliers : les quatre opérations arithmétiques sous la forme d'exercices avec correction immédiate, ou l'entraînement au calcul mental ; avec les logiciels de géométrie, l'enfant apprendra à se repérer dans un plan quadrillé ; des problèmes présentés sous forme de jeux offrent un entraînement au raisonnement mathématique ; l'orthographe et la grammaire seront approfondies à l'aide de logiciels proposant des exercices ; une initiation à la lecture se fera à partir de jeux où l'enfant trouvera une succession de lettres ou de syllabes dans un mot. Il existe aussi des logiciels servant de support aux activités d'éveil : en géographie, l'élève apprendra à reconnaître les principaux éléments géographiques de son pays : villes, fleuves, montagnes et mers qui l'entourent.

1^{er} cycle du secondaire (de dix à quinze ans)

Des logiciels similaires à ceux proposés aux classes primaires sont offerts aux élèves des collèges. S'y ajoutent notamment les programmes d'apprentissage des langues vivantes, ainsi que des grammaires latine et française.

2^e cycle du secondaire (plus de quinze ans)

C'est aux élèves des lycées que s'adressent la plupart des didacticiels, hormis les logiciels de jeux. Des programmes de mathématiques présentent et illustrent des concepts à l'aide d'exemples d'applications ; ils concernent notamment la logique fondamentale, la théorie des ensembles, les fonctions, le calcul des probabilités et les statistiques. En sciences physiques, les logiciels les plus populaires sont les programmes de simulation qui permettent un niveau élevé d'expérimentation, tout en évitant le danger des réactions chimiques ! En littérature française, des analyses de texte ou des exercices de versification sont proposés. Dans cette catégorie d'âge, une grande variété de logiciels prépare au baccalauréat.

Il ne faut pas omettre les nombreux logiciels d'initiation à la programmation, qui comprennent souvent l'enseignement du BASIC.

Mathématiques



J. Riby - Matra/Hachette

Pour ces derniers utilisateurs, suffisamment motivés par l'apprentissage, on ne constate généralement aucun effort particulier pour rendre la tâche attrayante par des moyens artificiels. Cela ne signifie pas que le sujet soit présenté de façon sommaire. Au contraire, ces logiciels sont souvent illustrés de graphiques statiques ou animés, et accompagnés d'effets sonores.

Dans un prochain numéro, nous étudierons plus en détail les différents logiciels éducatifs disponibles pour les ordinateurs domestiques les plus répandus.

Fonctions et syntaxe

Dans notre approche du BASIC Sinclair, nous abordons les fonctions VAL, GOSUB et GOTO, et les constructions WHILE...WEND et REPEAT...UNTIL.

Vous avez peut-être remarqué que certaines fonctions du BASIC Sinclair n'ont pas besoin de parenthèses, contrairement à ce qui se fait habituellement. Ainsi LEN(X\$) peut s'écrire indifféremment LEN X\$ ou LEN(X\$). Néanmoins, dans le doute, utilisez toujours des parenthèses. Cela vous fera peut-être éviter les mauvaises habitudes.

La fonction CODE est l'équivalent Sinclair de ASC() et se comporte exactement de la même façon. Le jeu de caractères autorisés, cependant, ne concerne que les équivalents ASCII de 32 à 122. Aussi PRINT CHR\$(7) renverra un message d'erreur au lieu du bip sonore normal dans la plupart des BASIC.

La fonction VAL correspond au BASIC standard, mais une instruction du type VAL("a45") égarera le programme du fait de la valeur non numérique de l'argument. Pour la plupart des autres BASIC, cela donnerait simplement zéro. Si cette bizarrerie devait vous poser problème, écrivez un sous-programme pour remplacer VAL. Vous pourrez également tester la valeur CODE du premier élément de l'argument. Si CODE A\$(1) < 48 ou si CODE A\$(1) > 57, alors A\$ est non numérique et ne peut servir d'argument à la fonction VAL.

VAL a cependant pour Sinclair l'avantage de pouvoir évaluer les expressions numériques.

```
LET A$ = "6*12":PRINT VAL A$
```

donnera 72, expression de "6*12". Pour la plupart des BASIC, VAL n'a pas cette puissance et donnerait ici la valeur 6.

Cette faculté d'évaluer les expressions peut être mise à profit de plusieurs manières. Prenons l'exemple de ce programme de représentation graphique d'une zone :

```
100 DIM S$(31)
200 LET S$ = "....."
300 INPUT «ENTREZ UNE FONCTION DE X»; F$
400 PRINT « Y = »; F$: PRINT
500 FOR X = 1 TO 10
600 PRINT S$(TO INT(VAL F$))
700 NEXT X
800 PRINT «=====»
900 PRINT «00000000111111111222222223»
950 PRINT «123456789012345678901234567890»
```

A exécution de ce programme, vous pouvez taper toute forme d'expression algébrique avec la variable X; (2*X + 3/X), par exemple. Vous aurez alors à l'écran la représentation schématisée de la fonction. Ici, l'axe des Y est horizontal, l'axe des X, vertical, les graphiques étant à l'échelle d'un pixel. Il ne faudrait pas beaucoup

plus de code pour établir l'échelle des valeurs de X et de Y immédiatement après la saisie de la fonction X. Ensuite, les ajustements d'échelle auraient lieu, les axes seraient affichés et le graphique haute résolution serait rendu possible. Le résultat serait un ensemble utilitaire graphique très abouti. Le petit programme ci-dessus est destiné seulement à illustrer l'utilité d'admettre une expression comme argument de la fonction VAL.

A cet égard, GOSUB et GOTO sont semblables à VAL. Ces fonctions autorisent également l'évaluation d'expressions dans des cas où la plupart des langages BASIC exigeraient que leurs arguments soient des numéros de ligne valides. Cet avantage est intéressant à plusieurs titres. Vous pouvez ainsi donner des noms à vos sous-programmes, définir des variables sous le même nom et avec des valeurs appropriées, pour ensuite adresser par leur nom les sous-programmes avec GOSUB.

Voici un exemple :

```
100 LET INIT = 1000
200 LET OUTPUT = 2000
300 LET CALCULATE = 3000
400 GOSUB INIT
500 GOSUB CALCULATE
600 GOSUB OUTPUT
700 STOP
1000 REM.....S P INIT.....
.....
2000 REM.....S P OUTPUT.....
.....
3000 REM.....S P CALCULATE.....
```

Le programme en lui-même est éloquent. Si vous remplacez GOSUB INIT par GOSUB(VAR1 + N*VAR2) ou par toute autre expression numérique valide, l'expression sera évaluée et le résultat sera considéré comme un numéro de ligne. En outre, avec le BASIC Sinclair, si l'argument de GOTO ou de GOSUB est un numéro de ligne qui n'existe pas, le contrôle passe au prochain numéro de ligne autorisé. Si vous écrivez GOTO 17 et que la ligne 17 n'existe pas, le contrôle passera sur la ligne 18 si elle existe, sans message d'erreur comme avec les autres BASIC.

Cette faculté du BASIC Sinclair de traiter les « sauts calculés » compense l'absence de constructions ON...GOTO ou ON...GOSUB.

Avec un autre BASIC, vous pouvez écrire :

```
2360 ON D GOSUB 100,200,300,400,500
```

Cela signifie que si la valeur de D est 1, le programme passera en 100 (GOSUB 100); si D = 2, il pas-

sera à 200 (GOSUB 200); et ainsi de suite. Avec le BASIC Sinclair, vous écrirez :

```
2360 GOSUB (100 'D)
```

avec exactement les mêmes résultats. Pour les deux versions, cependant, il vous faudra faire attention aux valeurs de D non entières, inférieures à 0 ou supérieures à 5.

Le BASIC Sinclair ne comporte pas les constructions syntaxiques REPEAT...UNTIL et WHILE...WEND. Mais ces dernières peuvent être remplacées de plusieurs manières. REPEAT est une boucle finissant avec UNTIL qui est suivie d'une expression conditionnelle. Lorsque cette dernière est vérifiée (valeur VRAIE), la boucle s'achève et l'exécution passe sur l'instruction suivante. Sur le Spectrum, cela peut s'obtenir ainsi :

```
100 DATA «A», «B», «C», «D», «E», « », «F», «G», « »
200 FOR L = 1 TO 1
300 READ X$
400 PRINT X$
500 IF X$ < > « » THEN LET L = 0
550 NEXT L
600 PRINT « FIN DE DONNÉES »
```

Le problème avec REPEAT...UNTIL provient de la position en fin de boucle de l'expression conditionnelle : la boucle sera nécessairement exécutée au moins une fois. Si cela n'est pas souhaitable, utilisez WHILE...WEND.

Il s'agit d'une boucle commençant avec WHILE (qui est suivie d'une instruction conditionnelle)

et se terminant avec WHILEIEND. La boucle est exécutée aussi longtemps que la condition inscrite entre WHILE et WEND n'est pas vérifiée. Lorsque cela arrive, c'est-à-dire lorsque la condition est vraie, le contrôle passe de WHILE sur l'instruction qui suit WEND, directement. Le programme donné plus haut s'écrirait :

```
100 DATA «A», «B», «C», «D», «E», « », «F», «G», « »
200 WHILE X$ < > « » DO
300 READ X$
400 PRINT X$
500 WEND
600 PRINT « FIN DE DONNÉES »
```

Ce qui pourrait être remplacé sur un Sinclair par une construction IF...THEN...GOTO.

Mais il existe une autre solution :

```
50 LET X$ = « »
100 DATA «A», «B», «C», «D», «E», « », «F», «G», « »
200 LET TEST = (X$ < > « »)
250 FOR L = 1 TO TEST
300 READ X$
400 PRINT X$
500 LET L = (X$ = « »)
550 NEXT L
600 PRINT « FIN DE DONNÉES »
```

Nous avons maintenant abordé les principales caractéristiques du BASIC Sinclair. Il existe d'autres différences de moindre importance, dont certaines sont plutôt des erreurs d'adaptation, mais, avec le manuel et en y prenant garde, vous devriez vous en sortir.





Traitement de texte

Le traitement de texte constitue le domaine d'application principal de la micro-informatique. Nous commençons un panorama des principaux progiciels de traitement de texte, avec WordStar.

Dans sa forme la plus simple, un traitement de texte n'est rien d'autre qu'une machine à écrire un peu sophistiquée. L'écran moniteur remplace la feuille de papier; et la touche DELETE, le ruban correcteur. Pour l'utilisateur gestionnaire, l'avantage consiste à pouvoir sauvegarder les documents sur disque ou sur cassette. Cela permet de pouvoir les utiliser librement. Il peut s'agir de phrases, de simples lignes ou encore de paragraphes entiers, de pages même. On peut les incorporer à un texte, ce qui se révèle utile pour des spécifications techniques, des contrats, et plus généralement des textes où un même passage est réutilisé.

Après l'aspect « libre manipulation » des documents (création, modification, archivage et restitution), voyons la présentation d'un document. Il existe trois présentations classiques : alignement à gauche, à droite, et ligne justifiée. Dans ce dernier cas, la longueur des lignes reste

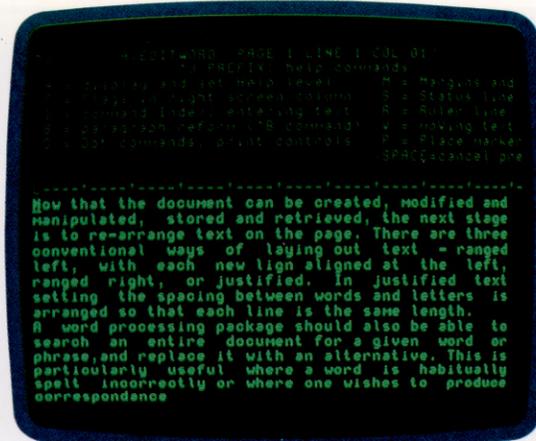
frappe de deux touches (CONTROL + touche spécifique au clavier) pour indiquer une commande. L'opérateur peut avoir un menu d'options affiché de manière permanente à l'écran, auquel cas dix lignes sont réservées en haut de l'écran. Des commandes peuvent modifier la présentation du texte, soit directement à la saisie, soit à l'impression. Elles apparaissent à l'écran, mais non à l'impression. Les tâches de gestion des fichiers et du traitement de texte sont assurées par le système d'exploitation.

