

LOISIRS TECHNIQUES D'AUJOURD'HUI

hors série

Led

MICRO

PROGRAMMATION

COURS 2^{ème} CYCLE

COURS
N°41

Suite
2^e cycle

N°21

**COURS DE
PASCAL**
les entrées-
sorties en pascal
standard

**COURS DE
PROGRAM-
MATION
APPROFONDIE :**
éditeur de texte

**Logiciels
scientifiques pour
électroniciens**



ISSN 0757-6889

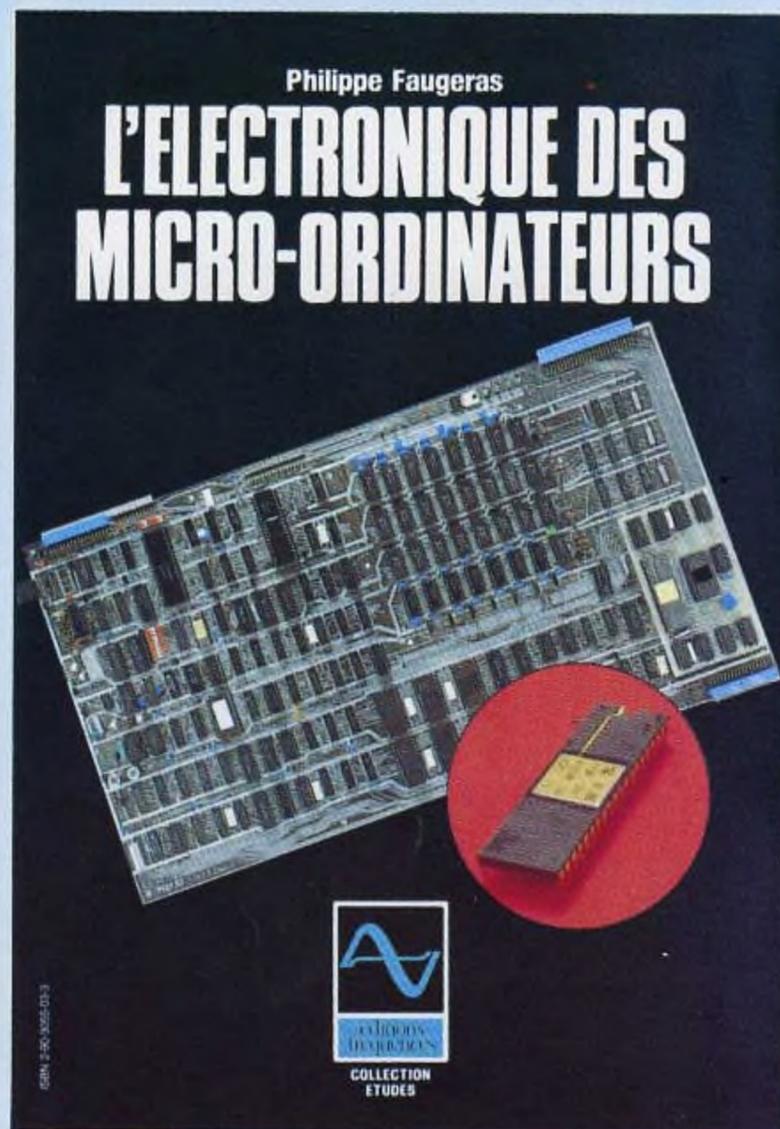
M 1988 - 41 - 18,00 F



3791988018008 00410

VOYAGE AU COEUR DES MICRO-ORDINATEURS

dans la
COLLECTION
«ETUDES»
aux
éditions
fréquences



une véritable schémathèque

- 128 pages
 - 101 schémas
 - 34 tableaux
- Prix : 150 F

Que ce soit pour concevoir des interfaces ou optimiser un programme (utilisation des périphériques, encombrement mémoire...) «un micro-informaticien performant» doit posséder une bonne connaissance de son matériel.

Ce livre s'adresse donc à tous les électroniciens qui désirent découvrir les différents

composants constituant un micro-ordinateur. Articulé autour du microprocesseur Z80, cet ouvrage contient de nombreux schémas (plan mémoire, interfaces série et parallèle, interface clavier, interface vidéo, CAN, CNA...) qui pourraient être le thème... de nouvelles extensions.

En vente chez votre libraire et aux Editions Fréquences

BON DE COMMANDE

Je désire recevoir l'ouvrage **L'électronique des micro-ordinateurs** au prix de **160 F** (150 F + 10 F de port).

Nom

Adresse

.....

A adresser aux EDITIONS FREQUENCES 1 boulevard Ney, 75018 Paris

Règlement ci-joint :

Par chèque bancaire par chèque postal par mandat

Philippe Faugeras, Docteur-ingénieur en électronique a acquis son expérience dans de grandes entreprises françaises où pendant cinq ans, il a travaillé sur des systèmes d'automatismes à base de microprocesseurs. Philippe Faugeras est responsable de la rubrique «Raconte-moi la micro-informatique» dans la revue LED.

LOMBOBOES D'AUJOURD'HUI

hors série

LED

MICRO

PROGRAMMATION COURS 2^e CYCLE

Société éditrice :
Editions Fréquences
 Siège social :
 1, bd Ney, 75018 Paris
 Tél. : (1) 46.07.01.97 +
 - SA au capital de 1 000 000 F
 Président-Directeur Général :
 Edouard Pastor

LED MICRO
 (cours 2^e cycle)
 Mensuel : 18 F
 Commission paritaire : 64949
 Directeur de la Publication :
 Edouard Pastor
 Tous droits de reproduction réservés
 textes et photos pour tous pays
 LED MICRO est
 une marque déposée ISSN 0757-6889

**Services Rédaction-Publicité-
 Abonnements :**
 1 bd Ney, 75018 Paris
 Tél. (1) 46.07.01.97
 Lignes groupées

Comité de rédaction :
 Dominique Chastagnier
 Jean-François Coblentz
 Charles-Henry Delaleu
 Patrick Gueneau

Secrétaire de Rédaction
 Chantal Cauchois

Publicité A la revue
 Tél. : 46.07.01.97
 Secrétaire responsable
 Annie Perbal

Abonnements
 10 numéros par an
 France : 160 F
 Etranger : 240 F

Réalisation
Composition-Photogravure
 Edi Systèmes
 Impression
 Berger-Levrault - Nancy

Les Editions Fréquences éditent
 La Nouvelle Revue du Son
 Son Vidéo Magazine, L'Audiophile
 Led Micro, Zero-VU magazine
 Music Vidéo Systèmes



JUIN/JUILLET 1987

Nous signalons à nos lecteurs que les Editions Fréquences seront fermées du 3 juillet au 3 août 1987 pour congés annuels.

COURS DE PASCAL
de la page 5 à la page 12
 - Utilisation de fichiers entrées-sorties en pascal standard .. p. 6
 ● Introduction et lamentations
 ● Le pascal standard
 - les premiers pas
 - ouvrir un fichier
 - les autres commandes
 ● Les entrées-sorties standard
 - Conclusion p. 12
Dominique Chastagnier
Jean-François Coblentz
Patrick Gueneau

COURS DE PROGRAMMATION
APPROFONDIE
de la page 13 à la page 23
 - Editeur de texte p. 14
Dominique Chastagnier
Jean-François Coblentz
Patrick Gueneau

DIALOGUE
AVEC NOS LECTEURS
de la page 24 à la page 28
 - Les carrés magiques p. 24
 - Les chiffres et les lettres ... p. 25

C'EST ARRIVE DEMAIN
de la page 29 à la page 33

LOGICIELS :
DELITS ET PIRATAGE
Pages 38-39

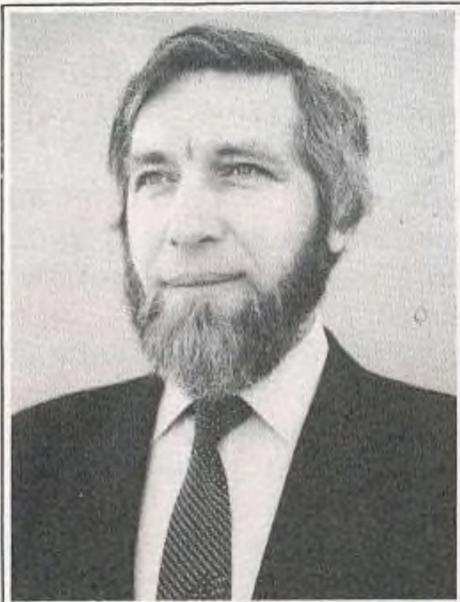
LOGICIELS SCIENTIFIQUES
POUR ELECTRONICIENS
de la page 40 à la page 49
 - La simulation de circuits électroniques p. 41
 - AC Circuits Analysis p. 41
 - Linear System Analysis p. 44
 - Analyse de signaux ou FFT . p. 46
Charles-Henry Delaleu

NOTRE COUVERTURE : La série 9000 de Hewlett-Packard, résolument spécialisée pour les domaines techniques et scientifiques : au-dessus un 9836 exécutant le logiciel FFT décrit dans ce numéro, au-dessous un 9000 série 500 converti en atelier de C.A.O.

Electronique digitale ?

Notre temps aura témoigné d'une nouvelle technique, une autre façon de communiquer avec l'électronique digitale.

Philippe Duquesne, professeur chargé de cours au CNAM a su dans cet ouvrage en expliquer clairement les fondements.



Philippe Duquesne, ingénieur électronicien (I.S.E.N.) est chargé du cours de microprocesseurs au C.N.A.M. de Paris. Depuis plus de dix ans, il a pris goût à l'enseignement et il est l'auteur d'un ouvrage didactique sur l'électronique digitale et notamment d'un cours pratique de microprocesseurs. Fervent pratiquant du « dialogue » école/industrie, après avoir exercé les fonctions de chef de département électronique chez Burroughs, second constructeur mondial en informatique, il est actuellement chef du service Etudes Electroniques au sein de la direction technique chez Messier Hispano Bugatti (groupe SNECMA) avec, pour principal objectif l'introduction des microprocesseurs dans les trains d'atterrissage.



En vente chez votre libraire et aux Editions Fréquences

Bon de commande

Je désire recevoir le livre : INITIATION A L'ELECTRONIQUE DIGITALE au prix de 105 F (95 F + 10 F de port).

Adresser ce bon aux EDITIONS FREQUENCES 1, bd Ney, 75018 PARIS

Nom Prénom

Adresse

Code postal

Règlement effectué : par C.C.P.

par chèque bancaire

par mandat

COURS DE PASCAL

Dominique Chastagnier
Jean-François Coblentz
Patrick Gueneau

Dans ce numéro, vous allez découvrir les joies de l'emploi de fichiers d'entrées/sorties tels qu'ils sont prévus par le Pascal standard. Il n'y a là que la norme stricto sensu mais ce n'est déjà pas mal.

COURS N° 12

PLAN DU COURS

1. Utilisation de fichiers entrées-sorties en Pascal standard
 - Introduction et lamentations
 - Le Pascal standard
 - les premiers pas
 - ouvrir un fichier
 - les autres commandes
 - Les entrées-sorties standard
2. Conclusion

I. UTILISATION DE FICHIERS ENTREES-SORTIES EN PASCAL STANDARD

I.1. Introduction et lamentations

Cette partie du Pascal est celle qui suit le moins un standard. C'est sans doute dû au fait que le créateur de Pascal, Niklaus Wirth, ne s'est intéressé qu'à la partie logique du langage et peu à la partie pratique. Il faut le déplorer, car le résultat est à la hauteur de nos espérances, c'est-à-dire lourd, peu efficace sur la plupart des systèmes. En fait c'est surtout net sur les plus gros systèmes où les temps d'accès pour de très grandes quantités de données peuvent être comparés significativement. Sur des petits systèmes, le plus souvent, la différence de temps avec d'autres langages n'est pas trop importante et en tout état de cause, ne peut justifier le rejet du langage.

Nous allons donc montrer comment utiliser des fichiers en Pascal et donner ensuite les différences suivant les Pascal utilisés. Nous traiterons en fait les Pascal UCSD, Turbo et Macintosh qui sont les plus répandus.