Au chargement, le menu des fonctions apparaît. Il présente les fonctions possibles. L'utilisateur peut créer ou éditer un fichier, soit sous la forme d'un document, soit sous une forme non documentée. Cette dernière concerne par exemple les programmes en langage assembleur devant être compilés, ou du texte devant être utilisé par un autre progiciel. L'utilisateur peut également fusionner des fichiers (en les imprimant

Le menu des fonctions



Le menu d'aide



Le menu des fonctions (no-file menu)

Pour commencer, WordStar affiche un menu qui présente les différentes fonctions de base. L'utilisateur peut créer ou éditer soit un fichier texte à l'aide des paramètres standard de format (non-division des mots, justification, etc.), soit un fichier non-texte. Il peut aussi modifier, copier ou détruire un fichier. En résumé, c'est un menu de routines de gestion de fichiers.

Le menu d'aide (the help menu)

On y accède par CTRL + [J]. Il affiche un ensemble relativement exhaustif d'explications sur le fonctionnement de certains modules. On l'appelle familièrement le menu d'aide. Il présente plusieurs options (H, B, F, D...) qui détaillent les diverses parties de WordStar.

la même. Cela est rendu possible par des espaces de longueur variable entre les mots. La plupart des imprimantes pour micros ne font varier que l'espacement entre les mots, et non celui entre les lettres d'un mot.

Un progiciel de traitement de texte devrait également être capable de rechercher une chaîne de caractères dans un document, et de la remplacer par une autre. Cela s'applique à l'orthographe (faute courante), ou à la correspondance personnalisée (insertion du nom). En termes techniques, WordStar est un traitement de texte avec des menus et un éditeur de texte doté de commandes de formatage « transparentes » à l'utilisateur (de formulation simplifiée). Il fonctionne sous CP/M et utilise la conjonction de

mant ou pas), changer l'affectation du lecteur de travail, effacer un fichier, le copier, en changer le nom, quitter le traitement de texte et retourner sous le système d'exploitation; enfin, il peut indiquer le niveau d'aide qu'il attend du progiciel.

Les options correspondent à la frappe d'une seule touche (pas de CTRL + une touche). Chacune débouche sur un sous-menu spécifique. La dernière option, celle d'aide, permet à l'utilisateur de déterminer si les commandes possibles seront affichées en permanence. Cette dernière option comporte des degrés, laissant par exemple davantage de place pour le texte dans le cas où le niveau d'aide est le plus bas (le minimum de lignes réservées aux commandes). La plupart



des commandes d'option représentent les noms mêmes des options : D pour créer un document, N pour créer un non-document.

Le traitement de texte commence par demander un nom pour le fichier à créer. Le menu de travail apparaît alors. Il consiste en des messages relatifs au déplacement du curseur, à l'insertion ou à l'effacement, à des codes sur un seul caractère servant à accéder à d'autres branches du menu principal (au menu de formatage par exemple). WordStar comporte ses propres commandes de déplacement du curseur.

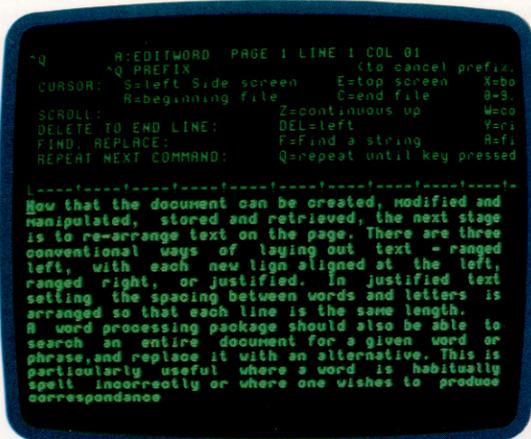
Le déplacement d'un seul caractère s'obtient au moyen de la touche CONTROL (CTRL), appuyée lors de la frappe d'une des touches suivantes : E, S, D ou X. Les machines comportant leur propre mode de déplacement du curseur implémentent WordStar de sorte qu'il y ait deux modes de déplacement possibles. WordStar est très puissant quant au contrôle du curseur. Il permet le déplacement par caractère, par mot, par ligne ou par paragraphe, dans toutes les directions. La suppression suit sensiblement le même principe. Les caractères individuels peuvent être supprimés soit sur la position du curseur, soit immédiatement à sa droite. De même, des mots entiers situés à droite du curseur, ou encore des phrases complètes (sans faire intervenir la position du curseur) peuvent être effacés. Il est possible de dérouler le texte à l'écran sans affecter la position du curseur, à l'aide des fonctions de

documents, mais WordStar dispose aussi d'autres fonctions, plus spécifiques. Ainsi, il peut centrer les lignes sur une page, provoquer plusieurs frappes afin de produire des caractères gras, surligner et indiquer les caractères ou en détacher un. Toutes ces fonctions et d'autres sont autorisées aussi bien à la saisie qu'à l'édition. La fonction « recherche et remplacement d'une chaîne » (*find and replace*) figure au menu Q (résumé des commandes), qui comporte également les autres fonctions de positionnement du curseur et d'effacement. L'utilisateur doit frapper la chaîne à rechercher. Il peut spécifier le nombre d'occurrences recherchées (le nombre de fois où la chaîne est rencontrée), ou demander toutes les occurrences de la chaîne dans le texte. L'utilisateur décide si la chaîne devra lui être simplement indiquée par un curseur, ou bien remplacée par une autre. Dans ce dernier cas, il spécifie si le remplacement doit être automatique.

Des blocs de texte, indiqués par un marqueur, peuvent de manière similaire être déplacés dans le document. Ce sont des fonctions du menu K (qui traite essentiellement de la gestion des fichiers) qui s'en chargent.

Les trois autres sous-menus de WordStar, J, O et P, traitent respectivement de l'aide à l'utilisateur pour créer ou éditer un document, du formatage, et de la présentation en format d'édition.

Le menu résumé des commandes



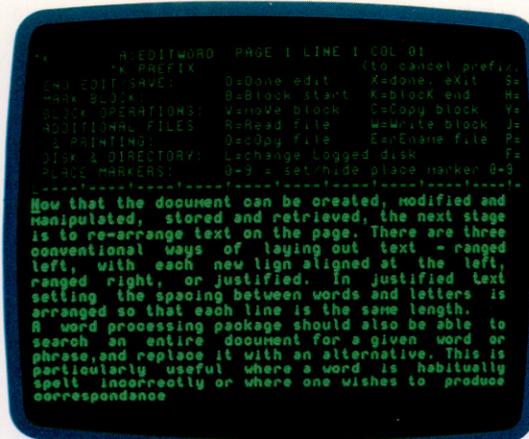
déplacement. Cela peut vous être très utile si vous désirez vous référer à une partie du texte.

En dernier lieu, l'utilisateur a la possibilité d'insérer un caractère sur la position du curseur, ou de remplacer le caractère existant. Cette fonction est selon le mode « bascule » : une première frappe de la touche en question (CTRL + V ici) fait passer le mode de l'insertion au remplacement; une deuxième, du remplacement à l'insertion. Le statut de la fonction s'inverse donc à chaque frappe.

Le progiciel WordStar assure de lui-même la pagination (nombre de lignes par page) et la non-division des mots en fin de ligne.

Les fonctions décrites constituent les outils indispensables à la création et à l'édition des

Le menu bloc de texte



WordStar est un progiciel de traitement de texte complet et professionnel. Il y a très peu de fonctions qui ne soient pas une grande réussite. Ajoutons qu'il a été conçu pour être adaptable à de très nombreuses machines.

A première vue, il peut sembler compliqué et très particulier, mais ses menus très complets en rendent néanmoins l'acquisition possible pour un débutant. La plupart des micro-ordinateurs qui implémentent WordStar comportent en standard des touches de fonction programmables.

L'Osborne 1, par exemple, permet l'utilisation de dix touches de fonction sur 256 octets. Ces fonctions sont accessibles en appuyant sur la touche CTRL et une touche numérique.

Le menu résumé des commandes

(the quick command menu)

Les sous-menus de WordStar sont accessibles par la frappe d'une seule touche, généralement une lettre rappelant le nom des fonctions. Le menu Q signifie ainsi *quick menu*. Il fournit un résumé des commandes de déplacement du curseur, ainsi que des fonctions de recherche et de remplacement de chaîne.

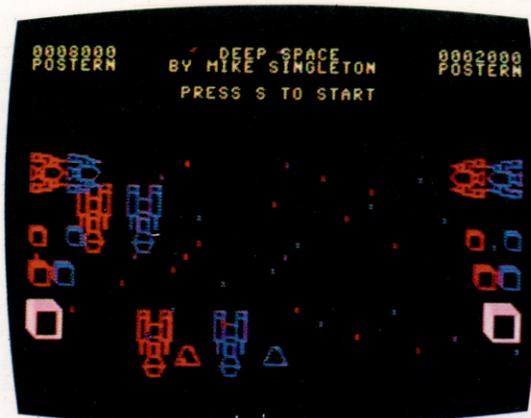
Le menu bloc de texte

(the block menu)

Certaines lettres désignant des noms de menus peuvent sembler incompréhensibles, ainsi K, par exemple. K est mis pour *block menu*. Il gère la sauvegarde et l'utilisation des fichiers, le déplacement des blocs de texte, ainsi que certaines formes d'accès aux fichiers. Il reproduit aussi (en double par rapport au menu des fonctions) certaines routines de gestion de fichiers. (Cl. Liz Heaney.)

Batailles spatiales

L'espace a donné naissance à de nombreuses variétés de jeux très prisées par les amateurs. Nous poursuivons notre description de certaines cassettes que l'on peut se procurer sur le marché.



CBM 64 3 D DEEP SPACE

Ce jeu propose une nouvelle version du classique « SPACE INVADERS ».

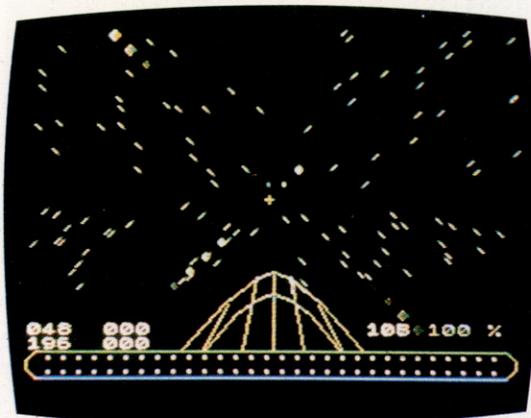
Son originalité réside dans le fait que les Aliens sont représentés en image double : l'une rouge, l'autre bleue, qui donnent l'impression du relief lorsqu'on regarde l'écran avec des lunettes spéciales.

De la même manière, votre canon laser peut se déplacer de droite à gauche et d'avant en arrière, ce qui rend assez délicate la destruction des Aliens. Mais la qualité du relief au travers des lunettes spéciales semble varier d'un individu à l'autre selon leur vue.



ATARI LUNAR LEEPER

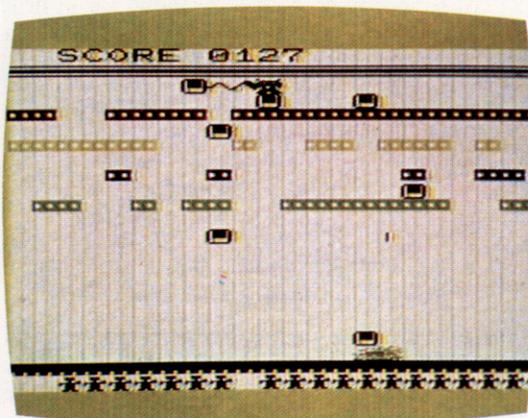
C'est un jeu « DEFENDER » organisé en deux parties. Dans la première, vous devez essayer de secourir des humanoïdes en détresse tout en évitant de dangereux prédateurs qui surgissent du sol pour vous dévorer. Dans la seconde, vous devez affronter différents périls en vous déplaçant dans une série de cavernes dont les murs se déforment. Les dessins des différents tableaux sont bons et les mouvements des humanoïdes et des prédateurs sont bien étudiés, mais le jeu présente quelques lacunes, surtout pour accéder à la seconde partie.



SPECTRUM BLACK PLANET

L'action se situe au XXI^e siècle et le joueur tient le rôle de Starmagon, officier de la patrouille galactique. La Terre a étendu son empire jusqu'aux confins de la Galaxie, mais les pirates, dont la base secrète est située sur la planète Noire, attaquent et rançonnent les vaisseaux de commerce. Pour localiser la planète Noire, le joueur doit retrouver les sept morceaux d'une clé qui sont éparpillés sur sept planètes différentes.

Le joueur doit se rendre sur chacune de ces planètes et y résoudre un problème chaque fois différent.



ANTIMATTER SPLATTER

Le puissant Traal est sur le point de détruire la Terre en lançant une attaque de bombes antimatières mortelles.

Votre seule arme dans le combat contre ces hordes est le canon à positrons qu'il faut arriver à manier très vite car les bombes tombent à une vitesse qui s'accroît avec la durée du jeu. On ne peut pas dire que le sujet innove beaucoup.

Mais il y a plus gênant : le graphique est un peu trop simpliste et les effets sonores très limités. En conclusion, un jeu sans grande originalité !



Sur l'écran

Les avantages d'un moniteur, par rapport à un simple écran de téléviseur, sont souvent déterminants pour certains travaux. Mais quel matériel convient le mieux à nos besoins ?

L'écran de visualisation est, parmi les terminaux d'ordinateur, l'un des plus essentiels. Dans la plupart des cas, il s'agit d'un poste de télévision, ou d'un simple écran noir et blanc, acheté pour l'occasion, et d'ailleurs peu coûteux. La qualité est alors acceptable, sans plus; mais elle devrait être bien supérieure.

En effet, le signal, au cours de son transfert de la mémoire à l'écran, doit subir à plusieurs reprises toute une série de codages et décodages. Quelle que puisse être la qualité des circuits, il en résulte nécessairement une perte de netteté et de lisibilité de l'image. Cette dernière devient difficile à lire au point de donner une impression désagréable de flou.

La manière d'obtenir une image de bonne qualité consiste donc à éliminer les causes de distorsion du signal, et comme celles-ci proviennent avant tout des circuits de codage et de décodage, le meilleur moyen est de les mettre hors circuit.

C'est ce que fait un moniteur. Il s'agit d'un tube cathodique identique à celui d'un téléviseur, mais dépourvu des systèmes de décodage. Plus simple, il donne des images plus fines et plus stables.

Le moniteur étant dépourvu des décodeurs, il ne peut fonctionner si on le branche sur la prise « télévision » d'un ordinateur. Il est indispensable de disposer d'une prise sortie-vidéo; celle-ci peut porter différents noms, mais il est capital qu'elle ne passe pas par le modulateur. Il suffit, pour s'en assurer, de consulter votre notice.

La manière d'obtenir une image sur un écran de télévision consiste à s'assurer que chaque événement se produit au bon moment. Le problème vient du fait que le mouvement de balayage du faisceau sur l'écran est commandé de l'intérieur du moniteur lui-même, et qu'il n'y a pas d'accès direct par l'ordinateur.

Allumé, mais non branché à une prise d'entrée, le faisceau balaie tout l'écran avec une fréquence de 50 cycles par seconde; ce qui se traduit par un éclaircissement uniforme. On obtiendra une image en allumant et en éteignant le faisceau tous les cinquantièmes de seconde, sur le même point. Toute instabilité affectant cette procédure se traduira par une perturbation de l'image, qui sera alors pénible à regarder, et pourra même devenir tout à fait insupportable si elle se poursuit. La procédure fait appel aux impulsions de synchronisation, qui, tout comme le signal de brillance, sont produites directement par l'ordinateur, et transmises au moniteur. Il existe deux sortes d'impulsions de

synchronisation : l'une concerne chaque ligne de l'image, l'autre l'image dans son ensemble. A la fin de chaque cycle, le moniteur reçoit une brève impulsion qui lui signale que toutes les lignes ont été balayées, et que le faisceau d'électrons (en fait la tache produite sur l'écran) doit revenir au coin supérieur gauche, pour décrire un nouveau cycle.

Le même événement se produit en fin de chaque ligne : l'impulsion signale qu'une nouvelle ligne doit être balayée.

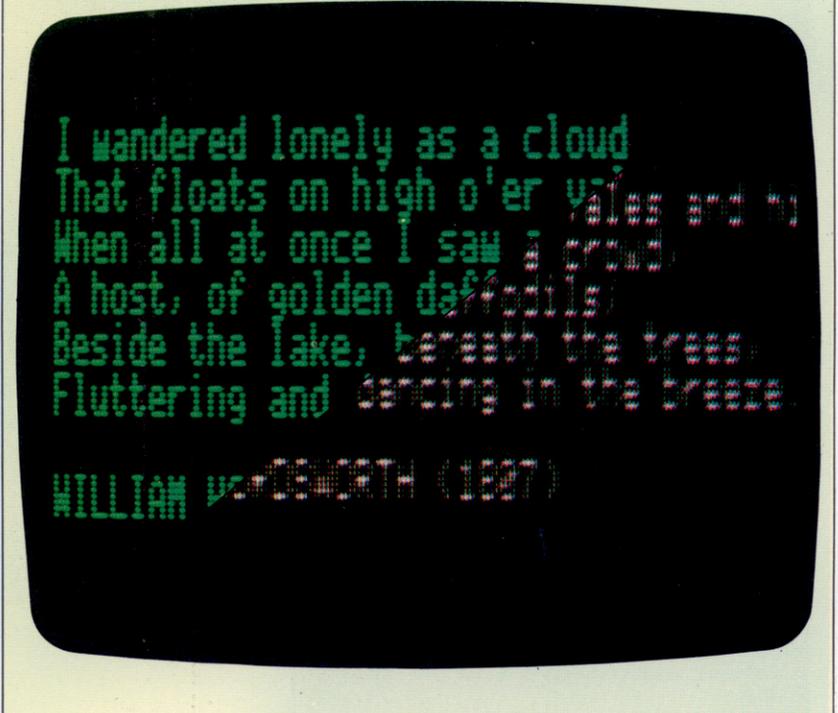
Il existe différents types de moniteurs, mais on peut en distinguer deux grandes catégories selon qu'ils sont en couleurs ou monochromes, et les classer ensuite selon les différents types de signaux qu'ils peuvent recevoir.

Les moniteurs monochromes sont fort simples, et le seul type important de signaux est dit « composite », ce qui signifie que les impulsions sont combinées avec le niveau de brillance en un seul signal « trié » par le moniteur pour former enfin l'image.

Miroir à deux faces

Cette vue montre la différence très sensible qui existe entre l'image que l'on obtient avec un moniteur monochrome

(en l'occurrence un Monitor III d'Apple) et celle que donne un poste de télévision de très bonne qualité. (Cl. Ian McKinnell.)





Les moniteurs couleur

Certains moniteurs couleur fonctionnent de manière comparable; mais en raison de la plus grande complexité du signal, ils s'apparentent davantage aux téléviseurs. Les principaux types ont pour nom PAL, SECAM et NTSC, comme les divers procédés de télévision couleur en usage de par le monde. Ils sont caractérisés par les méthodes de codage des trois couleurs « additives » (rouge, vert et bleu), et les deux types d'impulsions de synchronisation.

Les divers signaux peuvent également être transmis séparément au moniteur. Bien qu'il existe de subtiles nuances dans les procédés utilisés, on les désigne sous le sigle générique RVB, qui résulte des trois couleurs employées. Le plus simple est le TTL (transistor-transistor logic), dans lequel il n'existe, pour chaque couleur, que deux états possibles (allumé ou éteint). Le nombre de combinaisons possibles est donc 2^3 , soit 8 : ce sont les huit couleurs que l'on trouve sur les réseaux vidéotex.

Il suffit, pour obtenir une gamme de couleurs plus vaste, de moduler l'intensité de chaque teinte fondamentale. Généralement, l'intensité varie, non pas de manière continue, mais « numériquement », c'est-à-dire par sauts. Malgré cela de tels moniteurs sont dits « analogiques », ou encore « linéaires ».

Les moniteurs les moins chers ont un écran noir et blanc de 22,9 cm; leur prix est de l'ordre

Largeur de bande

L'importance de la largeur de bande (ou bande de fréquences) est capitale dans le choix d'un moniteur. Elle représente en effet la finesse d'un point sur l'écran. Dans une certaine mesure, on peut considérer qu'il y a intérêt à pouvoir obtenir les points les plus fins possibles; toutefois, ce sont des considérations d'un autre ordre qui peuvent amener à déterminer la meilleure largeur de bande convenant à un appareil donné.

Il importe avant tout de déterminer le minimum absolu de largeur de bande acceptable pour la machine. On multiplie tout d'abord le nombre de lignes que comporte l'écran par le nombre de caractères sur chaque ligne, puis ce produit par la largeur de la matrice, et enfin par la profondeur de celle-ci (mesurée en nombre de points). On obtient ainsi le nombre maximal de points qui peuvent figurer sur l'écran, nombre qui se situe généralement entre 10 000 et 1 500 000. Un écran ordinaire avec une matrice 7×9 comporte 110 960 points. Tous ces points sont illuminés par le faisceau à une fréquence de 50 cycles par seconde (60 aux États-Unis, et dans certains autres pays), ce qui multiplie d'autant le nombre de points que porte l'écran. On obtient ainsi un nombre qui se situe généralement entre 500 000 et 75 000 000 : c'est le nombre de fois par seconde que le circuit de contrôle doit pouvoir allumer et éteindre le faisceau.

Ce nombre représente une fréquence. Il s'exprime donc en hertz (Hz), et comme il est très grand, on emploie fréquemment un multiple, le mégahertz ($1 \text{ MHz} = 10^6 \text{ Hz}$). On trouve des moniteurs qui présentent des largeurs de bande de 5, 7, 10, 12, 15 et jusqu'à 20 MHz. Le prix croît très vite avec la fréquence. On peut réaliser des moniteurs ayant une largeur de bande de 100 MHz, mais leur prix de revient est extrêmement élevé.

D'une manière générale, plus la largeur de bande d'un moniteur est élevée, plus l'image est claire. Toutefois, une trop grande finesse de résolution peut présenter des inconvénients, notamment dans le cas des figures en couleurs.



Moniteurs monochromes
Ce sont les plus simples et les moins coûteux. Pour une somme de l'ordre de 1 000 F, ils offrent à l'utilisateur un choix de couleur « phosphore » et une définition acceptable. Ils sont bien adaptés au traitement des textes et aux usages commerciaux. On peut également les utiliser pour la représentation de figures simples.

Moniteurs à trois couleurs

Le CUB de Microvitec constitue un très bon exemple de moniteur couleur peu coûteux, à l'usage des personnes disposant d'un ordinateur domestique. Si son pouvoir de résolution reste moyen, il est adapté à divers usages, et notamment à la production d'images.

Polyvalence japonaise

Capable d'analyser le signal d'entrée, le TM90PSN de JVC est particulièrement destiné à ceux qui souhaitent employer leur ordinateur à plusieurs tâches différentes. Il est utilisable non seulement comme terminal d'ordinateur, mais également associé à un enregistreur vidéo.