I.2. Le Pascal standard

I.2.1. Les premiers pas

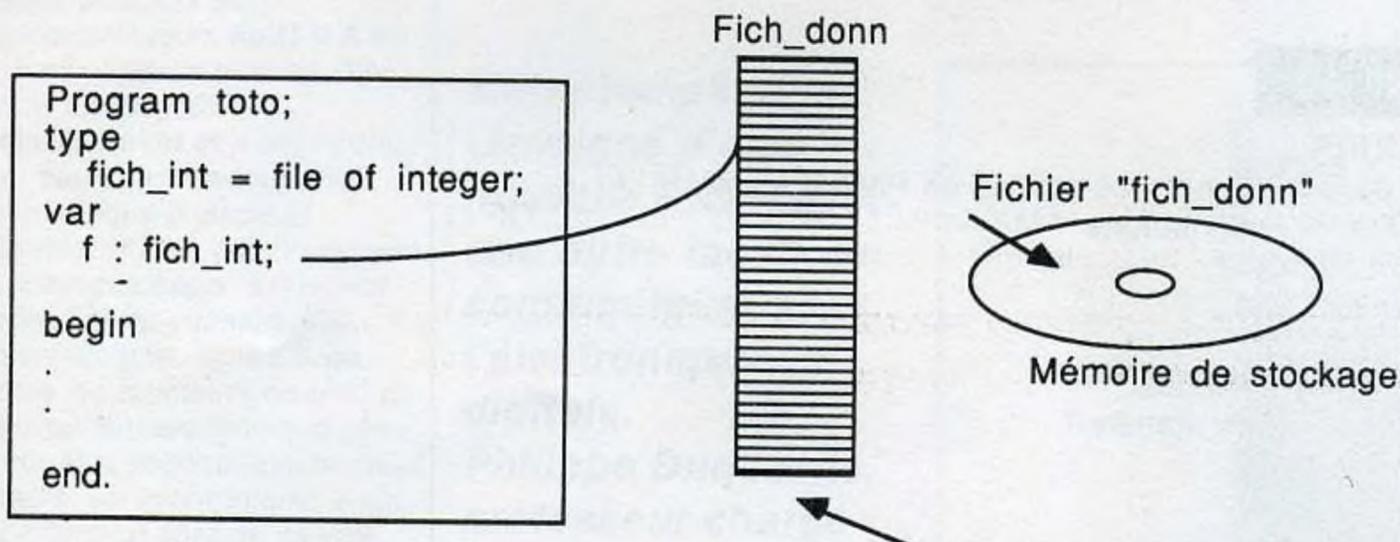
Tout d'abord, vous disposez d'un type fichier qui s'appelle FILE, anglais oblige. Pour l'utiliser, il suffit de déclarer un fichier de la façon suivante :

Type f = file of truc

où TRUC est un type déjà connu du programme, Integer,... ou tout type que vous aurez défini auparavant. Un type prédéfini de fichier est le fichier texte qui est un fichier contenant des caractères au format ASCII.

I.2.2. Ouvrir un fichier

Pour travailler sur un fichier, il est nécessaire de l'ouvrir. L'opération d'ouverture d'un fichier permet de lier le vrai nom du fichier à celui de la variable du programme qui permet de l'utiliser. Oh, je vois très bien que je ne suis pas clair !!! Donc, un petit dessin pour me faire comprendre :



f sera le nom que le programme utilisera de manière interne pour parler de fich_donn. Cela s'appelle le nom logique du fichier. Il ne représente rien, sinon une référence pour le programme.

Représentation exacte en mémoire de ce fichier. Ici, nous prenons un fichier d'entiers.

Ceci dit, ce lien entre le fichier et le nom que le programme utilise ne se fait pas tout seul. Il faut le dire au programme. Nous allons voir comment le faire. Pour commencer, une liste de toutes les commandes que nous allons détailler :

RESET
REWRITE
CLOSE

ERASE
IORESULT
READ
WRITE
SEEK
EOF
EOLN
SEEKEOF
SEEKEOL
FILRPOS
FILESIZE
READLN
WRITELN

Pour ouvrir un fichier, deux commandes selon ce dont on dispose déjà :

Si le fichier n'existe pas encore, il faut faire :

```
REWRITE(f,"nom__du__fichier");
```

Ceci crée le fichier qui porte le nom «nom__du__fichier» et dont le lien dans le programme est f. Si un fichier du même nom existe déjà, il est détruit.

Si le fichier existe déjà, il faut simplement l'ouvrir, et la commande est :

```
RESET(f,"nom__du__fichier");
```

Ceci représente les commandes de base. Maintenant, il est possible de les utiliser un peu différemment :

RESET d'un fichier déjà ouvert ramène à la lecture du fichier depuis le premier élément du fichier. Cela veut dire que c'est équivalent à le fermer et à le réouvrir. Dans ce cas, il n'est pas nécessaire d'indiquer le nom réel du fichier puisque le programme connaît le lien entre le fichier et son nom logique. Il suffit donc de faire :

```
RESET(f);
```

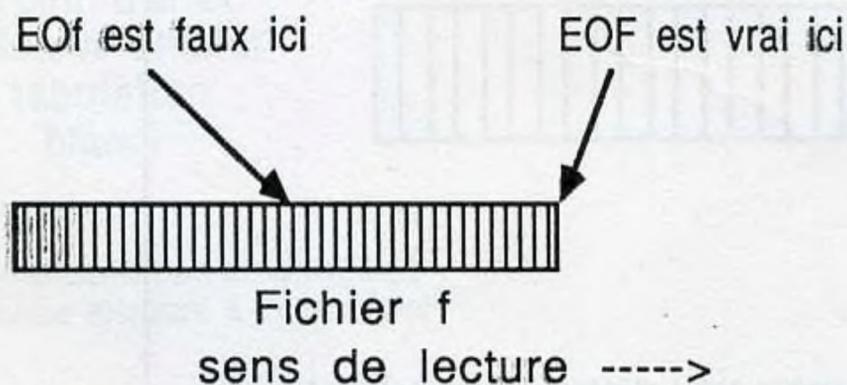
Bien sûr, ce fichier doit déjà être ouvert.

Pour réécrire sur un fichier déjà ouvert, il suffit de faire :

```
REWRITE(f);
```

Bien sûr, ce fichier doit déjà être ouvert.

Si f était un fichier texte, RESET le positionne en lecture seulement et REWRITE en écriture seulement. Après un appel à REWRITE, le booléen EOF est toujours vrai. EOF signifie End Of File et est vrai si le pointeur de lecture est sur le dernier élément du fichier. Comme le fichier est vide, il est clair que l'on est toujours après le dernier.



1.2.3. Les autres commandes

1.2.3.1. Pour tout type de fichiers

CLOSE(f) : ferme un fichier qui était ouvert. Rompt le lien entre le fichier et son nom logique dans le programme.

RENAME(ancien,nouveau) : Permet de changer le nom réel d'un fichier, nouveau et ancien sont des chaînes de caractères.

ERASE(nom) : permet d'effacer un fichier dont le nom est spécifié par la chaîne "nom".

IORESULT : renvoie un entier qui est la valeur de la dernière opération d'entrée-sortie effectuée. Les codes sont des nombres entiers négatifs. La documentation de Turbo Pascal donne la liste de ces codes en appendice.

1.2.3.2. Pour les fichiers qui ne sont pas des fichiers textes

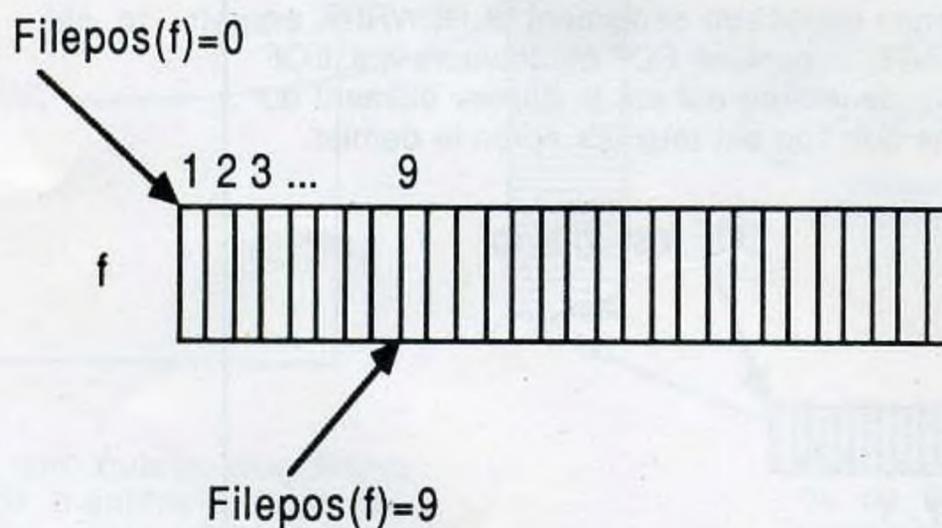
READ([f],v1 [,v2,...,vk]) : permet de lire dans le fichier dont f est le lien logique, la valeur v1 ou les valeurs v1 à vk. Il est possible de lire de cette manière autant de valeurs qu'on le veut. Les [] représentent des termes optionnels de cette commande. Si on veut lire trois valeurs, on mettra read(f,v1,v2,v3) mais pour en lire une seule, on mettra read(f,v1). Si EOF(f) est vrai, il n'y a plus rien à lire dans le fichier et tenter de lire un élément malgré tout est une erreur.

WRITE(f,v1 [,v2,...,vk]) : même type de travail que READ mais en écriture du fichier. Si l'on se trouve en fin de fichier, il est augmenté d'autant.

SEEK(f,n) : déplace le pointeur de lecture du fichier pour se placer au début de l'élément qui est le (n+1)^{ème} du fichier. n est un entier. Attention, le premier élément d'un fichier a le numéro 0 d'où le décalage de une place de notre définition. Si le fichier contient moins de n composants, le pointeur est placé en fin de fichier et EOF(f) devient vrai.

EOF(f) : fonction dont le résultat est un booléen et qui est vrai si l'on se trouve en fin de fichier, faux dans le cas contraire. Se référer à la figure de la page précédente.

FILEPOS(f) : permet de connaître la position du pointeur de lecture du fichier. Si l'on est en tête du fichier, la valeur est 0, ainsi de suite jusqu'à la fin du fichier, cas où la valeur est celle du nombre de composants du fichier qui est donné par la fonction FILESIZE(f), décrite ci-après.



FILESIZE(f) : donne le nombre total d'éléments du fichier f.

1.2.3.3. Pour les fichiers de type texte

Note préliminaire : Utiliser un fichier texte ne signifie pas que chaque élément de ce fichier doit être lu comme un élément de type CHAR. Simplement si un élément est lu par une variable qui est d'un type différent, la conversion est faite automatiquement par le programme. Les types de caractères compatibles avec cette conversion sont les types INTEGER, REAL, STRING (pour les versions qui en disposent) et CHAR naturellement.

READ[f],v1 [,v2,...,vk]) : lit un ou plusieurs éléments dans le fichier f.

Quelques propriétés particulières de cette commande dans le cas des fichiers texte :
 Si le paramètre f n'est pas mis, la lecture est faite sur le fichier standard INPUT qui est en général le clavier.

Si EOF(f) est vrai avant la lecture (on est en fin de fichier), il n'y a pas d'erreur comme dans le cas d'un fichier quelconque mais la variable de lecture est mise à CHR(0).

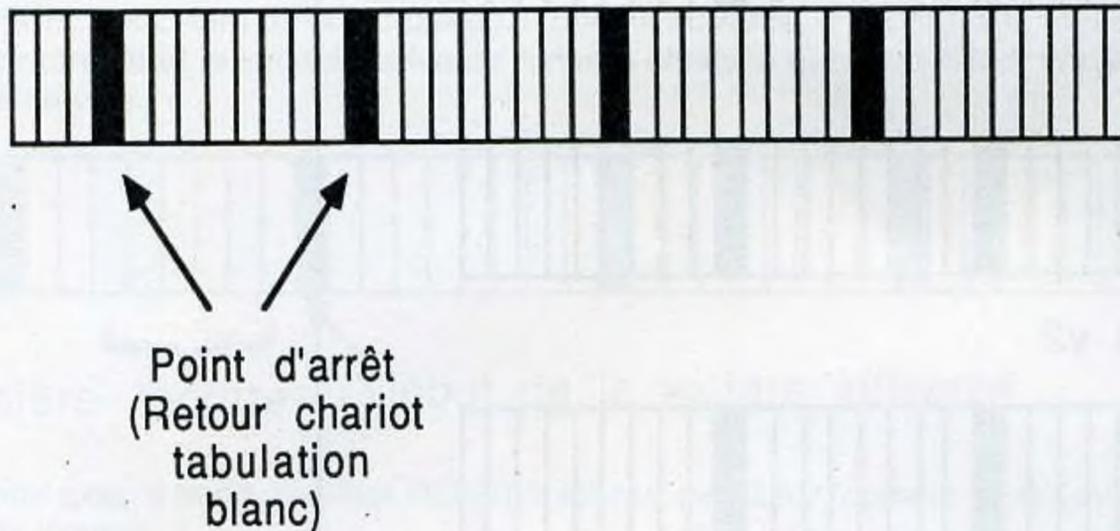
Si le pointeur se trouve sur un retour chariot, la variable est mise à CHR(13). Le READ-suivant lira le caractère suivant.