INPUT SELECT 1 2



Téléviseurs domestiques

La qualité des images fournies par les téléviseurs domestiques s'est rapidement améliorée avec les progrès réalisés dans la fabrication des tubes cathodiques et des canons à électrons. Certains fabricants sont pleinement conscients du double usage qu'on peut faire des téléviseurs, en montant la prise d'antenne sur l'avant des postes.



POWER
ON OFF

PRO VIDEO MONITOR

AUTO

PAL

SECAM

NTSC
(3.58)

NTSC
(4.43)

INPUT 1

INPUT 2

R G B

MAIN
POWER

de 900 F. Un moniteur de 30,5 cm, à écran vert, coûte de 1 500 à 2 000 F. Les moniteurs couleur sont plus coûteux, en raison du prix de revient plus élevé du tube. Les appareils standard ont un prix qui oscille entre 2 500 et 4 000 F, selon les dimensions de l'écran et de la bande de fréquences. La plupart des moniteurs couleur sont disponibles suivant deux versions, TTL et analogique. Mais il faut remarquer que dans les deux cas les prix sont voisins.

Le TM90PSN est un moniteur intéressant, très souple d'emploi, que produit la société JVC. Son prix est proche de celui d'un téléviseur couleur, et son encombrement est réduit, avec un écran de 25,4 cm.

La bande de fréquences n'est pas très élevée (ce qui limite le pouvoir de résolution), mais ce moniteur est adaptable à pratiquement tous les types d'entrée, depuis le composite monochrome ordinaire, en passant par le TTL et le RVB analogique, jusqu'aux signaux composites couleur PAL, SECAM, NTSC 3.58, ou NTSC 4.43. Il est pourvu d'un mécanisme de sélection qui vérifie le signal d'entrée et enclenche automatiquement le moniteur sur le mode approprié.

L'usage de ce type de moniteur est appelé à se généraliser, et on pourra trouver là un terminal polyvalent pour ordinateurs, enregistreurs vidéo, et appareils à vidéodisques notamment.

Persistence et couleur

Un critère important, dans le choix d'un moniteur monochrome, est le type de « phosphores » employé dans la fabrication de l'écran. Cette substance sous forme poudreuse en revêt la surface interne. Elle possède la propriété de devenir lumineuse sous l'impact d'un faisceau d'électrons. C'est ainsi que se forme l'image.

Les principales propriétés à prendre en considération sont la couleur et la « persistance ». Cette dernière caractérise le temps pendant lequel le phosphore reste lumineux après l'extinction du faisceau. Sa valeur exacte est rarement indiquée, et les constructeurs se contentent généralement de préciser « longue » ou « courte ». Les phosphores employés pour la réalisation des téléviseurs et de quasiment tous les moniteurs couleur ont une persistance courte, de même que la majorité des moniteurs monochromes, mais dans la plupart des cas d'utilisation, il y a intérêt à ce qu'une image subsiste pendant une fraction de seconde, ce qui supprime le clignotement entraînant une fatigue oculaire.

Le meilleur exemple de phosphore à persistance longue est fourni par les écrans de radar, où le balayage radial laisse de longues traînées lumineuses qui confèrent pour ainsi dire à l'écran l'équivalent d'une mémoire.

Toutefois, si le système comporte un crayon optique son phosphore sera généralement à persistance longue. Normalement, l'image doit se dissiper assez vite pour que l'ordinateur puisse localiser le crayon. Mais si le phosphore demeure lumineux, l'ordinateur notera la présence de lumière, sans prendre en compte la position du crayon et sans noter l'orientation du faisceau vers le point considéré.

Divers types et couleurs de phosphore sont employés selon l'usage auquel un moniteur est destiné. Pratiquement toutes les couleurs sont réalisables. Le phosphore blanc à persistance courte est bon marché et facile à obtenir, mais l'un des tons de vert est presque aussi courant et beaucoup plus agréable à l'œil (de même pour la couleur ambre). On trouve du bleu sur bon nombre de terminaux, notamment ceux qu'emploient les agences de voyages et les compagnies aériennes.

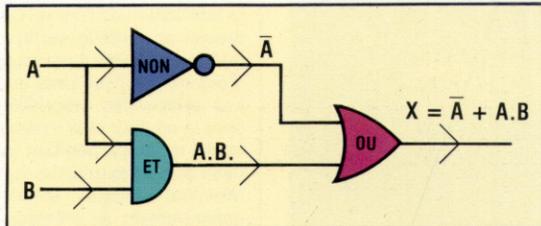
Blocs logiques

Dans le premier cours de cette série, nous avons vu trois blocs logiques ET, OU et NON. Nous étudions maintenant comment ces circuits effectuent la fonction d'addition.

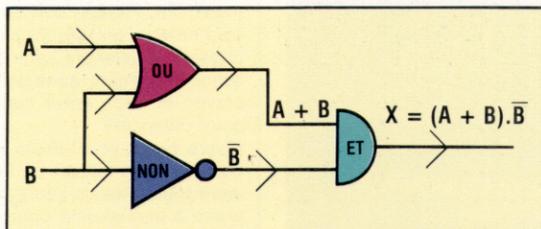
L'utilisation de la notation algébrique pour décrire des relations logiques est appelée *algèbre booléenne*, du nom du célèbre mathématicien George Boole (1815-1864). Cette forme d'algèbre est très utilisée dans les circuits logiques, parce qu'elle permet de les simplifier mathématiquement. Ce qui signifie moins de portes logiques pour effectuer une fonction et un accroissement de la vitesse d'exécution. Toute la base de l'architecture des ordinateurs dépend de cette logique du vrai et du faux. Votre propre micro-ordinateur utilise des circuits logiques découlant de l'algèbre de Boole.

Règles de priorité

Nous avons déjà rencontré la notation booléenne pour la sortie des trois portes logiques BASIC, ET ($A.B$), OU ($A + B$) et NON (\bar{A}). Ces expressions permettent d'exprimer des circuits plus complexes. Ainsi l'expression booléenne $X = \bar{A} + A.B$ peut se représenter par le circuit suivant :



Il est important de noter que l'ordre dans lequel les opérations ET et OU se déroulent n'est pas indifférent. La règle veut que ET ait la priorité d'exécution sur OU (ainsi que sur NON). Si l'on désire renverser l'ordre des priorités, il faut utiliser des parenthèses. Ainsi : $X = (A + B). \bar{B}$. L'opération ici suit l'ordre : OU, ET avec la valeur négative de B. Voici le diagramme du circuit logique :



Lorsque vous devez tracer un circuit logique à partir de son expression booléenne, il est préférable de partir de sa sortie (le résultat logique) et de remonter jusqu'aux entrées.

Ou inclusif et exclusif

OU peut signifier :

L'un OU l'autre OU les deux.

Mais OU peut également signifier (OU exclusif) :

L'un OU l'autre mais pas les deux.

Par exemple, pour participer à une course de véhicules à deux roues, on peut posséder un vélo OU une trottinette (pour une course très spéciale), mais on peut aussi posséder les deux. Il s'agit là du OU *inclusif*. Par contre, un individu peut être décrit comme grand OU comme petit, mais pas les deux. Le OU *exclusif* exclut la possibilité de vérifier les deux assertions.

En logique, le OU *exclusif* est très utile. Il s'agit de XOU. XOU peut être obtenu à partir des portes logiques ET, OU et NON. La table de vérité pour XOU est la suivante :

ENTRÉES SORTIES

A	B	$A \nabla B$
0	0	0
0	1	1
1	0	1
1	1	0

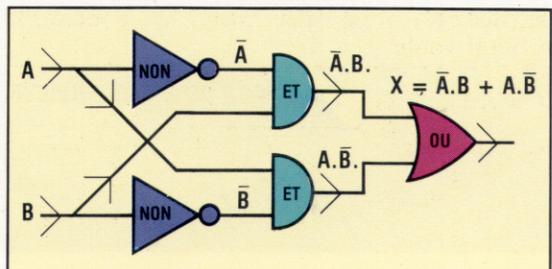
Ainsi que le montrent la deuxième et la troisième ligne, une valeur de 1 peut être obtenue ainsi :

$$\begin{aligned} & \text{NON}(A) \text{ ET } B \\ & \text{OU} \\ & A \text{ ET NON}(B) \end{aligned}$$

Cela s'écrit sous forme booléenne :

$$X = \bar{A}.B + A.\bar{B}$$

Un circuit possible pour XOU à partir de l'expression booléenne ci-dessus serait :



Ce qui réaliserait un circuit à cinq portes logiques. Nous verrons plus loin comment simplifier ce circuit.

Portes logiques et arithmétique

Bien que la plupart des ordinateurs domestiques soient capables d'effectuer la totalité des fonctions arithmétiques, seule l'addition est directement tributaire des circuits logiques. Toutes les autres fonctions, comme la soustraction, la multiplication et la division, sont prises en charge par un système utilisant conjointement des circuits logiques *additionneurs* et du logiciel servant au contrôle de la disposition des bits. Soit l'addition binaire suivante :

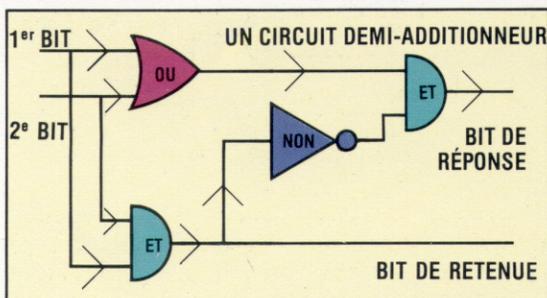
	8s	4s	2s	1s
	0	1	0	1
	0	1	1	1
	1	1	0	0
RETENUES	1	1	1	

Si nous considérons de manière isolée une colonne de l'addition, par exemple la colonne des 2, nous pouvons lister les entrées et les sorties afférentes. Les entrées sont : les deux bits à ajouter et le bit de retenue provenant de la colonne précédente. Les sorties sont : le bit mis dans la colonne des 2 en tant que réponse à l'opération, et le bit de retenue à reporter à la colonne suivante. Le système qui prend les données en entrée et produit en sortie les données appropriées s'appelle un *additionneur*. Il est assez complexe, aussi nous commencerons par une version simplifiée, le *demi-additionneur*. Soit le circuit demi-additionneur :

ENTRÉES SORTIES

1 ^{er} bit	2 ^e bit	bit de retenue	bit de réponse
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Il est facile de constater que le bit de retenue est égal à 1 lorsque le 1^{er} bit ET le 2^e sont tous les deux égaux à 1. Le *bit de réponse* est obtenu en soumettant les entrées à OU, sauf dans le cas où le bit de retenue est 1. La condition pour que le bit de réponse soit 1 est : « Le premier bit est 1, OU le deuxième bit est 1, et NON 1 si le bit de retenue est 1. » Le circuit suivant fera obtenir le résultat voulu :



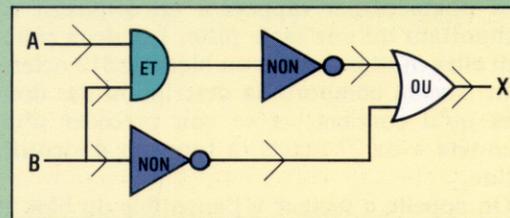
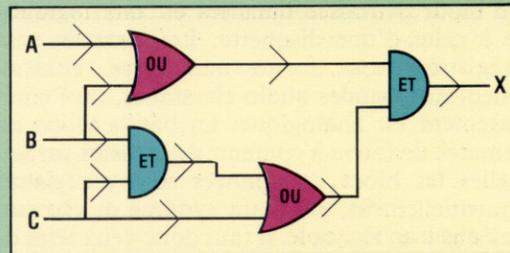
Exercice 2

1) Tracez les circuits logiques correspondant aux expressions booléennes suivantes :

- a) $X = (A + B).C$
- b) $X = A.B + (A + C)$
- c) $X = \bar{A}.B + (A + B)$
- d) $X = \bar{A}.B.(A + B)$

2) Écrivez l'expression booléenne avec A et B respectivement comme données en entrée pour le bit de retenue et le bit de réponse d'un circuit demi-additionneur.

3) Écrivez les expressions booléennes pour les circuits suivants :



Réponses au premier exercice.

1)

A PASSÉ SON PERMIS	ACCOMPAGNÉ	PERMIS OBTENU
FAUX	FAUX	FAUX
FAUX	VRAI	VRAI
VRAI	FAUX	VRAI
VRAI	VRAI	VRAI

2)

AVEC CASSETTE DE JEUX	AVEC LECTEUR	POUR UN AUTRE MICRO	LE PROGRAMME CHARGERÀ
FAUX	FAUX	FAUX	FAUX
FAUX	FAUX	VRAI	FAUX
FAUX	VRAI	FAUX	VRAI
FAUX	VRAI	VRAI	FAUX
VRAI	FAUX	FAUX	VRAI
VRAI	FAUX	VRAI	FAUX
VRAI	VRAI	FAUX	VRAI
VRAI	VRAI	VRAI	FAUX

3)

A	B	P	Q	C
0	0	1	0	0
0	1	1	1	1
1	0	0	1	0
1	1	0	1	0



Une boucle sans fin

Le Spectrum est le résultat d'une idée chère à sir Clive Sinclair : être à la fois puissant et bon marché. Mais il manquait d'une mémoire de masse et d'accès rapide, jusqu'à l'apparition des microdrives.

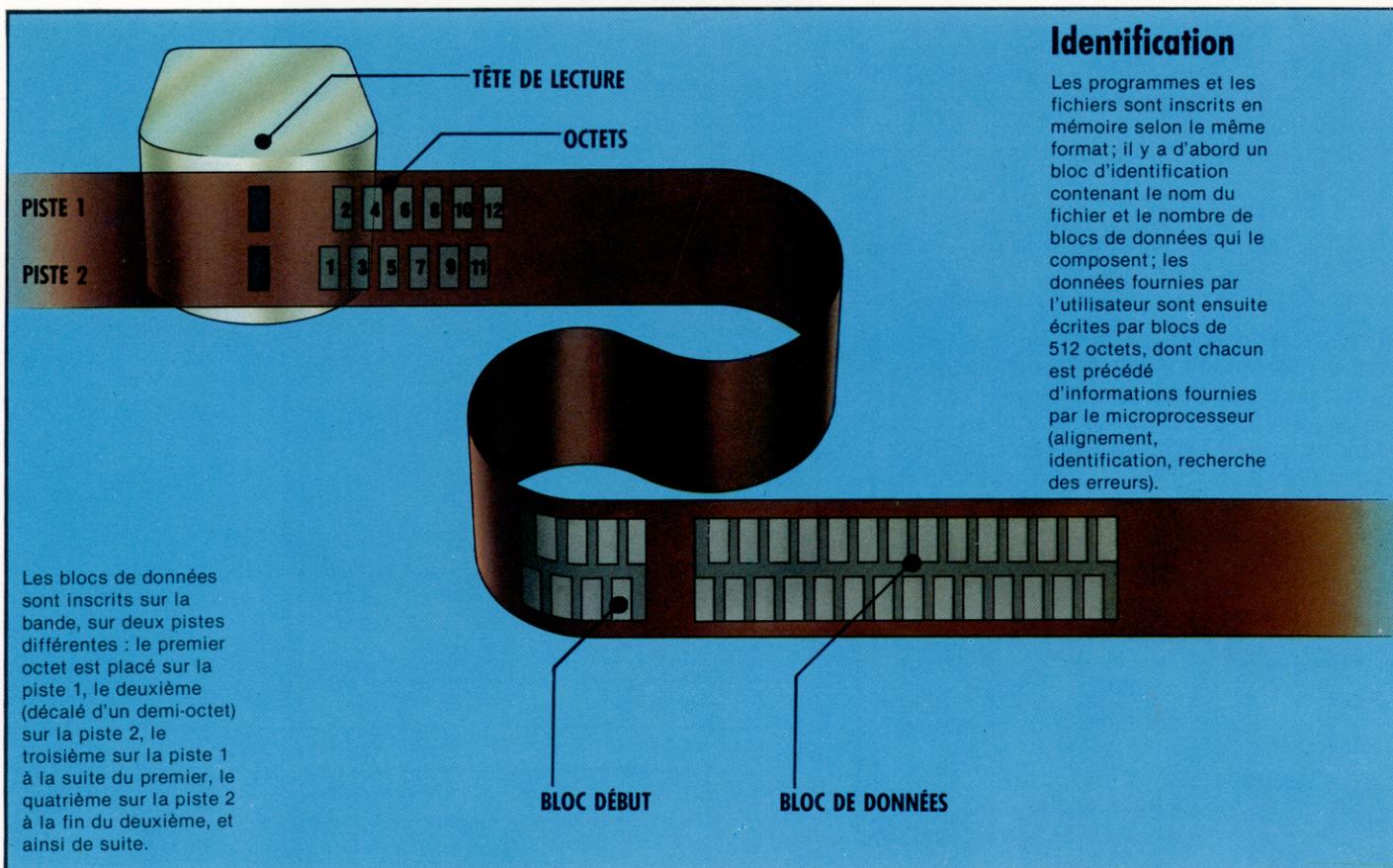
Un microdrive utilise une bande magnétique vidéo sans fin, de 2 mm de large et de 8 m de long. Elle tourne sur elle-même toutes les sept secondes, à l'intérieur d'une petite cartouche. Son mode de fonctionnement est assez semblable à celui d'une disquette. Les données sont enregistrées sous forme numérique, contrairement aux bandes audio classiques, où l'enregistrement est analogique. La bande vidéo est formatée de façon à contenir deux pistes sur lesquelles les blocs de données sont enregistrés séquentiellement, selon un système de chevauchement « en zigzag ». Il faut donc deux têtes de lecture. Mais la densité et la vitesse du stockage sont doublées par rapport à un dispositif ne comportant qu'une seule piste. Les deux pistes sont elles-mêmes divisées en blocs de 512 octets, dont chacun comporte la description des données qu'il contient, et se voit précéder d'un « en-tête » de 27 octets, à fonction d'identification.

On appelle « secteur » l'ensemble du bloc et de son en-tête; vu sa longueur, la bande peut en

accueillir près de 200, ce qui donne une densité d'enregistrement de près de 200 octets par centimètre. Un fichier, identifié par un nom, peut être stocké dans un seul secteur s'il occupe moins de 512 octets. S'il déborde sur un ou plusieurs secteurs, l'espace restant n'est pas utilisé jusqu'à effacement de la bande. C'est pourquoi la capacité mémoire théorique de 100 K se trouve en fait réduite à 85 ou 90 K. Le temps moyen nécessaire pour retrouver un programme et le charger en mémoire est compris entre 10 et 15 secondes.

L'interface ZX

Le microdrive est connecté au Spectrum par l'intermédiaire d'une interface ZX 1, qui se fixe au connecteur plat de l'appareil. On peut mettre en parallèle jusqu'à huit microdrives, selon un arrangement « en guirlande ». La ZX 1 sert aussi d'interface standard RS232, d'interface pour réseau local, et se raccorde sur l'imprimante ZX. Elle comporte aussi en ROM des ins-



Identification

Les programmes et les fichiers sont inscrits en mémoire selon le même format; il y a d'abord un bloc d'identification contenant le nom du fichier et le nombre de blocs de données qui le composent; les données fournies par l'utilisateur sont ensuite écrites par blocs de 512 octets, dont chacun est précédé d'informations fournies par le microprocesseur (alignement, identification, recherche des erreurs).

Les blocs de données sont inscrits sur la bande, sur deux pistes différentes : le premier octet est placé sur la piste 1, le deuxième (décalé d'un demi-octet) sur la piste 2, le troisième sur la piste 1 à la suite du premier, le quatrième sur la piste 2 à la fin du deuxième, et ainsi de suite.



tructions qui viennent s'ajouter au BASIC du Spectrum, et permettent une manipulation plus aisée des données, que ce soit avec le microdrive ou d'autres interfaces.

L'ensemble (microdrive et interface 1) accroît considérablement les possibilités du Spectrum, et se comporte comme un lecteur de disquettes conventionnel. Mais la fiabilité à long terme des cartouches reste sujette à caution, et le système est dépourvu de véritables fichiers à accès sélectif. C'est un gros handicap dans le domaine de la gestion, où l'on a sans cesse besoin de situer, de mettre en mémoire et de rappeler des petits groupes de données. Par ailleurs, assez peu de logiciels ont été commercialisés sous forme de cartouches; mais il est vrai que la situation devrait rapidement changer... à condition que celui-ci tienne ses promesses et se révèle pleinement fiable.



Les commandes du microdrive

Elles sont les suivantes :

FORMAT CAT SAVE* VERIFY* LOAD*
MERGE* ERASE OPEN# PRINT# INPUT#
INKEY\$# CLOSE# MOVE

Dans chaque cas, M permet le choix du microdrive, N est le numéro (entre 1 et 8) du microdrive particulier auquel on veut s'adresser, et S (entre 4 et 15) est le numéro du canal de sortie des données.

Format

Formate la bande contenant les données, attribue un nom à la cartouche et efface toutes les données précédemment en mémoire. On s'y prend de deux façons :

FORMAT «M»;N;«NOM» ou
FORMAT M\$;N;C\$

NOM est le nom choisi pour la cartouche (10 lettres maximum), M\$ (que ce soit M ou m) remplace M, et C\$ (10 caractères maximum) a le même rôle que NOM.

Cat

Charge vers l'écran (ou vers tout canal de sortie particulier) un catalogue de tous les fichiers contenus sur une cartouche placée dans un microdrive défini. La méthode est

CAT N ou
CAT #;N

Un catalogue contient le nom de la cartouche, un nombre maximal de 50 fichiers, et mentionne, en octets, l'espace encore disponible sur la cartouche.

Save

Crée un fichier de programmes qui peuvent comporter des programmes proprement dits, des chaînes ou des données ayant reçu un nom. Plusieurs possibilités :

1. SAVE*«M»;N;«NOM DE FICHIER»
2. SAVE*«M»;N;«NOM DE FICHIER» SCREEN\$
3. SAVE*«M»;N;«NOM DE FICHIER»DATA A#
4. SAVE*«M»;N;«NOM DE FICHIER»LINE X

avec les résultats suivants :

1. un fichier;
2. un fichier écran (SCREEN\$);
3. un fichier contenant les données A#;
4. un fichier qui commencera (RUN) à charger le programme (LOAD) à partir de la ligne X.

Verify*, Merge* et Erase

Ces commandes sont analogues à la première de SAVE*. VERIFY* compare un «NOM DE FICHIER» avec le contenu de la mémoire utilisateur, et fait apparaître un message d'erreur s'ils ne concordent pas. MERGE* fait fusionner un «NOM DE FICHIER» avec le contenu de la mémoire, et ERASE* efface «NOM DE FICHIER».

Load*

S'obtient de la même manière que «1. SAVE*» et «2. SAVE*». Sa fonction est de faire passer en mémoire le contenu du fichier qui lui a été spécifié.