Si READ est fait avec un INTEGER (READ(f,i) où i est un entier), il est indispensable que la suite de caractères lus forment un nombre sinon une erreur est générée. La lecture va d'un blanc, retour chariot ou tabulation à un autre blanc, retour chariot ou tabulation ou si EOF(f) devient vrai. Si tout se passe bien, la valeur comprise est donnée à la variable de lecture, i dans l'exemple précédent. La lecture ignore les blancs, retours-chariot et tabulations. Si EOF(f) est vrai avant le début de la lecture ou au cours de la lecture, la valeur 0 est affectée à la variable.

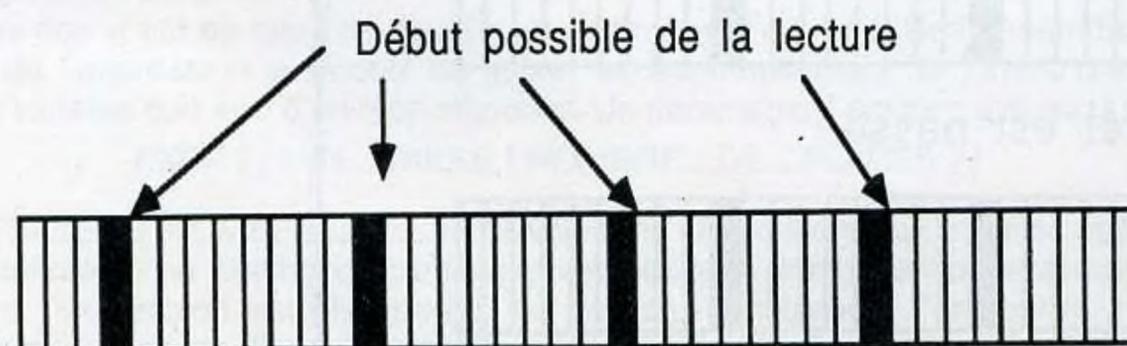
Il faut bien comprendre ce que signifie la lecture d'un point d'arrêt (tabulation, blanc, retour chariot) à un autre, avec saut des points d'arrêts en cours de lecture. Cela signifie que :

- la lecture commence toujours à un tel point d'arrêt ;
- si de tels points sont trouvés en cours de lecture, ils sont ignorés ;
- si la lecture s'arrête entre deux points d'arrêts et non sur un tel point, la lecture de l'élément suivant débutera après le second.

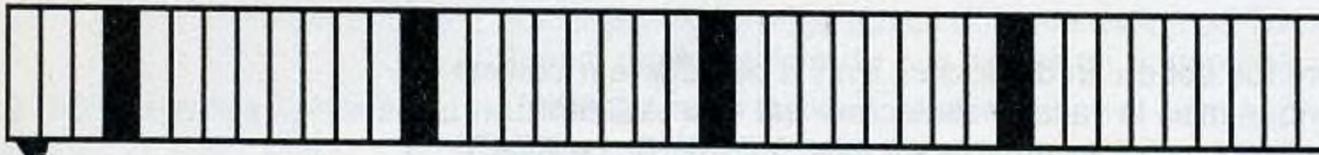
Une illustration rapide de ce qui précède :



Reprenons point par point ces trois règles :
 La lecture commence toujours à point d'arrêt :



Si de tels points sont rencontrés en cours de lecture, ils sont ignorés :



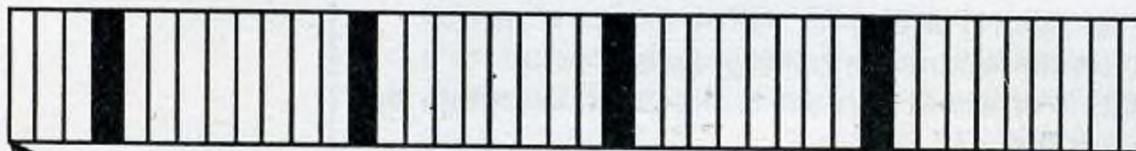
Début de la lecture

`READ(f,v1,v2,v3,v4);`

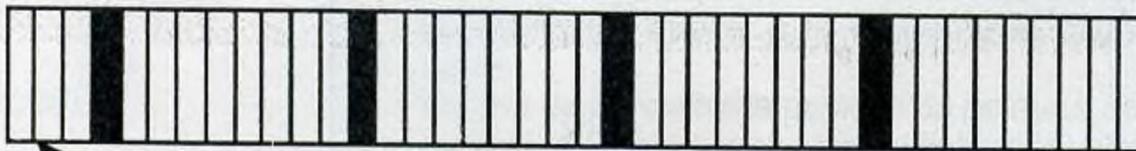
Après avoir lu v3, le pointeur est sur un point d'arrêt.
Mais la lecture n'est pas finie. Le point d'arrêt est ignoré.
La lecture continue avec l'élément qui suit ce point
d'arrêt jusqu'à la fin de la lecture.

Voici le détail :

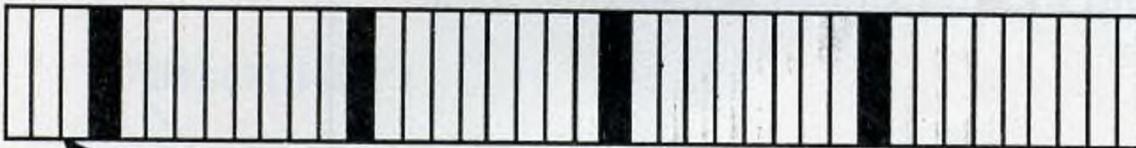
`READ(f,v1,v2,v3,v4);`



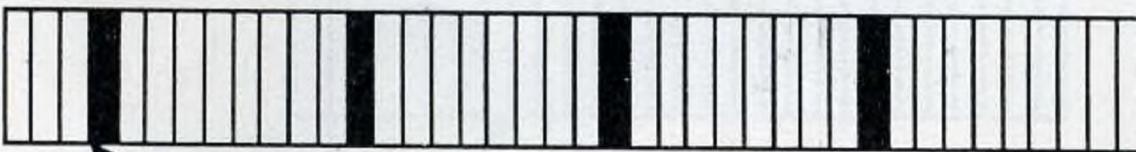
Début de la lecture



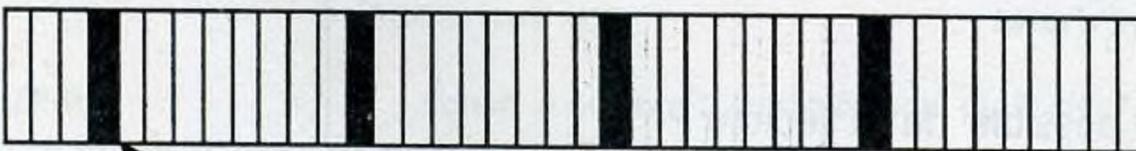
Après lecture de v1



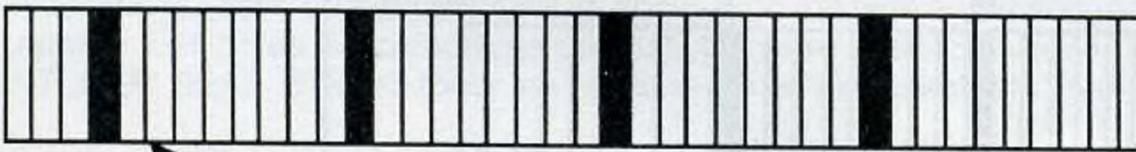
Après lecture de v2



Après lecture de v3



Le point d'arrêt est passé



Après lecture de v4

Si la lecture s'arrête entre deux points d'arrêts et non sur un tel point, la lecture de l'élément suivant débutera après le second : reprenons le cas ci-dessus.
Après avoir lu v4, la commande READ est terminée. A la prochaine commande READ, la lecture reprendra après le point d'arrêt suivant :



Si READ est fait avec un REAL (READ(f,x) où x est un réel), il est indispensable que la suite de caractères lus forment un nombre réel, sinon une erreur est générée. Comme précédemment, la lecture va d'un blanc, retour chariot ou tabulation à un autre blanc, retour chariot ou tabulation ou si EOF(f) devient vrai. Si tout se passe bien, la valeur comprise est donnée à la variable de lecture, x dans l'exemple précédent. La lecture ignore les blancs, retours chariot et tabulations. Si EOF(f) est vrai avant le début de la lecture ou au cours de la lecture, la valeur 0 est affectée à la variable. Nous voyons donc que tout se passe de la même manière que pour des entiers.

Si READ est fait avec une variable de type CHAR (READ(f,a) où a est un CHAR), la lecture va d'un retour chariot à un autre retour chariot, ou si EOF(f) devient vrai. Si tout se passe bien, la valeur comprise est donnée à la variable de lecture, a dans l'exemple précédent. Si EOF(f) est vrai avant le début de la lecture ou au cours de la lecture, la valeur 0 est affectée à la longueur de la variable. De plus, quelques différences apparaissent dans le cas de la lecture de variable de type CHAR.

Ainsi, si vous déclarez une longueur (n) insuffisante pour votre chaîne, elle est tronquée en lecture, ne gardant que les n premiers caractères, les autres sont ignorés.

Après la lecture d'une chaîne, le pointeur va au retour chariot suivant, SANS LE DEPASSER. Donc, lors de la lecture de la chaîne suivante, le programme ne verra que ce retour chariot et la variable suivante ne sera chargée que d'un retour chariot donc une chaîne vide.



Pour éviter cela, il suffit d'utiliser READLN au lieu de READ, comme nous l'avons fait en usage normal.

WRITE([f],v1 [,v2,...,vk]) : écrit un ou plusieurs éléments dans le fichier f. Quelques propriétés particulières de cette commande dans le cas des fichiers texte :

Si un paramètre f n'est pas mis, la lecture est faite sur le fichier standard OUTPUT qui est en général l'écran.

Chacune des vi est ce que l'on appelle un paramètre d'écriture. Ces paramètres sont formés de l'expression à ajouter au fichier et, éventuellement, le format d'écriture. Chaque variable doit être d'un type standard. Un paramètre d'écriture a donc la forme :

EXPR [: MIN_TAILLE [NOMBRE_DE_PLACES]]

MIN_TAILLE et NOMBRE_DE_PLACES sont des entiers. Le premier spécifie la taille minimale d'un champ de cette expression, taille minimale qui sera respectée, même si l'expression est plus petite, en ajoutant des blancs. Par contre une taille supérieure est tolérée. Si ce paramètre optionnel est omis, l'écriture s'adapte à chaque taille. NOMBRE_DE_PLACES spécifie pour une variable de type REAL exclusive-

ment le format d'écriture, avec le nombre de digits souhaité pour des problèmes de précision (dimension de la mantisse). Notons que l'écriture d'un entier se fait au format décimal et non binaire ou ASCII.

Une plus grande précision relève de la lecture de chaque PASCAL, les variations commençant en général à partir de ce point.

`WRITELN([f],v1 [,v2,...,vk])` : cette commande fait exactement la même chose que `WRITE`, puis ajoute un retour chariot. Si la liste de paramètres est vide, seul un retour chariot est écrit.

`EOF` et `EOLN` travaillent de la même manière mais deux nouvelles commandes apparaissent.

`SEEKEOF([f])` : permet de travailler comme `EOF` mais en sautant tout ce qui encombre une liste de paramètres numériques, comme les blancs, les tabulations et les retours chariot. C'est une fonction de type booléenne.

`SEEKEOLN([f])` : permet de travailler comme `EOF`, mais en sautant tout ce qui encombre une liste de paramètres numériques, comme les blancs, les tabulations et les retours chariot.

I.3. Les entrées sorties standard

Deux entrées sorties sont standard en Pascal, une en entrée, l'autre en sortie. En entrée, `CONSOLE` permet de lire des caractères venant du clavier, en sortie, `PRINTER` permet d'envoyer des données à l'imprimante, ou à tout système connecté à la sortie imprimante.

II. CONCLUSION

Nous avons vu ici une liste fastidieuse de commandes. Il est assez facile de les utiliser mais, dans la plupart des cas, il vous faudra vous lancer dans la lecture du manuel de votre version pour connaître les variations, les nuances et surtout les additions qui, de plus en plus, voient le jour. Le cas le plus flagrant est bien sûr le Macintosh dont la complexité de l'interface requiert des commandes beaucoup plus nombreuses que les autres systèmes. Menus, souris et fenêtres, ressources et File Manager demandent une gestion plus lourde pour une utilisation simplifiée. On n'a rien sans rien.

COURS DE PROGRAMMATION APPROFONDIE

Dominique Chastagnier
Jean-François Coblentz
Patrick Gueneau

Voici donc la fin de cette série consacrée à l'éditeur de texte pleine page. Elle se résume en la fourniture d'un listing PASCAL d'une version de base d'éditeur. Nous sommes désolés pour les non-«PASCALIENS», de ne pas avoir fourni de version BASIC, mais l'adaptation avec l'aide des routines en pseudo-langage ne devrait pas s'avérer insurmontable (ce serait toutefois une bonne raison pour se lancer dans la programmation PASCAL).