Open#, Print#, Input#, Inkey\$, Close# et Move

Toutes ces commandes sont en relation avec la manipulation des fichiers de données. Ceux-ci sont placés en mémoire de façon séquentielle, mais on peut les traiter de telle sorte qu'on imite certaines particularités des fichiers à accès sélectif : un fichier spécial est lu en entier, et les données nécessaires sont extraites après chargement. De même, une fois logé dans la mémoire de l'ordinateur, il peut être modifié, puis stocké de nouveau. On se sert d'OPEN et de CLOSE pour manipuler des flux de données, en faisant référence à un canal de sorties particulier. Ainsi

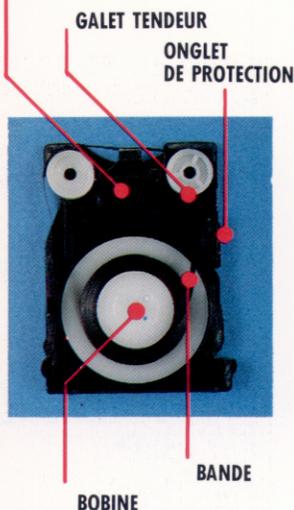
OPEN#S;«M»;N;«NOM DE FICHIER»

connecte le canal de sortie S au programme «NOM DE FICHIER», contenu dans le microdrive N. Ce programme peut alors être inscrit en utilisant PRINT#S, et lu en se servant de INPUT#S ou encore INKEY#S. MOVE permet de transférer des fichiers de données à l'intérieur d'une cartouche, d'un microdrive à l'autre, ou en direction de n'importe quel dispositif accessible par un canal de sortie. Lorsque ce dernier a rempli son rôle, il est désactivé par la commande CLOSE#S.

Bande souple

Le microdrive possède à la fois la simplicité d'usage et le prix raisonnable d'une mémoire sur cassette, ainsi que la rapidité d'une disquette. L'interface 1 permet de le connecter au Spectrum : elle dispose d'un port RS232 et de prises phono permettant le raccordement sur cassette. (Cl. Ian McKinnell.)

PATIN PRESSEUR



Le mécanisme du microdrive

Une bande vidéo de 2 mm de large est utilisée en raison de sa solidité et de sa haute capacité de stockage. Elle forme une boucle de 8 m de long. Elle est pourvue d'un revêtement spécial pour glisser sans problèmes à l'intérieur de la cartouche. La tête de lecture la maintient contre le patin presseur. Un onglet de protection permet d'y lire ou d'y inscrire des données.

Mémoire paginée

La compréhension des bases de la programmation en langage machine consiste à examiner la façon dont les ordinateurs organisent et gèrent leur mémoire.

Dans la première partie de ce cours, nous avons pris l'exemple d'une usine où chaque employé a sa propre configuration d'interrupteurs qui allument quatre ampoules dans le bureau du directeur. Ce dernier sait ainsi qui est à son poste. On a donc montré comment l'information (le nom de la personne qui travaille) pouvait être représentée par le passage d'électricité.

Dans notre exemple, avec quatre interrupteurs et quatre ampoules on pouvait représenter les nombres compris entre 0 et 15. Autrement dit, il n'y avait que 16 configurations possibles. Toutefois, si nous avons utilisé huit interrupteurs et ampoules au lieu de quatre, nous aurions obtenu 256 configurations distinctes ($2 \times 2 = 256$) et nous aurions donc pu compter de 0 à 255.

Dans votre ordinateur, la mémoire est disposée en blocs individuels de huit interrupteurs, et chacun de ces blocs est appelé octet. En général, l'UC traite 1 octet à la fois, ce qui implique qu'elle peut seulement additionner, comparer et stocker des nombres compris entre 0 et 255. Cela paraît limiter ses capacités arithmétiques, mais ce n'est pas le cas. En considérant l'addition $63951 + 48770 = ?$ vous verrez qu'en fait vous manipulez les chiffres l'un après l'autre. De même, l'UC peut effectuer des opérations sur des grands nombres en utilisant 1 octet à la fois.

Comme il comprend huit interrupteurs, 1 octet peut stocker un nombre binaire à huit chiffres. Chacune des positions de chiffres binaires est appelée bit — la plus petite unité possible d'information. Dans 1 octet, 1 bit est soit ouvert, soit fermé, et un chiffre binaire est soit 1, soit 0.

Question d'adresse

Il importe souvent de parler de bits individuels dans 1 octet, aussi convient-on de numéroter les bits de 0 à 7 à partir de la droite de l'octet. Si un octet contient le nombre binaire 00000001, nous disons que le bit 0 est à 1 ou qu'il est ouvert. Tous les autres bits sont à zéro ou fermés. Ainsi dans le nombre binaire 01001000, le bit 3 est ouvert, le bit 6 est ouvert, le bit 4 est fermé, le bit 7 est à zéro, le bit 0 aussi, etc. Dans 1 octet, le bit 0 est également appelé le bit le moins significatif (LSB) et le bit 7 le plus significatif (MSB).

Une mémoire d'ordinateur peut être représentée par une longue bande de papier quadrillé, large de huit carreaux et longue de plu-

sieurs milliers de carreaux : chaque rangée de huit carreaux est 1 octet, chaque carreau 1 bit. La mémoire ne sert à rien si l'on ne peut pas y localiser les articles, de sorte que chacun des octets est identifié par un label appelé adresse; celle-ci n'est écrite nulle part sur le papier (ou dans l'octet), c'est simplement le numéro de l'octet dans la mémoire, en comptant à partir du début de la mémoire. Le premier octet a donc l'adresse 0, le suivant l'adresse 1 et ainsi de suite. Si vous voulez écrire quelque chose dans l'octet 43, vous comptez à partir de l'octet 0 jusqu'au 43.

Lorsque vous l'aurez atteint, rien n'identifiera le numéro de cet octet, sauf sa position — vous avez commencé à compter à partir de l'octet zéro, vous avez atteint le nombre 43, donc cela doit être l'octet 43. En fait, les octets de mémoire sont de minuscules blocs de huit transistors (un par bit, huit par octet) gravés dans les puces de votre machine, et qui ne diffèrent que par leur position physique car, pour le reste, ils sont parfaitement ressemblants entre eux.

Ce système d'adressage serait parfait s'il n'y avait que quelques centaines d'octets. L'UC peut compter jusqu'à 100 en une fraction de milliseconde; mais les ordinateurs ont des milliers d'octets, et il faudra un certain temps, même à un microprocesseur, pour compter de 0 à 20 000. L'ordinateur surmonte ce problème en divisant la mémoire en pages, exactement comme dans le cas d'un livre.

En considérant toujours la mémoire d'ordinateur comme une bande de papier quadrillé de huit carreaux de largeur et de milliers de carreaux de longueur, nous pouvons imaginer de couper cette bande tous les 100 octets (c'est-à-dire couper entre l'octet 99 et l'octet 100, entre le 199 et le 200, etc.). Chacune des bandes de papier comprise entre deux coupures est maintenant une page de 100 octets. La page 0 commence à l'octet 0 et se termine à l'octet 99; la page 1 va de l'octet 100 à l'octet 199, etc. A présent, pour trouver un octet, par exemple l'octet 3518, il n'est plus nécessaire de compter 3 518 octets à partir du commencement de la mémoire. Nous n'avons qu'à compter 35 pages à partir du commencement, puis les octets à partir du début de la page jusqu'à 18 : nous aurons atteint l'octet 3518. Essayez avec une bande de papier quadrillé si vous n'avez pas compris.

Ce système de mémoire paginée est pratique car il suffit de regarder l'adresse d'un octet et de



la partager en deux — les chiffres à gauche de la colonne des centaines représentent le numéro de page de l'octet, alors que les deux premiers chiffres à droite (dizaines et unités) donnent le nombre d'octets comptés à partir du bas de la page. Dans l'exemple précédent, nous avons partagé l'adresse 3518 en deux nombres : le numéro de page 35, et le numéro d'octet 18 sur cette page. On dit que 18 est *décalé*, ou *décalé de page*, car c'est le nombre par lequel il faut décaler (ou augmenter) l'adresse de l'octet du bas de la page pour atteindre l'octet en question.

Néanmoins, l'ordinateur ne compte pas en décimal, comme nous, mais en binaire. Le système de pagination dépend de la capacité à trouver la page et le décalé au premier coup d'œil à l'adresse de l'octet. L'adresse décimale 99 est représentée par 01100011 en binaire, et 100 par 01100100; 199 décimal est égal à 11000111 et 200 décimal à 11001000 en binaire. Ces exemples nous montrent qu'il n'est pas évident de trouver les numéros de pages à partir des adresses en binaire. La raison en est le choix de la taille des pages.

Taille des pages

Nous avons précédemment choisi 100 justement parce que c'est un nombre significatif dans le système décimal (c'est une puissance de 10). Si nous comptons en binaire, nous devons choisir un numéro de page convenant à ce système. La taille de page utilisée par nos ordinateurs est 256, de sorte que la page 0 commence par l'octet 0 et se termine par l'octet 255; la page 1 va de l'octet 256 à l'octet 511, et ainsi de suite. Pour comprendre l'avantage de cette pagination, écrivons ces adresses en binaire :

Page 0 : octet 00000000 – octet 11111111
Page 1 : octet 100000000 – octet 111111111

Comme vous voyez, on peut compter en binaire de 0 à 255 avec un nombre de 8 bits; le nombre suivant — 256 — nécessite 9 bits, et avec 9 bits on peut compter jusqu'à 511, etc. Nous voyons maintenant que si la taille d'une page est 256 et si on compte en binaire, le décalé comprend les 8 bits de droite, et le numéro de page est donné par les bits à gauche du bit 7.

Cela peut poser des problèmes, puisque nous avons vu que l'UC ne peut traiter que des octets, et 1 octet ne contient que 8 bits. On peut donc se demander à quoi bon parler de nombres de 9 ou 10 bits. La réponse est que toutes les adresses de mémoire sont traitées comme des nombres de deux octets, que l'UC traite 1 octet à la fois. Si nous récrivons les limites de pages en nombres à 2 octets, ce système devient plus clair :

la page 0 commence par 00000000 00000000
et se termine par 00000000 11111111
la page 1 commence par 00000001 00000000
et se termine par 00000001 11111111
la page 10 commence par 00000010 00000000
et se termine par 00000010 11111111

la page 11 commence par 00000011 00000000
et se termine par 00000011 11111111

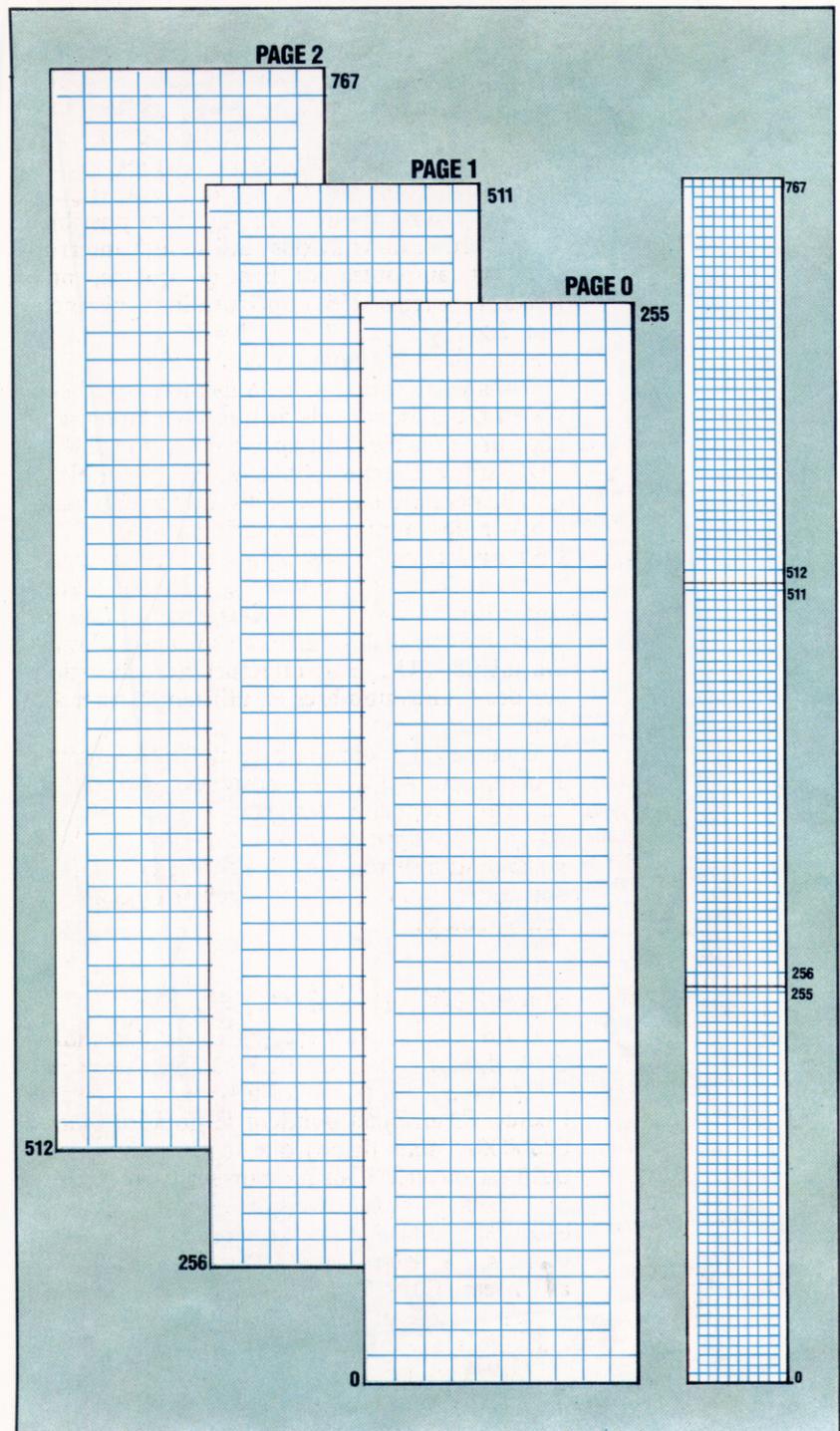
et ainsi de suite.

Nous voyons bien maintenant que lorsque l'UC cherche ou met une information dans un octet de mémoire, cet octet sera identifié par une adresse à 2 octets. Le premier, le plus à gauche, donne le numéro de page, tandis que le second, le plus à droite, donne le décalé.

Plus loin, nous donnons des programmes pour convertir des nombres décimaux en binaires ainsi qu'en hexadécimaux. Ces derniers sont largement utilisés en code machine, et nous en discuterons en détail dans la suite de ce cours.

L'adressage paginé

L'adressage paginé divise la mémoire en blocs ou pages imaginaires de 256 octets. Toutes les adresses sont alors exprimées comme des nombres à 2 octets : le premier donne le numéro de page, le second le décalé à partir du commencement de cette page.





Croqueur de nombres

Les trois programmes présentés ici, pour les micro-ordinateurs BBC, Spectrum et Commodore, acceptent des nombres décimaux et restituent leurs équivalents binaires.

Commodore 64

```

10 REM ***** COMMODORE *****
40 S$ = " = " : X$ = "0123456789ABCDEF"
50 REM S$ CONTIENT 9 ESPACES
60 PRINT CRH$(147) : REM EFFACE L'ECRAN
70 PRINT " POUR AFFICHER LES NOMBRES
  DECIMAUX"
80 PRINT " ET LEURS EQUIVALENTS
  BINAIRES"
90 PRINT:PRINT " ***** TAPEZ 0 POUR
  ARRETER ***** ":PRINT
100 FOR K = 1 TO 1
110 FOR L = 1 TO 1
120 INPUT "TAPEZ UN NOMBRE ENTIER POSITIF ";
  A$
130 NU = VAL(A$)
140 IF NU = 0 THEN PRINT "FIN DE PROGRAMME":
  STOP
150 IF INT(NU) <> ABS(NU) THEN LET L = 0
160 IF NU > 65535 THEN PRINT NU: " EST TROP
  GRAND": L = 0
170 NEXT L
200 NM = NU: H$ = " ": GOSUB 2000
210 PRINT NU: TAB(5): N$:
220 IF RIGHT$(A$, 1) = " + " THEN GOSUB 4000
230 PRINT H$: PRINT: PRINT
240 K = 0: NEXT K
250 END
300 END

1000 REM ***** S-P OCTET BINAIRE *****
1010 B$ = ""
1020 FOR D = 8 TO 1 STEP -1
1030 N1 = INT(N/2)
1040 R = N - 2*N1
1050 B$ = MID$(STR$(R), 2) + B$
1060 N = N1
1070 NEXT D
1080 RETURN
2000 REM ***** S-P CONVERSION EN BINAIRE *****
2010 IF NM < 256 THEN N = NM: GOSUB 1000: N$ = S$ + B$
  : RETURN
2020 HI = INT(NM/256): LO = NM - 256*HI
2030 N = HI: GOSUB 1000: N$ = " " + B$
2040 N = LO: GOSUB 1000: N$ = N$ + " " + B$
2050 RETURN
3000 REM ***** S-P CONVERSION HEX *****
3010 HB = INT(N/16): LB = N - HB*16
3020 B$ = MID$(X$, HB + 1, 1) + MID$(X$, LB + 1, 1)
3030 RETURN
4000 REM ***** S-P CONVERSION REX *****
4010 IF NM > 256 THEN N = NM: GOSUB 3000: H$ =
  " " + B$: RETURN
4020 HI = INT(NM/256): LO = NM - 256*HI
4030 N = HI: GOSUB 3000: H$ = " " + B$
4040 N = LO: GOSUB 3000: H$ = H$ + " " + B$
4050 RETURN

```

BBC Micro

Même programme que pour le Commodore, mais avec les changements suivants :

```

60 CLS: X$ = "
210 PRINT TAB(0): NU: TAB(5): N$:
1050 B$ = STR$(R) + B$

```

Ce programme n'utilise pas les possibilités de représentation des nombres du BBC pour des raisons de compatibilité avec les autres machines; on pourrait le récrire sous une forme plus concise.

Spectrum

```

10 REM ***** SPECTRUM *****
40 LET S$ = " = " : LET X$
  = "0123456789ABCDEF"
50 REM S$ CONTIENT 9 ESPACES
60 CLS
70 PRINT " POUR AFFICHER LES NOMBRES
  DECIMAUX"
80 PRINT " ET LEURS EQUIVALENTS
  BINAIRES"
90 PRINT:PRINT " ***** TAPEZ 0 POUR
  ARRETER ***** ":PRINT
100 FOR K = 1 TO 1
110 FOR L = 1 TO 1
120 INPUT "TAPEZ UN NOMBRE
  ENTIER POSITIF": A$
130 LET NU = VAL(A$)
140 IF NU = 0 THEN PRINT
  "FIN DE PROGRAMME": STOP
150 IF INT(NU) <> ABS(NU) THEN LET
  L = 0
160 IF NU > 65535 THEN PRINT NU: "
  EST TROP GRAND": LET L = 0
170 NEXT L
200 LET NM = NU: LET H$ = " ":
  GOSUB 2000
210 PRINT NU: TAB(5): N$:
220 IF A$(LEN A$) = " + "
  THEN GOSUB 4000
230 PRINT H$: PRINT: PRINT
240 LET K = 0: NEXT K
300 END

1000 REM *** S-P OCTET BINAIRE ***
1010 LET B$ = ""
1020 FOR D = 8 TO 1 STEP -1
1030 LET N1 = INT(N/2)
1040 LET R = N - 2*N1
1050 LET B$ = STR$(R) + B$
1060 LET N = N1
1070 NEXT D
1080 RETURN
2000 REM *** S-P CONVERSION BIN ***
2010 IF NM > 256 THEN LET N
  = NM: GOSUB 1000: LET N$ = S$ + B$:
  RETURN
2020 LET HI = INT(NM/256): LET LO
  = NM - 256*HI
2030 LET N = HI: GOSUB 1000: LET N$
  = " " + B$
2040 LET N = LO: GOSUB 1000: LET N$
  = N$ + " " + B$
2050 RETURN
3000 REM ***** S-P OCTET HEX *****
3010 LET HB = INT(N/16): LET LB
  = N - HB*16
3020 LET B$ + X$(HB + 1) + X$(LB + 1)
3030 RETURN
4000 REM ***** S-P CONVERS HEX *****
4010 IF NM > 256 THEN LET N = NM: GOSUB
  3000: LET H$ = " " + B$: RETURN
4020 LET HI = INT(NM/256): LET LO
  = NM - 256*HI
4030 LET N = HI: GOSUB 3000: LET H$ =
  " " + B$
4040 LET N = LO: GOSUB 3000: LET H$
  = H$ + " " + B$
4050 RETURN

```

Si vous entrez un nombre se terminant par un " + " par exemple 6435 + , alors, outre ses représentations décimale et binaire, sa représentation hexadécimale sera aussi affichée.



Lève-toi et joue

Lorsque Nolan Bushnell connecta un microprocesseur à son poste de télévision et mit au point Pong, le premier jeu vidéo, il ne se doutait sans doute pas des conséquences.

L'idée de Bushnell était simple : donner au spectateur le moyen de contrôler ce qu'il voyait sur l'écran... De quoi bouleverser l'industrie du divertissement, captiver des millions de jeunes, et causer le plus grand tort à leur argent de poche.

Avec deux partenaires, Ted Dabney et Larry Bryan, Bushnell consacra quelques centaines de dollars à la promotion de Pong. A sa première apparition en 1972, à Sunnyvale (Californie), Pong ne tarda pas à se montrer extrêmement rentable. Les jeux vidéo venaient de naître et, avec eux, la société Atari, que Nolan Bushnell fonda à cette époque. Peu après, la décision fut prise d'acheter les droits de l'invention de Bushnell. C'est en 1976 que la grande compagnie américaine, la Warner Communications Inc., rachète Atari. Au passage, Bushnell reçoit quelque 15 millions de dollars. Multimillionnaire en moins de quatre ans!... De là date la domination d'Atari sur le marché des jeux vidéo, même si c'est la filiale américaine de Philips, Magnavox, qui proposa la première le système de la console de jeux branchée sur un téléviseur.

La firme maintint sa position dominante jusqu'à la fin des années soixante-dix, quand les habitués commencèrent à délaisser les jeux d'arcades pour les micro-ordinateurs. Vendre des jeux est un peu comme vendre des disques : il faut dénicher les futures vedettes et assurer leur promotion. Il était donc logique qu'Atari entrât dans le giron de Warner Communication International, cet énorme conglomérat financier bien connu dans les milieux de la musique et du cinéma. Mais la période des vaches grasses prit fin, et en 1983 les recettes diminuèrent d'environ 25 %.

Habilement lancé par Atari, Space Invaders, de la firme japonaise Taito, est sans doute le plus connu de tous les jeux pour ordinateurs. Il devint un véritable phénomène social, et donna naissance à d'innombrables répliques de ce « combat à mort à travers la galaxie ». Continuant sur sa lancée, Atari mit en circulation d'autres gros succès : Asteroids, Battlezone, Centipede, Lunar Lander, Missile Command, The Tempest, et, jusqu'à la fin des années soixante-dix, fut au centre du boom des jeux d'arcades.

Puis celui-là s'interrompit aussi brutalement qu'il avait commencé. Les clients étaient passés aux micro-ordinateurs pour deux grandes raisons : on pouvait jouer aux mêmes jeux, mais sans bourse délier ; et on disposait par surcroît d'une machine capable de faire bien des choses.

Atari entreprit d'abord d'adapter ses meilleurs jeux d'arcades à l'intention des ordinateurs domestiques. Des cartouches à semi-conducteurs s'enfichaient à l'arrière de ceux-ci et suppléaient ou remplaçaient la mémoire morte. Le système était très pratique, puisqu'on n'avait pas besoin de charger le programme à partir d'une cassette ou d'une disquette ; mais l'emploi des semi-conducteurs imposait des prix de vente élevés. Par ailleurs, les cartouches n'étant pas reprogrammables (leur contenu était gravé dans les circuits), l'échec commercial d'un jeu laissait à la firme des stocks inutilisables.