COURS N° 21

PLAN DU COURS
Editeur de texte

EDITEUR DE TEXTE

Les différents éléments sont en fait la traduction en PASCAL des routines en pseudo-langage décrites dans les numéros précédents. La cohérence de l'ensemble nous a obligé à apporter quelques modifications dans la description de certaines d'entre elles (et aussi pour corriger quelques erreurs). Notamment la procédure de destruction d'un caractère a été simplifiée ; par contre, les routines de mouvements de curseur ont été complétées et le curseur clignotant est désormais géré manuellement (intéressant pour tous ceux qui ne peuvent directement intervenir sur la position du curseur standard de leur ordinateur). Une dernière remarque, les variables `x_curseur` et `y_curseur` ont été substituées par `X` et `Y` dans une partie des descriptions des routines, il s'agissait bien des mêmes variables indiquant la position du curseur. Le programme fourni a été testé, il vous permettra donc une adaptation facile sur votre ordinateur car il «tourne». Bon courage !...

```
program EDITEUR (input, output);
```

```
const
```

```
  max_buffer = 200;
  nb_ligne = 24;
  nb_col = 64;
  curseur = '_';
  insert = 9;      { CTRL-I}
  surimp = 19;     { CTRL-S}
  destruc = 8;     { Backspace}
  destruc_ligne = 4; { CTRL-D}
  termi = 6;      { CTRL-F}
```

```
{ paramètres dépendants de l'ordinateur ( ici les 4 flèches du clavier)
```

```
  gauche = 28;
  droite = 29;
  haut = 30;
  bas = 31;
  cr = 13;
```

```
type
```

```
  ligne = string[nb_col]; { ou packed array[1..nb_col] of char}
```

```
var
```

```
  { tableau en mémoire}
```

```
  buffer : array[1..max_buffer] of ligne;
```

```
  { constantes définies pour les caractères de contrôle }
```

```
  sonnerie : char;
```

```
  { variables pointeurs liées au buffer }
```

```
  prem_ligne, nb_occup : integer;
```

```
  x_curseur, y_curseur : integer;
```

```
  x_ecran, y_ecran : integer;
```

```
  { flags de gestion du programme }
```

```

fin, mode_ins : boolean;

ch : char;
kb : text;

{-----}
{          INITIALISATION du PROGRAMME          }
{-----}

procedure INITIALISER;

  var
    i : integer;

  begin
    { initialisation des variables principales }

    x_curseur := 1;
    y_curseur := 1;
    nb_occup := 0;
    prem_ligne := 1;
    fin := false;
    mode_ins := true; { par défaut, on est en mode insertion }
    for i := 1 to max_buffer do
      buffer[i] := "";
    end;

    {-----}
    { POSITION_ECRAN est dépendant de la machine et du Pascal }
    { ce sera à vous de l'écrire en fonction de votre ordinateur }
    {-----}

    procedure POSITION_ECRAN (x, y : integer);

    begin
      x_ecran := x;
      y_ecran := y;

      { insérez ici le code dépendant }

    end;

    {-----}
    { LITCAR est une fonction de lecture du clavier sans echo }
    { a écrire en fonction de votre machine renvoie le caractère entré s'il }
    { existe sinon le caractère de code ascii 0 }
    {-----}

    function LITCAR : char;

    begin

    { code dépendant ici }

    end;

```

```

{-----}
{ECRIT_CAR pour l'écriture d'un seul caractère}
{ routine particulière à chaque machine }
{-----}

procedure ECRIT_CAR (c : char);

begin

{ code dépendant ici }

end;

{-----}
{ appel LITCAR et rafraichit le curseur à l'écran }
{-----}

function LECTURE : char;

const
  max_compt = 50; { à modifier suivant la vitesse souhaitée pour le curseur }

var
  c : char;
  curseur : boolean;
  compt, l : integer;

begin
  c := LITCAR;
  curseur := false;
  compt := max_compt;
  l := prem_ligne + y_curseur - 1;
  while c = chr(0) do
    begin
      if compt = max_compt then
        begin
          compt := 0;
          if curseur then
            begin
              POSITION_ECRAN(x_curseur, y_curseur);
              ECRIT_CAR('_');
            end
          else
            begin
              POSITION_ECRAN(x_curseur, y_curseur);
              if x_curseur <= length(buffer[l]) then
                ECRIT_CAR(buffer[l][x_curseur])
              else
                ECRIT_CAR(' ');
            end;
          curseur := not curseur;
        end;
      compt := compt + 1;
      c := LITCAR;
    end;
  end;

```

```

if not curseur then
  begin
    POSITION_ECRAN(x_curseur, y_curseur);
    if x_curseur <= length(buffer[1]) then
      ECRIT_CAR(buffer[1][x_curseur])
    else
      ECRIT_CAR(' ');
    end;
    LECTURE := c;
  end;

{-----}
{ ERREUR produit un simple beep par exemple ; dépend de votre ordinateur }
{-----}

procedure ERREUR;

begin
{ code dépendant ici }

end;

{-----}
{ ECRIT est dépendant de l'environnement }
{ correspond le plus souvent à un writeIn PASCAL }
{-----}

procedure ECRIT (s : ligne);

begin
{ code dépendant ici }

end;

{-----}
{ Procédure INSERER }
{-----}

procedure INSERER (car : char);

var
  s1, s2 : ligne;
  l, dep, taille : integer;

begin
  l := prem_ligne + (y_curseur - 1);
  taille := length(buffer[1]);
  if x_curseur = nb_col then
    ERREUR
  else
    begin
      if (mode_ins) then
        dep := 0
    end;
  end;

```

```

else
  dep := 1;
  s1 := copy(buffer[1], 1, x_curseur - 1);
  s2 := copy(buffer[1], x_curseur + dep, taille - x_curseur + (1 - dep));
  x_curseur := x_curseur + 1;
  buffer[1] := concat(s1, car, s2);
  POSITION_ECRAN(1, y_curseur);
  ECRIT(buffer[1]);
end;
end;

```

```

{-----}
{ DEF_HAUT_BAS provoque le défilement du texte de haut en bas }
{-----}

```

```
function DEF_HAUT_BAS : boolean;
```

```

var
  i : integer;

```

```

begin
  prem_ligne := prem_ligne - 1;
  if prem_ligne = 0 then
    begin
      DEF_HAUT_BAS := true;
      prem_ligne := 1;
    end
  else
    DEF_HAUT_BAS := false;
  for i := prem_ligne to (premier_ligne + nb_ligne - 1) do
    begin
      POSITION_ECRAN(1, i - premier_ligne + 1);
      ECRIT(buffer[i]);
    end;
  end;
end;

```

```

{-----}
{ DEF_BAS_HAUT provoque le défilement du texte de bas en haut }
{-----}

```

```
function DEF_BAS_HAUT : boolean;
```

```

var
  i : integer;

```

```

begin
  prem_ligne := premier_ligne + 1;
  if prem_ligne > nb_occup then
    begin
      DEF_BAS_HAUT := true;
      prem_ligne := nb_occup;
    end
  else

```

```

DEF_BAS_HAUT := false;
for i := prem_ligne to (prem_ligne + nb_ligne - 1) do
  begin
    POSITION_ECRAN(1, i - prem_ligne + 1);
    ECRIT(buffer[i]);
  end;
end;

{-----}
{ DEP_GAUCHE permet le déplacement vers la gauche }
{-----}

procedure DEP_GAUCHE;

  var
    err : boolean;

  begin
    x_curseur := x_curseur - 1;
    if (x_curseur = 0) then
      begin
        y_curseur := y_curseur - 1;
        if (y_curseur = 0) then
          begin
            err := DEF_HAUT_BAS;
            y_curseur := 1;
            if err then
              begin
                x_curseur := 1;
                ERREUR;
              end
            else
              x_curseur := length(buffer[prem_ligne + y_curseur - 1]) + 1;
            end
          else
            x_curseur := length(buffer[prem_ligne + y_curseur - 1]) + 1;
          end;
        end;
      end;

{-----}
{ DEP_DROITE pour la flèche à droite }
{-----}

procedure DEP_DROITE;

  var
    l : integer;
    err : boolean;

  begin
    l := prem_ligne + y_curseur - 1;
    if x_curseur > length(buffer[l]) then
      begin

```

```

if (y_curseur > (nb_occup - prem_ligne + 1)) then
  ERREUR
else
  begin
    y_curseur := y_curseur + 1;
    if y_curseur > nb_ligne then
      begin
        err := DEF_BAS_HAUT;
        y_curseur := nb_ligne;
        if err then
          ERREUR
        else
          x_curseur := 1;
        end
      end
    else
      x_curseur := 1;
    end;
  end
else
  x_curseur := x_curseur + 1
end;
{-----}
{ MONTE pour la flèche vers le haut }
{-----}

procedure MONTE;

var
  l, x0 : integer;
  err : boolean;

begin
  y_curseur := y_curseur - 1;
  if y_curseur = 0 then
    begin
      err := DEF_HAUT_BAS;
      y_curseur := 1;
      if err then
        ERREUR;
      end;
    end
  l := prem_ligne + y_curseur - 1;
  x0 := length(buffer[l]);
  if x_curseur > x0 then
    x_curseur := x0 + 1;
  end;
{-----}
{ DESCEND pour la flèche vers le bas }
{-----}

procedure DESCEND;

var
  err : boolean;
  l, x0 : integer;

```

```

begin
  if (y_curseur > (nb_occup - prem_ligne + 1)) then
    ERREUR
  else
    begin
      y_curseur := y_curseur + 1;
      if (y_curseur > nb_ligne) then
        begin
          err := DEF_BAS_HAUT;
          y_curseur := nb_ligne;
          if err then
            ERREUR;
          end;
        l := prem_ligne + y_curseur - 1;
        x0 := length(buffer[l]);
        if x_curseur > x0 then
          x_curseur := x0 + 1;
        end;
      end;
    end;
  end;

```

```

{-----}
{ DES_LIGNE efface la ligne où se trouve le curseur }
{-----}

```

```

procedure DES_LIGNE;

```

```

  var
    l, i : integer;

```

```

begin
  if nb_occup = 0 then
    ERREUR
  else
    begin
      x_curseur := 1;
      l := prem_ligne + y_curseur - 1;
      for i := 1 to (nb_occup - 1) do
        buffer[i] := buffer[i + 1];
      buffer[nb_occup] := "";
      nb_occup := nb_occup - 1;
      for i := y_curseur to nb_ligne do
        begin
          POSITION_ECRAN(1, i);
          ECRIT(buffer[i + prem_ligne - 1]);
        end;
      end;
    end;
  end;

```

```

{-----}
{ DES_CAR efface le caractère à gauche du curseur }
{-----}

```

```

procedure DES_CAR;

```

```

  var
    l, ll : integer;
    s1, s2 : ligne;

```

```

begin
  l := prem_ligne + y_curseur - 1;
  ll := length(buffer[l]);
  if x_curseur = 1 then
    if ll = 0 then
      DES_LIGNE { on efface la ligne vide }
    else
      ERREUR
    else
      begin
        if (x_curseur > ll) then
          buffer[l] := copy(buffer[l], 1, ll - 1)
        else
          begin
            s1 := copy(buffer[l], 1, x_curseur - 2);
            s2 := copy(buffer[l], x_curseur, ll - x_curseur + 1);
            buffer[l] := concat(s1, s2);
          end;
          x_curseur := x_curseur - 1;
          POSITION_ECRAN(1, y_curseur);
          ECRIT(buffer[l]);
        end;
      end;
end;
}-----}
{ NOUVEL_LIGNE permet d'ajouter une ligne au texte }
}-----}
procedure NOUVEL_LIGNE;

var
  l, ll, i : integer;

begin
  if nb_occup = max_buffer then
    ERREUR
  else
    begin
      l := prem_ligne + y_curseur - 1;

      ll := length(buffer[l]);
      buffer[l + 1] := copy(buffer[l], x_curseur, ll - x_curseur + 1);
      buffer[l] := copy(buffer[l], 1, x_curseur - 1);
      for i := nb_occup downto l + 1 do
        buffer[i + 1] := buffer[i];
      nb_occup := nb_occup + 1;
      for i := y_curseur to nb_ligne do
        begin
          POSITION_ECRAN(1, i);
          ECRIT(buffer[i + prem_ligne - 1]);
        end;
        x_curseur := 1;
        DESCEND; { on laisse la routine DESCEND effectuer le travail }
      end;
    end;
end;

```

```

{-----}
{ INTERACTIVE procedure principal gerant les fonctions de base }
{-----}

procedure INTERACTIVE;

  var
    ch : char;

begin
  ch := LECTURE;
  if (ord(ch) > 31) and (ord(ch) < 127) then
    INSERER(ch)
  else
    case ord(ch) of
      gauche :
        DEP_GAUCHE;
      droite :
        DEP_DROITE;
      haut :
        MONTE;
      bas :
        DESCEND;
      cr :
        NOUVEL_LIGNE;
      destruc :
        DES_CAR;
      destruc_ligne :
        DES_LIGNE;
      termi :
        fin := true;

      { ajouter ici la gestion des modes insertion/surimpression }
      { et ou d'autres fonctions plus complexes }

      otherwise
        ERREUR; { pour tous les autres codes de contrôle }
    end;
  end;

  { ***** }
  { ***** DEBUT DU PROGRAMME PRINCIPAL ***** }
  { ***** }

begin
  INITIALISER;

  { maintenant on boucle jusqu'à la fin demandée par l'utilisateur }

  repeat
    INTERACTIVE;
  until fin;

  { on pourra ajouter ici les sauvegardes en sortie par exemple }

end.

```

DIALOGUE AVEC NOS LECTEURS

I. LES CARRÉS MAGIQUES

La nouvelle solution des carrés magiques a dû vous paraître bien hermétique dans notre précédent numéro. Nous vous prions de bien vouloir nous excuser mais une erreur a fait disparaître la solution proposée par M. Gueron rendant nos explications la concernant hors de propos. Aussi la voici :

```
10 CLS
20 REM programme realise par Jacques GUERON
30 REM BASIC standard + print using
40 REM permet d'obtenir l'ensemble
50 REM des 3600 carrés magiques d'ordre 5 tous distincts
60 REM les uns des autres
70 REM valeur magique=65
80 REM dans toute ligne,colonne,diagonale
90 REM y compris les diagonales brisees !.
100 INPUT " CARRE DE DEPART (Numero 1 a 3600) ";M:N=M-1
110 FOR A=0 TO 20 STEP 5
120 FOR B=0 TO 20 STEP 5:IF B=A THEN 470
130 FOR C=0 TO 20 STEP 5:IF (C=A) OR (C=B) THEN 460
140 FOR D=0 TO 20 STEP 5:IF (D=A) OR (D=B) OR (D=C) THEN 450
150 FOR E=0 TO 20 STEP 5:IF (E=A) OR (E=B) OR (E=C) OR (E=D) THEN 440
160 FOR X=1 TO 5
170 FOR Y=1 TO 5:IF Y=X THEN 420
180 FOR Z=1 TO 5:IF (Z=X) OR (Z=Y) THEN 410
190 FOR T=1 TO 5:IF (T=X) OR (T=Y) OR (T=Z) THEN 400
200 FOR U=1 TO 5:IF (U=X) OR (U=Y) OR (U=Z) OR (U=T) THEN 390
210 IF A+X < E+U AND A+X < D+Z AND A+X < C+Y AND B+Y < C+T THEN 220 ELSE 300
220 K=K+1
230 IF K < N+1 THEN 300
```

```

240 PRINT "      *";K;"*"
250 PRINT USING " ##"; A+X;B+Y;C+Z;D+T;E+U
260 PRINT USING " ##"; C+T;D+U;E+X;A+Y;B+Z
270 PRINT USING " ##"; E+Y;A+Z;B+T;C+U;D+X
280 PRINT USING " ##"; B+U;C+X;D+Y;E+Z;A+T
290 PRINT USING " ##"; D+Z;E+T;A+U;B+X;C+Y
300 IF X+A<U+E AND X+A<T+C AND X+A<Z+B AND Y+B<Z+D THEN 310 ELSE 390
310 K=K+1
320 IF K<N+1 THEN 390
330 PRINT "      **";K;"**"
340 PRINT USING " ##"; X+A;Y+B;Z+C;T+D;U+E
350 PRINT USING " ##"; Z+D;T+E;U+A;X+B;Y+C
360 PRINT USING " ##"; U+B;X+C;Y+D;Z+E;T+A
370 PRINT USING " ##"; Y+E;Z+A;T+B;U+C;X+D
380 PRINT USING " ##"; T+C;U+D;X+E;Y+A;Z+B
390 NEXT U
400 NEXT T
410 NEXT Z
420 NEXT Y
430 NEXT X
440 NEXT E
450 NEXT D
460 NEXT C
470 NEXT B
480 NEXT A

```

De plus, dans ces explications, il fallait lire non pas :

$$200 U = 15 - X - 4 - 2 - T$$

mais

$$U = 15 - X - Y - Z - T$$

II. LES CHIFFRES ET LES LETTRES

Voici maintenant la solution de notre problème de chiffres et de lettres. Nous nous contenterons de vous présenter cette fois-ci l'affichage des solutions sans chercher à les faire résoudre par l'ordinateur.

Program Chiffres_Lettres ;

Type

```

t_lettres = 'A'..'I' ;
t_chiffres = 1..9 ;
t_couple = record
    lettre : t_lettres ;
    chiffre : t_chiffres ;
end ;
t_solution = array [1..4] of t_couple ;

```

Var

```

solution      /
proposition   /
sol_modifiee  /
prop_modifiee : t_solution ;
coup          /
couples_bons  /

```

```

couples_mal_places ,
solitaire           :   INTEGER   ;
trouve              :   BOOLEAN   ;

Procedure Entree_solution ( OUT solution : t_solution ) ;

Var
  i : INTEGER ;

Begin
  for i := 1 to 4 do ( les valeurs possibles sont A a H et 1 a 8 )
  begin
    write ( ' Entrez le couple n.',i:1,' (lettre puis chiffre)' ) ;
    readln ( solution[i].lettre , solution[i].chiffre ) ;
  end ;
End ;

Function Bien_Places ( IN proposition   : t_solution ;
                      OUT sol_modifiee ,
                        prop_modifiee : t_solution ;
                      OUT solitaire    :   INTEGER ) : INTEGER ;

Var
  i , j , couple , simple , bon : INTEGER ;

Begin
  sol_modifiee := solution ;
  prop_modifiee := proposition ;
  couple := 0 ;
  simple := 0 ;
  for i := 1 to 4 do
  begin
    bon := 0 ;
    if sol_modifiee[i].lettre = prop_modifiee[i].lettre then bon := bon + 1 ;
    if sol_modifiee[i].chiffre = prop_modifiee[i].chiffre then bon := bon + 1 ;
    case bon of
      1 : simple := simple + 1 ;
      2 : begin
          sol_modifiee[i].lettre      := 'I' ;
          sol_modifiee[i].chiffre     := 9 ;
          prop_modifiee[i].lettre     := 'I' ;
          prop_modifiee[i].chiffre    := 9 ;
          couple := couple + 1 ;
        end ;
    end ;
  end ;
  solitaire := simple ;
  Bien_places := couple ;
End ;

Function Mal_Places ( IN sol_modifiee
                    prop_modifiee : t_solution ) : INTEGER ;

Var
  i , j , couple : INTEGER ;

Begin
  couple := 0 ;

```

```

for i := 1 to 4 do
  begin
    j := 1 ;
    while ( solution[i].chiffre < 9 ) AND ( j <= 4 ) do
      begin
        if i <> j then
          if ( solution[i].lettre = proposition[j].lettre )
            AND ( solution[i].chiffre = proposition[j].chiffre ) then
            begin
              solution[i].lettre      := 'I' ;
              solution[i].chiffre     := 9 ;
              proposition[i].lettre   := 'I' ;
              proposition[i].chiffre  := 9 ;
              couple := couple + 1 ;
            end ;
          j := j + 1 ;
        end ;
      end ;
    Mal_places := couple ;
  End ;

```

{----- Programme Principal -----}

```

Begin
  Entree_solution (solution) ;
  coup := 0 ;
  repeat
  Entree_solution (proposition) ;
  coup := coup + 1 ;
  couples_bons := Bien_places( proposition , sol_modifiee ,
                              prop_modifiee , solitaire ) ;
  trouve := ( couples_bons = 4 ) ;
  if not trouve then couples_mal_places := Mal_places ( sol_modifiee , prop_modifiee ) ;
  writeln ( ' reponse ' , couples_bons:2 , solitaire:2 , couples_mal_places:2 ) ;
  until trouve ;
  writeln('vous avez trouve la solution en ',coup:2,' coups') ;
End

```

Ce programme laisse 8 choix possibles pour les lettres et autant pour les chiffres. La lettre I et le chiffre 9 sont réservés à la programmation. Le programmeur donne une combinaison solution qui est un tableau de 4 couples lettres-chiffres. Ensuite, le joueur propose une première hypothèse qui est également un tableau de 4 couples. Le compteur des coups est incrémenté.

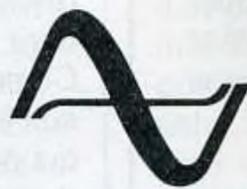
Solution	(A)	C	B	A
	(1)	2	2	1
Proposition	(A)	B	B	A
	(1)	2	3	4

- -

La première chose à faire est de compter le nombre de simplets (un demi-couple) et de couples bien placés. Dans ce but, pour chaque élément du tableau, on incrémente un compteur si la lettre ou le chiffre est correct. Si le compteur indique :

- 0 : rien n'est bon,
- 1 : un des deux seulement est bon, on ne touche rien,

C'EST ARRIVE DEMAIN



(en direct de notre envoyé permanent dans la Silicon Valley)

Les nouveaux Apple sont arrivés. Ils sont deux, pour simplifier. En fait, le SE est une sorte de Mac+, à peine amélioré, sinon la présence d'un «slot» permettant de connecter des cartes d'extensions. L'autre est plus intéressant, avec une architecture ouverte, un 68020, un coprocesseur mathématique et il gère la couleur. Un seul détail, il coûte dans les 4 000 dollars en version de base (24 000 francs), ce qui le mettra dans les 45 000 francs dans notre beau pays. La version couleur est, bien entendu, plus chère puisque un écran couleur est plus coûteux que son homologue noir et blanc. Et il faut naturellement une interface spéciale. L'ordinateur devient moins personnel chez Apple, mais il concurrence en fait le AT de chez IBM sans la moindre compatibilité. Une carte MS-DOS et

UNIX seront bientôt disponibles pour pallier cet inconvénient. Vous disiez familial. Puisque nous en sommes à parler de la société Apple, une information style Paris-Match. Mr. John Sculley, P.d.g. de la firme, semble gagner cette année la course la plus suivie de la région, celle du plus haut salaire. On prévoit dans les milieux informés et généralement autorisés (à moins que ce ne soit le contraire) une riposte foudroyante du P.d.g. de Chevron Oil, glorieux détenteur du titre depuis des lustres et totalement ridicule cette année, à plus de trois millions de dollars de Sculley (8 à 5). Notons que tout de même, le second empoche près de 3 milliards de centimes cette année, soit 250 000 000 de centimes par mois. Ambroise Roux, ex-patron le plus payé de France doit être vert de rage.

Un nouveau disque dur pour l'Amiga, d'une capacité de 20 Mb. Ceci peut sembler banal, mais le produit ne l'est pas. Il propose en effet, dans un même boîtier, une mémoire RAM de 1 Mb et deux slots d'extension supplémentaires. De plus, une carte horloge est fournie avec. Un système complet, qui est de plus performant en raison d'une carte d'interface travaillant en DMA (en accès direct sur le bus de donnée).

Enfin, une interface SCSI est incorporée, permettant de connecter d'autres périphériques. Tout au moins dans le principe car, pour l'instant, le seul grand utilisateur de ce type d'interface, Apple, a réussi à créer un faux standard SCSI, non compatible avec le vrai. Fermez le ban.

Les utilisateurs de Mac connaissent Switcher, programme permettant de passer d'un logiciel à un autre, sans ouvrir ni fermer quoi que ce soit. Il suffit de cliquer sur une flèche et vous êtes au cœur d'un autre programme. L'utilité de tels programmes, en dehors du simple confort d'utilisation, est évidemment dans la rapidité et la souplesse du transfert de données, de texte ou d'images, d'une application à une autre. Les possesseurs de PC, compatibles et d'Atari, peuvent maintenant obtenir les mêmes facilités grâce à des programmes nommés DOSamatic et Software Carousel. Le premier est un avatar d'un programme qui était distribué en «Shareware», c'est-à-dire gratuitement, à charge au possesseur qui l'utilise de verser quelques dollars (en général 20) à l'auteur. Il s'avère que la moralité des utilisateurs se perd et que le shareware ne paie plus. Donc, ce programme, revu et corrigé, est maintenant dans le circuit commercial. Software Carousel également est un très bon produit, souple et simple. Tous deux permettent d'engranger autant de programmes que la mémoire le tolère soit, pour une configuration de 1 Mb en moyenne, quatre programmes. Pas mal du tout quand on veut éviter une perte de temps importante.