La roue tourne

Atari commit aussi de grosses erreurs de marché. Les prévisions de ventes avaient été établies par référence à celles de PacMan, qui avait connu un succès colossal ; il fallut bien payer le prix de cet optimisme surfait. Les invendus firent l'objet d'un inventaire, et quatorze camions menèrent jusqu'à un grand trou,



Nolan Bushnell

Le destin d'Atari est lié à l'origine à celui d'un seul homme : Nolan Bushnell. Lorsqu'en 1971 il créa Pong, le premier jeu vidéo, il ignorait sans doute quelle boîte de Pandore il venait d'ouvrir. (Cl. Atari.)

creusé dans le désert du Nevada, des cartouches qui valaient entre 8 et 25 dollars...

Atari négligea également une particularité unique des jeux pour ordinateurs : il n'est pas nécessaire de les déposer sur un support tangible ; on peut les transmettre sans problème par téléphone ou par câble, les diffuser à la radio ou à la télévision.

Pour cela, il est désormais possible de recourir à des techniques ou à des matériels nouveaux. En 1983, par exemple, une compagnie américaine, la Romox Corporation, dévoila une



Une console de jeu spécialisée

Pour 1 500 francs environ, le Video Computer System (VCS) d'Atari est livré avec deux manches à balai, un adaptateur secteur et un jeu PacMan.

Une telle console est exclusivement prévue pour les jeux (tous installés sur cartouche), et ne peut être utilisée comme un ordinateur d'usage général.



machine appelée Romox Programming Terminal. Elle comporte un disque d'une capacité de 15 méga-octets, qui peut être mis en relation par téléphone avec une base de données comportant de nombreux logiciels de jeux. Il y a aussi un programmeur d'EPR0M, sur lequel on peut enficher la plupart des cartouches. Cela signifie qu'on peut désormais se rendre chez le détaillant le plus proche, consulter la liste des jeux les plus vendus, en choisir un et le voir aussitôt gravé sur une cartouche Romox vierge.

Toujours aux États-Unis, Bill van Meister a lancé Gameline, dont on pourrait résumer la formule par : « Jouez maintenant, payez plus tard ! » La firme vend un modem qui raccorde la console de jeux Atari VCS au téléphone. Il en coûte 1 dollar pour trois quarts d'heure de jeu.

Major Havoc

Au cours des années soixante-dix, Atari tira d'énormes profits des jeux d'arcades comme celui-ci (un des innombrables combats dans la galaxie). L'avènement des micro-ordinateurs imposa un bouleversement complet des méthodes de marché.



Deux des plus gros réseaux informatiques de communication des États-Unis, Comuserve et The Source, proposent des jeux parmi l'éventail de services qu'ils offrent à leurs abonnés. Coleco, autre célèbre fournisseur de jeux vidéo, a conclu un accord avec ATT (American Telephone and Telegraph) pour promouvoir un système de jeu interactif. Atari, pensant également que là est l'avenir, s'est associé avec Activision pour un projet nommé Ataritel. Encore mystérieux, ce projet doit aussi lui permettre l'envoi de programmes de jeux par téléphone. Tant que la firme appartiendra au groupe Warner, tout restera possible : car la maison mère possède la Warner Amex Cable Communications, un réseau dont tout dépend ; si elle décidait de mettre en vente Atari, le projet n'aurait plus de raison d'être.

Les problèmes d'Atari, bien réels, viennent en partie des luttes internes entre le département des jeux vidéo, en perte de vitesse, et celui des ordinateurs domestiques, en pleine croissance. La firme propose cependant toute une gamme de micro-ordinateurs qui ont longtemps été parmi les meilleurs pour ce qui est des possibilités graphiques et de la commodité d'emploi. Les nouveaux modèles disposent de trois circuits intégrés spécifiques, familièrement appelés Pokey (ports d'entrée et de sortie), Antic (le graphisme) et GTIA (la couleur).

Tous sont construits autour du microprocesseur 6502, et de nombreux programmes utilitaires sont déjà disponibles. On peut citer parmi eux : VisiCalc, Atariwriter (un traitement de texte), et un programme de gestion domestique. Les ordinateurs Atari peuvent par ailleurs recevoir le Personal CP/M de Digital Equipment, grâce à l'adjonction d'une carte comportant un microprocesseur Z80.

Atari ne se désintéresse pas pour autant des logiciels. Ses filiales étrangères sont largement mises à contribution et les meilleurs programmes sont transformés afin qu'ils puissent tourner sur d'autres ordinateurs (ceux qui se vendent le mieux). Des talents nouveaux de créateurs sont recherchés. Après une brève période très difficile, Atari semble pouvoir retrouver sa grandeur d'antan.

PROGRAMME N° 13

COMPARAISON DES CHAÎNES

La comparaison de chaîne est faite (par BASIC) de gauche à droite, caractère par caractère. Dès qu'un caractère d'une chaîne devient différent du vis-à-vis, la comparaison s'arrête.

```
10 INPUT "PREMIER NOM :"; N1$
20 INPUT "DEUXIEME NOM :"; N2$
30 PRINT
40 IF N1$ > N2$ THEN PRINT N1$
  "EST PLUS GRAND QUE"; N2$
45 IF N1$ < N2$ THEN PRINT N1$
  "EST PLUS PETIT QUE"; N2$
50 GOTO 10
```

```
$RUN
PREMIER NOM : MARIN
DEUXIEME NOM : MARTIN
MARIN EST PLUS PETIT QUE MARTIN
PREMIER NOM : MARTIN
DEUXIEME NOM : MARIN
MARTIN EST PLUS GRAND QUE MARIN
```

CONCATÉNATION DES CHAÎNES

La concaténation (ou l'addition) des chaînes se fait à l'aide de l'opérateur « + » (comme pour les variables numériques).

Exemple :

```
10 INPUT "DONNEZ UN NOM :"; N1$
20 PRINT : INPUT "DONNEZ UN AUTRE NOM :"; N2$
30 PRINT
40 Z$ = N1$ + N2$
50 K$ = N1$ + " " + N2$
60 PRINT Z$
70 PRINT
80 PRINT K$
85 PRINT : PRINT : PRINT :
90 GOTO 10
```

```
$RUN
DONNEZ-MOI UNE PHRASE : ABC INFORMATIQUE C'EST
GENIAL
```

```
ABC
INFORMATIQUE
C'EST
GENIAL
```

Autre exemple, découpons une phrase :

```
10 INPUT "DONNEZ-MOI UNE PHRASE :"; A$
20 PRINT
25 REM BOUCLE SUR LA LONGUEUR DE A$
30 FOR I = 1 TO LEN (A$)
35 REM ON EXTRAIT LE I EME CARACTERE DE LA BOUCLE
40 Z$ = MID$ (A$, I, 1)
45 REM ON CONCATENE LES CARACTERES CONSECUTIFS
  DIFFERENTS D'UN BLANC
50 IF Z$ < >" " THEN T$ = T$ + Z$
55 REM ON AFFICHE LE MOT RECONSTITUE ET ON REMET
  A 0 T$
60 IF Z$ = " " THEN PRINT T$ : T$ = " "
65 REM FIN DE LA BOUCLE
70 NEXT I
80 PRINT T$
```

Autre instruction nécessaire à la maîtrise des chaînes :

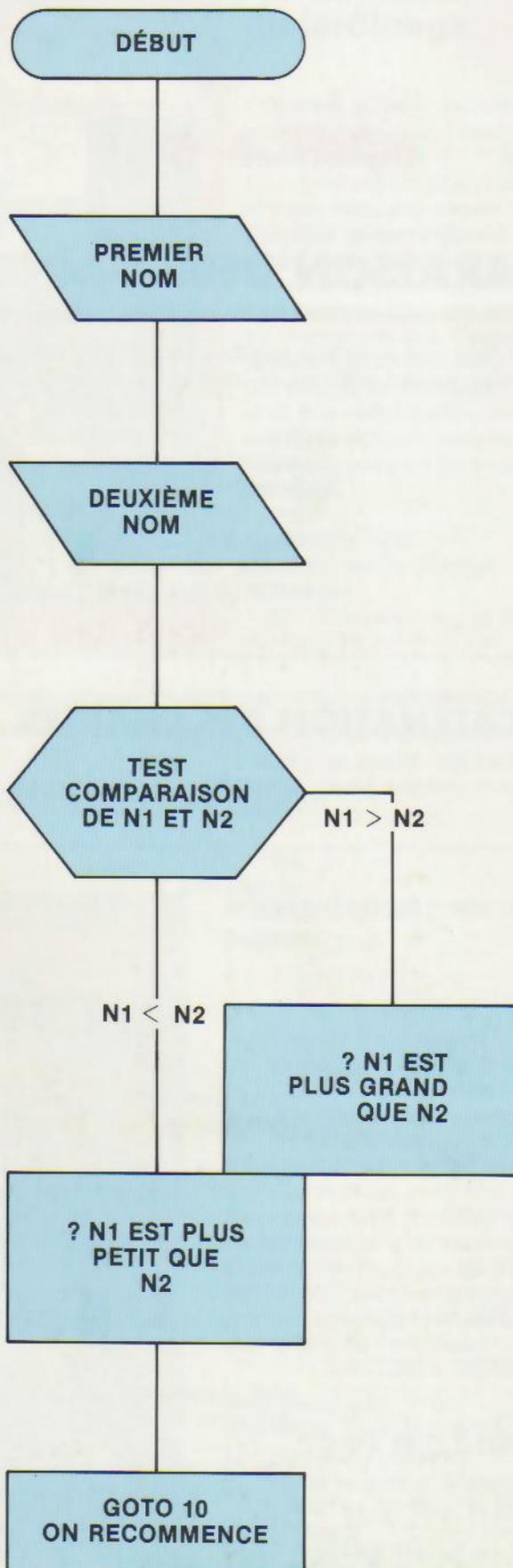


ASC

ASC (caractère) donne le code ASCII de ce caractère.

Tous les caractères alphabétiques, numériques, etc., sont représentés de façon interne au niveau de la machine pour une forme binaire. Le programmeur peut accéder à ce mode sous sa forme décimale.

ORGANIGRAMME



Un petit exemple :

```

10 Z$ = "A"
20 ? ASC (Z$)
RUN
65
  
```

Les codes ASCII des lettres de l'alphabet A B C D... Z sont 65 66 67 68... 91.

Une autre utilisation de ASC est sous forme de l'instruction ASC (chaîne).

Cette manière d'utiliser ASC donne le code ASCII du premier caractère de la chaîne placé à l'intérieur des parenthèses.

Remarque : la chaîne ne doit pas être vide.

Un exemple :

```

10 Z$ = "ABRACADABRA"
20 ? ASC (Z$)
RUN
65
  
```

La première lettre de la chaîne est A dont le code obtenu par la machine est 65. La « réciproque » de l'instruction ASC est, nous le rappelons, CHR\$ (code ASCII), instruction qui donne le caractère dont le code ASCII est fourni entre parenthèses.

Certains caractères seulement sont imprimables.

Un exemple :

```

10 Z = 65
20 ? CHR$ (Z)
RUN
A
  
```

Le caractère dont le code ASCII est 65 est A.

Un autre exemple :

```

10 FOR I = 65 TO 91
20 ? CHR$ (I);
20 NEXT I
RUN
  
```

A B C D... Z,
caractères qui correspondent à :
65 66 67 68... 91.

Exemple de code ASCII non imprimable :

? CHR\$ (10) provoque un saut de ligne (sans retour en début de ligne).

? CHR\$ (13) provoque un retour en début de ligne (sans saut de ligne).

Si votre unité centrale dispose d'un haut-parleur, ? CHR\$ (7) vous donnera un top sonore.

Pour prolonger cette sonnerie, on peut écrire :

```

10 FOR I = 1 TO 100
20 ? CHR$ (7)
30 NEXT I
  
```

(Pour éviter le saut de ligne à l'écran, taper; après CHR\$7.)