Il existait jusqu'à présent un réseau de télévision par satellite permettant aux professionnels, revendeurs et grosses sociétés, de s'informer par des nouvelles rapidement traitées et commentées. Le prix était prohibitif et de plus, il fallait un réseau de réception pour chaque client. Une ruine. Malgré cela, MicroAge Computer Store, créateur de la formule, a décidé de jeter l'éponge, incapable qu'elle était de joindre les deux bouts par les abonnements. A la place, elle propose un magazine mensuel sur cassettes vidéo, d'une durée de deux ou trois heures. Le prix est toujours élevé, mais semble plus raisonnable pour les professionnels visés. Il ne reste donc plus sur le marché du magazine télé que la société The Computer Show, qui continue à proposer des programmes réguliers sur une fréquence particulière.

Commodore prépare une machine très puissante, l'Amiga 2000, qui permettra du vrai multi-tâche, avec une gestion multi-programmation, grâce à un 68020 à 14 MHz et une architecture ouverte. Sa compatibilité avec le 1000 est supposée parfaite. Il est livré avec un lecteur de disquette 5,5 pouces et permettra de travailler sous MS-DOS si on lui rajoute la carte ad-hoc. Sans cette carte, la configuration est de 1 500 dollars. Cette machine n'est pas encore officiellement annoncée (livraison prévue début avril), mais une revue, *Compute*, a dévoilé toutes les informations dont elle dispose à son sujet avant l'autorisation de Commodore. En réponse, le fabricant du 2000 poursuit la revue pour non respect d'une clause de secret qui ne fait pas l'objet d'accords écrits en général, les deux partis y gagnant à se respecter mutuellement. Cette fois-ci, ce ne semble plus être le cas et le procès promet d'être intéressant.

Commodore annonce une autre machine, bas de gamme à 500 \$, destinée au marché domestique. Ce sera une nouvelle tentative pour redresser les finances de la société qui, la première, s'est lancée dans l'ordinateur personnel.

Atari, malgré son dynamisme, continue à naviguer avec une gestion dans le rouge. Atari, avouons-le, ne s'y prend pas toujours au mieux de ses intérêts. Par exemple, après l'annonce d'un PC compatible performant et peu cher pour bientôt, on apprend que cette machine n'est toujours pas soumise à la commission chargée de tester sa conformité aux normes électriques américaines. C'est très inhabituel, cette conformité faisant le plus souvent l'objet des toutes premières démarches afin de pouvoir figer la machine sur le plan blindage et architecture générale, ce qui n'est pas rien. En conséquence, les actions d'Atari sont en chute libre, la société ayant annoncé que la future machine serait le déclic permettant le redressement cette année. Les bourdes commerciales ne sont pas l'apanage de nos compatriotes, c'est plutôt rassurant.

Ada, langage longtemps cité comme la réussite française aux U.S.A. et choisi par le Pentagone pour unique langage de ses ordinateurs, fait un flop complet. En effet, aucun système n'est sorti, en 5 ans, utilisant ce langage comme base. Moralité, les opposants commencent à parler d'un échec vrai et le Pentagone, devant la résistance de ses programmeurs à abandonner COBOL, envisage de faire machine arrière. Ce serait la fin de la standardisation espérée car cet organisme utilise à l'heure actuelle plus de 500 langages différents. Pas mal, non ?

Puisque nous en sommes à parler procès, IBM a gagné le sien contre Matsushita qui, apparemment copiait le BIOS (Basic input/output system) du PC-DOS, pour équiper ses machines au standard PC. Le coupable a accepté de payer aux alentours de 2 000 000 de dollars à IBM, ce qui n'est sans doute pas énorme en comparaison des sommes investies dans ce standard par les sociétés japonaises.

Une commission officielle du ministère du commerce américain a rendu un rapport sévère envers la politique commerciale japonaise, l'accusant notamment de ne pas suivre des engagements pris, dans le domaine de l'électronique, vis-à-vis des Américains, lors d'une réunion au sommet en juillet dernier. En conclusion, le rapport demande des sanctions économiques et douanières pour les fabricants japonais de composants. Il semble que le gouvernement soit décidé à suivre ce conseil. Si c'est le cas, cela représentera une première dans cette catégorie d'échanges où les U.S.A. étaient bénéficiaires jusqu'en 1985 mais où leur déficit se creuse maintenant à une vitesse qui semble préoccupante.

Les programmes de traductions apparaissent petit à petit. En voici un qui permet de convertir un texte du japonais en anglais et réciproquement. Il permet la traduction de 2 000 mots japonais en anglais par heure, dit la publicité. Il me semblait que le japonais n'était pas fait de mots, mais de pictogrammes, mais ils doivent savoir mieux que moi de quoi il s'agit. Un dictionnaire de plus de 100 000 mots est utilisé dans chaque langue pour la version de base et vingt dictionnaires spécialisés sont disponibles, couvrant des domaines allant de la médecine à l'informatique en passant par la biologie. Un programme complet, mais dont j'aimerais bien voir la qualité de la traduction, problème majeur dans un monde où le contre-sens et les faux-amis sont légions. Un groupe de mathématiciens anglais, dans les années 60, avait d'ailleurs conclu que de telles machines à traduire ne seraient jamais fiables à cause de la trop grande complexité des problèmes rencontrés. Prix du programme : 4,145 \$, ce qui n'est pas rien, puisque cela représente près de 25 000 francs.

Encore un procès, mais plus souriant celui-ci. Le distributeur d'un jeu d'aventures japonais a réussi à faire bloquer la parution d'un magazine qui révélait la solution du jeu et des indices pour y parvenir. Le fabricant justifie son intervention par le tort que de telles précisions pourraient porter à son jeu qui, comme la plupart des jeux de ce type, ne valent que par leur mystère et ne sont plus intéressants lorsque la passion de le trouver est dissipée.

Et encore un. Lotus, distributeur de 1-2-3, poursuit la société de Adam Osborne qui diffuse VP-Planner, un

programme japonais qui ressemble un peu trop à 1-2-3. Il semble en fait que VP-Planner soit beaucoup plus puissant, plus simple et surtout moins coûteux que 1-2-3. Mais, pour s'attaquer au marché de 1-2-3, il a dû reprendre ses commandes et l'allure de son interface utilisateur, afin de ne pas dérouter les clients potentiels. Moralité, VP-Planner, qui est sur le marché depuis plus de deux ans, commence à prendre de grosses parts de marché à 1-2-3, dont des positions dans les ministères et organismes officiels U.S. Lotus, devant le recul de son produit préfère bloquer un produit concurrent plutôt que faire progresser le sien. Il faut dire que 1-2-3 vaut 400 \$ et que VP-Planner, plus complet et rapide, en coûte 99. La concurrence est sans espoir pour Lotus. En fait, ce procès est très suivi ici, car si tout le monde est pour une certaine protection de l'idée des programmes, chacun pense que le copyright n'est pas défendable dans certains cas comme celui-ci où il n'y a pas plagiat mais amélioration d'un produit. Simplement, pour ne pas perdre des clients habitués à 1-2-3, VP-Planner *ressemble* à 1-2-3. Même les plus fanatiques des défenseurs du copyright espèrent que Lotus va perdre, car sinon cela empêchera des programmes comportant de très bonnes idées nouvelles de sortir et fera perdre beaucoup de temps à l'innovation informatique. Ainsi, cela va de soi, qu'à la baisse du prix des programmes.

Vous le savez sans doute, les USA ont imposé des taxes sur les produits japonais. La raison en est simple, les Japonais proposent, à des prix défiant toute concurrence, des produits dont le concept a été défini aux USA, en s'affranchissant ainsi des frais de recherche et de développements. Or, ces frais sont très élevés. Moralité, ce qui coûte dix dollars ici revient à 10 cents au Japon. La démesure est telle que toutes les sociétés électroniques américaines deviennent déficitaires et ne peuvent l'accepter. La réaction américaine est à la mesure, ou à la démesure, de son exaspération. C'est pourquoi voyant cela, les autorités japonaises ont tenté des ouvertures, mais il faut espérer qu'un véritable échange commercial soit possible, car sinon il n'est sans doute pas utopique de comparer la situation de l'Occident des années 2000 et celle de l'Allemagne des années 30 après l'humiliation de la grande guerre et la crise économique.

Et pourtant, la demande en composants a augmenté de plus de 20 % cette année, ce qui devrait permettre à tout le monde de vivre, et même plus.

La société Microsoft, reconnue entre autres pour sa possibilité à sortir des produits sans « bugs », va sans doute perdre un peu de son aura. En effet, la version de Word qui vient de sortir, la 3.0 pour le MAC, est tellement buggée dans des commandes importantes, que la vente est stoppée, quoi qu'en disent les porte-parole de la firme. Cela est d'autant plus grave pour Microsoft que de très gros investissements ont été engagés

dans ce programme et que la concurrence annonce déjà beaucoup mieux. Writer+ (cocorico, c'est français) et FullWrite proposent tout ce qui est dans Word, plus bien d'autres commandes et de la simplicité. Cela fait beaucoup. Cela dit, ces problèmes ne devraient pas affecter la version française qui sortira bien plus tard que la version US corrigée. Cela n'inquiète pas Microsoft qui vient d'entrer dans le monde de la grande informatique. En effet, IBM a annoncé que Microsoft serait le fournisseur officiel de son système d'exploitation pour les nouveaux PS/2 et que ce système sera d'une très grande puissance. Par cette annonce, IBM a voulu visiblement mettre Microsoft en orbite et éliminer les autres concurrents potentiels. Gageons que cette annonce atteindra cet objectif.

Une grande société de Miami est poursuivie pour un motif qui semble devenir une raison fréquente de procès. Elle est poursuivie pour exportation de PC compatibles vers Cuba. Elle est passible de 10 ans d'emprisonnement pour ses dirigeants, et de 50 000 \$ d'amende par membre de son conseil d'administration, pour intelligence avec l'ennemi, commerce avec l'ennemi. Cela pour une livraison de plus de 1 million de dollars qui avait transité par Panama. Pour leur défense, les patrons de cette société ont plaidé qu'ils ne faisaient rien de plus grave que l'état équatorien qui fournit chaque année une tonne de compatibles PC à l'URSS. Avec ce pays, on ne compte pas en unité, mais en tonnes, même pour des ordinateurs. Ceci dit, vous venez sans doute d'apprendre que l'Equateur est un producteur de matériel informatique.

La revue *Fortune*, spécialiste des études des revenus des entreprises, vient de publier son bilan annuel pour 1986. IBM est respectivement 1, 4, 2 et 15 dans les domaines de la valeur globale de ses actions, en valeurs de ses ventes, en profits et en argent en caisse. Le relatif retard de ce dernier rang est dû au fait que les banques ont traditionnellement une masse critique supérieure. Cela dit, cela représente un bond en avant pour IBM qui se retrouve plus haut que jamais.

Le premier disque optique pour micro-ordinateur est sorti. Il est produit par Matsushita et sera livré avec les ordinateurs de la gamme PS/2 de chez IBM. C'est là un grand événement, à plus d'un titre. En effet, c'est le premier disque optique disponible. Il sera donc livré en série à la demande et il fait l'objet d'une coopération US-Japon. C'est donc vraiment un pas important pour beaucoup de gens.

Soyons heureux, les problèmes de poursuites pour copies existent aussi aux pays du plagiat. NEC, spécialiste de la copie de puces, poursuit Seiko-Epson pour copie de son ordinateur, le NEC PC-9801. Cela

représente une première qui est suivie de très près par d'autres firmes japonaises qui ont du matériel compatible prêt à sortir. Si le procès se passe mal pour Seiko, les autres firmes devront arrêter la production de compatibles. Au Japon, le NEC est un standard comme le PC dans le reste du monde.

Un signe important de la vitalité du marché des ordinateurs dans la vallée : le prix des matériels d'occasion a augmenté de 20 % en moyenne l'an dernier, ce qui prouve que la demande surpasse l'offre de neuf. Les vedettes sont Apple et Compaq, dont les prix de l'occasion ont augmenté de plus de 60 % en moyenne.

Un nouveau programme, pour l'instant pour IBM seulement, permet de reconstruire des données détruites sur des disques durs et souples. Selon sa publicité, il fait usage de techniques évoluées d'intelligence artificielle et possède plusieurs cordes à son arc puisqu'il lui est possible de détecter, réparer, reconstruire et même prévenir dans une certaine mesure. Pour cela, le programme teste toutes les pistes d'un disque et déplace des données si une piste lui semble poser le moindre problème. N'étant pas un spécialiste, je ne vous donnerai pas de détails supplémentaires mais les essais semblent très prometteurs. Vivement une version sur les plus petits micros.

Je viens de lire une publicité d'un nouveau genre, pour les revues de micro-informatique en tous cas. Un groupe d'avocats propose ses services pour tous les types de plagiat, copies et autres infractions aux lois sur le copyright. Le marché est porteur, c'est certain, mais malgré tout c'est la première fois que ce type d'annonce paraît, à ma connaissance. Il faut de tout pour faire évoluer le monde de la micro, mais ce n'est sans doute pas le plus sympathique.

Borland, toujours dynamique, vient de faire deux annonces intéressantes. Tout d'abord, la création d'une revue destinée aux utilisateurs de TurboPascal, qui sera mensuelle et permettra aux abonnés de se maintenir au courant des problèmes, des solutions sophistiquées développées par certains, etc. L'autre annonce est pour le développement, avec Willey and Sons, d'une gamme de livres destinés aux écoliers et aux lycéens (américains pour l'instant), pour mieux approcher l'utilisation des langages de chez Borland. Cela devrait inciter les écoles et lycées à utiliser les Turbo (Pascal, Basic et Prolog). Bravo encore à cette société vraiment dynamique.

Aldus, qui proposera sous peu ses nouveaux Page-Maker, pour Mac et PC, a annoncé que ces deux programmes sont totalement similaires, avec plus de 80 % du code semblable et que les fichiers sont au même

format sur les deux machines. C'est sans doute le premier pas vers un nouveau type de compatibilité, moins large mais très importante, celle qui permet de récupérer des fichiers sur n'importe quel ordinateur venant de n'importe où. Ce sera alors une nouvelle vision de la compatibilité, entièrement «soft» et très importante pour les professionnels utilisant des données dont la provenance ne peut pas être toujours connue ou au bon format.

Vous vous souvenez sans doute de ce que j'écrivais voilà quelques mois sur le marché du disque dur. Je vous disais alors que les prix, en chute libre, témoignaient de la volonté de certains d'en découdre et d'éliminer la concurrence. Ceci est devenu une réalité puisque sur les 80 sociétés qui en proposaient l'an dernier, déjà 19 ont fait faillite, dont Micah et peut-être bientôt Corvus. Ce ne sont pas des petites compagnies et ce n'en est que plus significatif. Les compagnies qui survivent ont en général une structure soit très dynamique (SuperMac) soit un personnel à peu près inexistant et une vente par correspondance uniquement. Ces deux phénomènes semblent être la clé de la longévité dans un domaine où la chasse est ouverte.

Jean-Louis Gassée, ancien patron d'Apple France et maintenant numéro 3 d'Apple USA, est une très grande vedette depuis la sortie de l'Open Mac, le Mac II. En effet, la rumeur s'est largement répandue que c'est lui qui l'a voulu. De plus, il a mis à profit cette sortie pour publier un nouveau livre, «La troisième pomme», qui fait état d'un certain nombre de points de «philosophie» informatique, d'Apple, etc. Comme la ruée des Français dans la région devient un phénomène de société, cela fait un ensemble qui porte Gassée à une célébrité pas entièrement innocente ou involontaire.

Une nouvelle qui fera grand plaisir aux utilisateurs de dinosaures : un nouveau WordStar pour CP/M est sorti. Cela fait plaisir de savoir que ce système d'exploitation n'est pas entièrement abandonné. En effet, il semblait que CP/M soit définitivement passé à l'état d'animal de musée ainsi que WordStar d'ailleurs, même si ce dernier reste la référence du moment dans le domaine des traitements de textes sur PC et compatibles.

Le Commodore Show a tenu ses promesses cette année, apportant son lot de nouveautés. Les conférenciers furent passionnants, de Jim Butterfield à Brian Dougherty en passant par le créateur de Aztec C, Jim Goodnow. Parmi les produits attendus depuis tant de temps qu'on ne les espérait plus, citons WordPerfect, le traitement de texte qui fait la pluie et le beau temps sur les PC en ce moment, en version Amiga, avec l'interface Intuition qui ressemble un peu à celle du Mac. Il y a tout dans ce traitement de texte, du vérifica-

teur d'orthographe au colonnage, avec compatibilité avec les fichiers de WordPerfect sur IBM. Un bien beau produit. Pour ceux qui en veulent plus, The Calligrapher permet de créer des polices de caractères pour avoir la police qu'il vous faut quand il vous la faut. Vous pouvez utiliser 16 couleurs dont la taille maximale est de 160 par 256 pixels. Le Kerning est réglable, il est possible de faire tourner chaque caractère, de le renverser, de le redimensionner, de le remplir avec un dessin personnel, etc. Un vrai produit avec une présentation professionnelle et un prix d'amateur (29 \$). Je ne peux que le recommander sans restrictions si vous êtes l'heureux possesseur d'un Amiga. Bien d'autres programmes étaient donnés, ainsi que des cartes d'extensions, dont une carte 68020/68881, tournant à 16 MHz, soit deux fois la vitesse du Mac +. Un bijou qui était montré avec des jeux d'arcades, qui tournent ainsi à une vitesse telle que les plus aguerris ne pouvaient jouer à des niveaux élevés. Bien sûr, pour 1 300 \$, ce n'est pas à la portée de chaque bourse. Mais le bilan de cette exposition est que l'Amiga est bien vivant, ce qui fait très plaisir car il n'est pas juste que cet ordinateur, meilleur et moins cher que la concurrence, n'ait pas de succès. Longue vie à l'Amiga.

Pour ne pas être en reste, son concurrent de toujours, l'Atari ST avait aussi sa fête. Une fête gâchée par la décision d'Atari de ne pas y venir pour de médiocres raisons commerciales. Parmi les vedettes de ce salon, WordPerfect. Pour des détails, remonter de vingt lignes car c'est la même version qui est adaptée ici. La base, dans les deux cas, est la version 4.1 pour PC. Ici, c'est GEM qui sert d'interface et plus que jamais l'expression «elle ressemble à celle du Mac» est de circonstance. Nous avons déjà parlé de GEM, donc vous reporter à un précédent numéro pour cela. Antic proposait un produit très drôle, de visualisation 3D, avec des lunettes à verres rouge et bleu. Le résultat est impressionnant mais il est très difficile de concevoir des images. D'ailleurs, la démonstration est très explicite de par son temps d'affichage de chaque image. Mais le résultat est très propre, surtout en animation. Le clou de la fête fut malgré tout Data Pacific qui propose la cartouche permettant d'émuler un Mac. La nouvelle version fait la même chose, mais avec une émulation Mac +. Un très grand produit, à très petit prix (moins de 300 \$). Mais en plus, maintenant, la société propose un programme qui permet de transformer un lecteur d'Atari St en lecteur de Mac, assurant la lecture directe de programmes (même protégés) Mac sur l'Atari. Ceci marque la fin de la période de frustration pour les propriétaires de l'émulateur qui devait amener les programmes par voie de Modem ou par connexion directe d'un Mac et d'un Atari, empêchant totalement d'utiliser des programmes protégés. Je vous le disais, c'est un très bon produit. Pour le reste, beaucoup de jeux : Balance of Power et ChessMaster 2000, entre autres.

Branchez-vous !

Musique et ordinateur - MIDI Studio - Vidéo créative

lisez chaque mois...

Music Vidéo Systèmes

Musique
assistée par
ordinateur :
la scène

Systèmes M

ISSN 0298-377X

M 1616 n° 5 18 F

Mensuel novembre 86





LA BIBLIOTHEQUE TECHNIQUE DES EDITIONS FREQUENCES

offre des ouvrages techniques très actuels rédigés par des auteurs passionnés et impliqués complètement dans le sujet qu'ils traitent.

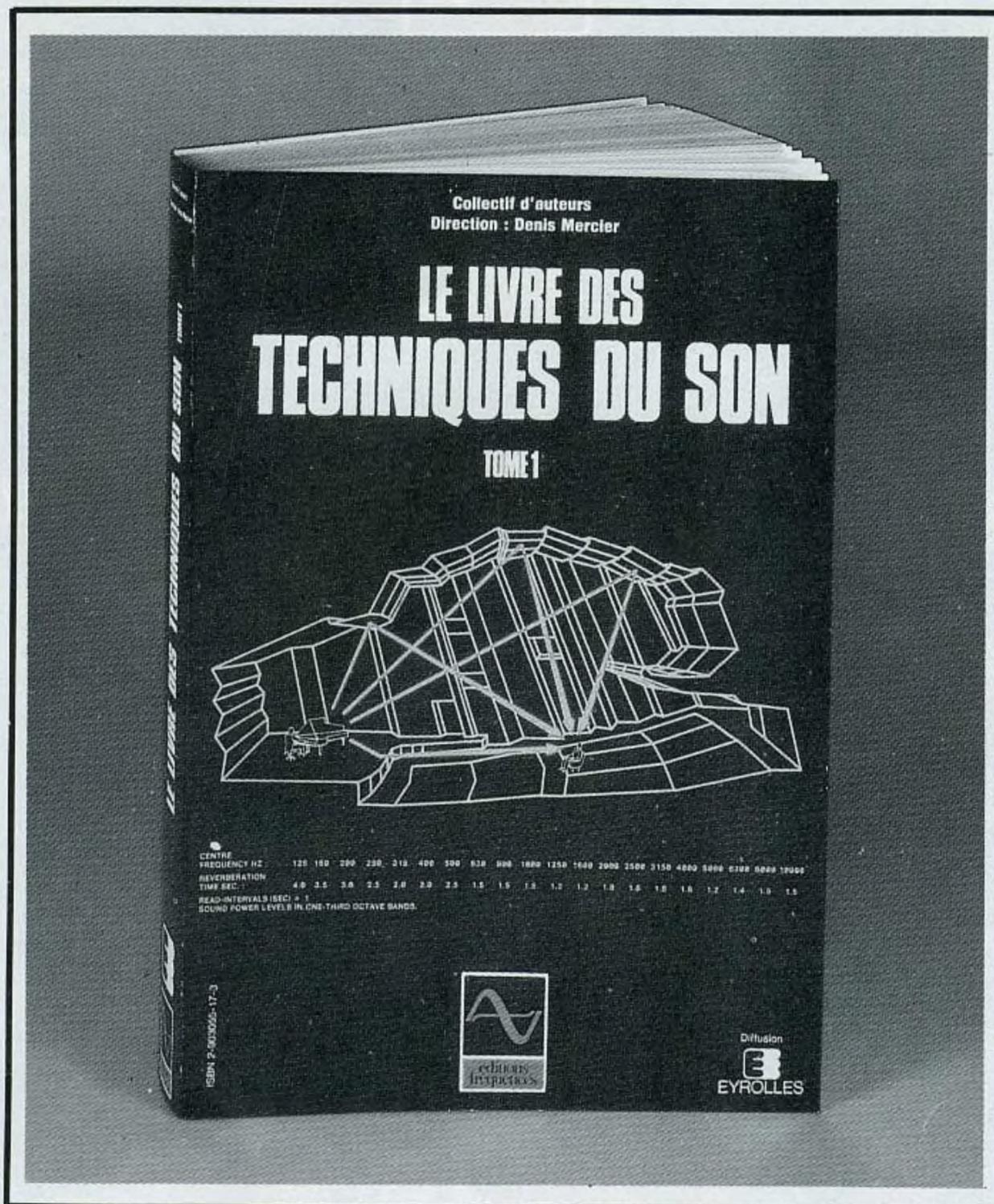
Vous trouverez soit des études approfondies sur les techniques ou les technologies de votre métier, soit des initiations théoriques et pratiques de techniques ou technologies que vous désirez approcher ou mieux cerner. Vous découvrirez au verso la description des ouvrages récemment parus ainsi que les commentaires sur les additifs d'éventuelles rééditions.

Les titres dont la parution est prochaine sont également mentionnés.

La page suivante comporte la liste complète des titres, leurs codes et leurs prix.



VIENT DE PARAÎTRE:



- 11 auteurs
 - 360 pages
 - 300 schémas et illustrations
- Prix : 350 F

Il y a bientôt trois ans démarrait ce travail de fond. Plus de vingt auteurs étaient sollicités pour concentrer en trois tomes les techniques du son. Le premier tome vient de paraître. Il traite de l'**Acoustique fondamentale**, des **Sources acoustiques**, de l'**Acoustique architecturale**, de la **Perception auditive**, des **Notions fondamentales de l'Electricité**, de l'**Enregistrement magnétique** ainsi que de la **Technologie audio-numérique**.

L'équipe des plus grands spécialistes actuels a été animée par Denis Mercier. Ensemble, ils ont mis sur pied un ouvrage actuellement unique au monde.

PROCHAINEMENT :

Collection jaune Etude autour du 6809 (constructions et logiciels) de Claude Vicidomini
L'image numérique de Jean-Marc Nasr
Le Basic structuré de Jean-François Coblenz
Divertissements en Basic de Franck Brown

Collection noire La création musicale par ordinateur de Frédéric Levé
Pratique de l'Amiga de Henri Cohen et François Dress

Collection noire (format 165 x 240)

Réf. Prix TTC

LES SYNTHETISEURS, UNE NOUVELLE LUTHERIE de Claude Gendre - 184 p. - Face au développement spectaculaire des synthétiseurs, grâce à l'électronique numérique, le besoin d'un ouvrage complet accessible et surtout bien informé des dernières ou futures techniques, se faisait ressentir. Le vœu est comblé, en 180 pages.

LES HAUT-PARLEURS de Jean Hiraga - 320 p. - Un gros volume qui connaît un succès constant : bien plus qu'un traité, il s'agit d'une véritable encyclopédie, alliant théorie et pratique, histoire en une mine inépuisable d'informations, reconnue dans le monde entier.

INTRODUCTION A L'AUDIO-NUMERIQUE de Jean-Pierre Picot - 160 p. - C'est le premier ouvrage paru en langue française traitant de l'audio numérique ; écrit par un professionnel, avec rigueur et simplicité, il explique brillamment les bases de cette technique : quantification, conversion, formats, codes d'erreurs.

L'OPTIMISATION DES HAUT-PARLEURS ET ENCEINTES ACOUSTIQUES de Charles-Henry Delaleu - 240 p. - Seconde édition améliorée d'un ouvrage fort attendu des passionnés d'électroacoustique. Ce livre permet aux amateurs et aux professionnels de se familiariser avec les rigoureuses techniques de modélisation des haut-parleurs et enceintes acoustiques et d'en mener à bien la réalisation.

LES MAGNETOPHONES de Claude Gendre - 160 p. - Pour tout savoir sur le magnétophone depuis l'avènement de cette mémoire des temps modernes, jusqu'aux enregistreurs numériques, en passant par la cassette «Les magnétophones» est un ouvrage pratique, complet, indispensable à l'amateur d'enregistrement magnétique.

LES MAGNETOSCOPES ET LA TELEVISION de Claude Gendre - 256 p. - Complément direct des «Magnétophones» «Les magnétoscopes et la télévision» débute par un bel historique de la télévision et la description des premiers magnétoscopes. La théorie et la pratique de la capture et de l'enregistrement moderne des images vidéo en sont la teneur essentielle.

L'ELECTRONIQUE DES MICRO-ORDINATEURS de Philippe Faugeras - 128 p. - Cet ouvrage est destiné aux électroniciens désireux d'aborder l'étude du «hard» des micro-ordinateurs. Cette étude s'articule autour du microprocesseur Z-80 très répandu, et en décrit les éléments périphériques : mémoire, clavier, écran, interfaces de toutes sortes.

PERIPHERIQUES : INTERFACES ET TECHNOLOGIE de Philippe Faugeras - 136 p. - Faisant suite à la parution de «L'électronique des micro-ordinateurs», cet ouvrage s'adresse aux électroniciens désireux de s'initier aux montages périphériques des micro-ordinateurs, interfaces en particulier, qui permettent la communication avec monde extérieur.

SELECTION DE L'AUDIOPHILE - TOME 1 : L'ELECTRONIQUE 256 p.

SELECTION DE L'AUDIOPHILE - TOME 2 : LES TRANSDUCTEURS 256 p.

Introuvable aujourd'hui, une sélection des meilleurs articles de la célèbre revue «L'Audiophile». Le tome 1 traite de l'électronique audio à tubes et transistors. Dans un esprit identique, le tome 2 traite du domaine passionnant que constituent les transducteurs en audio.

LE MINI STUDIO de Denis Fortier - 160 p. - Le monde de l'audio évolue... Un secteur d'activité entièrement neuf vient d'apparaître : les mini-studios. L'ouvrage de Denis Fortier, ingénieur du son, aborde le sujet de la manière la plus globale. Après les données physiques indispensables, le choix des maillons, la manière d'installer et d'exploiter.

LES TECHNIQUES DU SON Collectif d'auteurs sous la direction de Denis Mercier - 360 p. - Le Livre des Techniques du Son est le premier ouvrage interdisciplinaire en langue française s'adressant aux professionnels du son.

E 15	140 F
E 01	165 F
E 05	155 F
E 04	154 F
E 02	92 F
E 03	155 F
E 06	150 F
E 22	150 F
E 13	155 F
E 12	165 F
E 25	140 F
E 33	350 F

Collection rouge (format 135 x 210)

CONSEILS ET TOURS DE MAIN EN ELECTRONIQUE de Jean Hiraga 160 p. - Le «dernier coup de patte» apporté à un montage, celui qui fait la différence entre la réalisation approximative et le kit bien fini, ce savoir-faire s'acquiert au fil des ans... ou en parcourant «Conseils et tours de main en électronique».

LES LECTEURS DE COMPACT-DISCS 200 p. - Tout beau, tout nouveau, le lecteur laser. Qu'en est-il réellement ? Pour en savoir plus, un livre traitant du sujet s'imposait. «Les lecteurs de compact-discs» permet de faire son choix parmi 37 modèles testés, analysés, examinés et écoutés.

LEXIQUE DE L'ELECTRONIQUE ANGLAIS-FRANÇAIS de Jean Hiraga - 72 p. - Pour la première fois en électronique, un lexique anglais-français est présenté sous une forme pratique avec en plus des explications techniques, succinctes mais précises. Ce sont plus de 1 500 mots ou termes anglais qui n'auront plus de secret pour vous.

FILTRES ACTIFS ET PASSIFS POUR ENCEINTES ACOUSTIQUES de Charles-Henry Delaleu - 160 p. - Finis les calculs fastidieux et erronés ! Grâce à cet ouvrage, les concepteurs d'enceintes acoustiques gagneront un temps appréciable durant la phase d'étude et de mise au point : 120 abaques et tableaux pour tous types de filtres et d'impédances de HP !

17 MONTAGES ELECTRONIQUES de Bernard Duval - 128 p. - Voici enfin réunies dans un même ouvrage, dix-sept descriptions complètes et précises de montages électroniques simples. Il s'agit de réalisations à la portée de tous, dont bon nombre d'exemplaires fonctionnent régulièrement. Les schémas d'implantation et de circuits imprimés sont systématiquement publiés.

WEEK-END PHOTO de Philippe Folie-Dupart - 208 p. - Accessible à tous, «Week-end photo» permet de découvrir de façon simple les différents aspects de la photographie actuelle. Vous y trouverez les bases indispensables pour vous perfectionner, un guide de choix des appareils 24 x 36 et des illustrations abondamment commentées.

L 07	68 F
L 10	130 F
L 09	65 F
L 11	85 F
L 14	95 F
L 20	130 F

Collection jaune (format 210 x 270)

INITIATION A LA ROBOTIQUE 96 p. - Cet ouvrage eut un succès retentissant dès sa sortie. Bien plus qu'un cours d'initiation, il s'agit aussi du premier recueil d'informations données par les concepteurs, les utilisateurs et les fans de cybernétique enfin réunis !

INITIATION A LA MICRO-INFORMATIQUE COURS 1^{er} CYCLE - LE VOLUME 1 de Claude Polgar - 272 p.

INITIATION A LA MICRO-INFORMATIQUE COURS 1^{er} CYCLE - LE VOLUME 2 de Claude Polgar - 208 p.

INITIATION A LA MICRO-INFORMATIQUE COURS 1^{er} CYCLE - LE VOLUME 3 de Claude Polgar - 250 p.

Passé les premiers remous de la révolution que fut l'avènement de la micro-informatique, il fallut bien tenter d'en réunir les enseignements. Une lacune apparut : celle d'un ouvrage d'initiation à la programmation, universel et complet.

INITIATION A L'ELECTRONIQUE DIGITALE de Philippe Duquesne - 104 p. - Ce cours d'initiation à l'électronique digital est dû à Ph. Duquesne, chargé de cours de microprocesseurs au CNAM. L'objet de cet ouvrage est de présenter les opérateurs logiques et leurs associations. La technologie est évoquée, brièvement, elle aussi.

INITIATION AUX MICROPROCESSEURS de Philippe Duquesne - 136 p. - Du même auteur, Ph. Duquesne, on nous propose cette fois-ci, de pénétrer au cœur même de l'ordinateur, de comprendre le fonctionnement de l'élément vital qu'est le microprocesseur et enfin de maîtriser l'assembleur, langage du microprocesseur.

INITIATION TV : RECEPTION, PRATIQUE, MESURES, CIRCUITS de Roger-Charles Houzé - 136 p. - Issu d'un cours régulièrement remis à jour, ce livre permet à l'amateur comme au professionnel de se tenir au courant de l'état actuel de la technologie en télévision. De nombreux schémas explicatifs illustrent le contenu du livre.

INITIATION A LA MESURE ELECTRONIQUE de Michel Casabo - 120 p. - Il n'existait pas, jusqu'à présent, un ouvrage couvrant de manière générale mais précise, l'ensemble des problèmes relatifs à l'instrumentation et à la méthodologie du laboratoire électronique. C'est chose faite aujourd'hui avec ce volume récemment paru.

INITIATION AUX AMPLIS A TRANSISTORS de Gilles Le Doré - 96 p. - Après un bref historique du transistor, cet ouvrage traite essentiellement de la conception des amplificateurs modernes à transistors. La théorie est décrite de manière simple et abordable, illustrée d'exemples de réalisations commerciales. Le but du livre est de donner à chacun la possibilité de réaliser soi-même son amplificateur.

INITIATION AUX AMPLIS A TUBES de Jean Hiraga - 152 p. - Complémentaires des «Amplis à transistors» «les Amplis à tubes» sera certainement une petite encyclopédie sur ce sujet : historique, mais aussi polémique puisque les tubes sont encore d'actualité et parce que les arguments en faveur de cette technique et ses défenseurs sont encore nombreux.

INITIATION A L'ELECTRICITE ET A L'ELECTROTECHNIQUE de Roger Friederich - 110 p. - Vous trouverez aisément en librairie des ouvrages d'initiation à l'électronique ou aux techniques les plus avancées des circuits intégrés, etc. Mais si vous désirez une initiation aux bases de l'électricité et de l'électrotechnique sans vous en remettre à des ouvrages scolaires, alors vous ne trouverez pas !

INITIATION A LA VIDEO LEGERE - THEORIE ET PRATIQUE de Claude Gendre - 72 p. - Choix d'un standard ? Caméscopes VHS, VHS-C ou 8 mm ? Connexion ? Compatibilité ? Accessoires ? Montage ? Enfin... comment filmer ? Le nouveau livre de Claude Gendre répond à toutes ces questions. Cet ouvrage essentiellement pratique n'a pas d'équivalent en librairie aujourd'hui.

LES MONTAGES ELECTRONIQUES de Jean-Pierre Lemoine - 276 p. - Véritable encyclopédie. Plus de 1 000 dessins, 25 montages originaux.

LE TELEPHONE ET LES RADIOTELEPHONES de Roger-Charles Houzé - 96 p., 73 schémas.

LES BASES DE L'ELECTRONIQUE de Raymond Breton - 84 p., 162 schémas. Vous ne connaissez pas l'électronique : ce livre vous permet d'accéder aux bases nécessaires mais néanmoins d'atteindre un niveau vous permettant d'aborder des constructions de bases.

P 08	115 F
P 16	130 F
P 17	130 F
P 27	190
P 19	95 F
P 18	95 F
P 21	135 F
P 23	140 F
P 24	130 F
P 26	155 F
P 28	150 F
P 29	100 F
P 30	250 F
P 31	130 F
P 32	120 F

Diffusion auprès des libraires assurée exclusivement par les Editions Eyrolles.

Bon de commande à retourner aux Editions Fréquences 1, boulevard Ney 75018 Paris.

Je désire recevoir le(s) ouvrage(s) ci-dessous référencé(s) que je coche d'une croix :

E 01 <input type="checkbox"/>	E 02 <input type="checkbox"/>	E 03 <input type="checkbox"/>	E 04 <input type="checkbox"/>	E 05 <input type="checkbox"/>	E 06 <input type="checkbox"/>	L 07 <input type="checkbox"/>	P 08 <input type="checkbox"/>	L 09 <input type="checkbox"/>	L 10 <input type="checkbox"/>
L 11 <input type="checkbox"/>	E 12 <input type="checkbox"/>	E 13 <input type="checkbox"/>	L 14 <input type="checkbox"/>	E 15 <input type="checkbox"/>	P 16 <input type="checkbox"/>	P 17 <input type="checkbox"/>	P 18 <input type="checkbox"/>	P 19 <input type="checkbox"/>	L 20 <input type="checkbox"/>
P 21 <input type="checkbox"/>	E 22 <input type="checkbox"/>	P 23 <input type="checkbox"/>	P 24 <input type="checkbox"/>	E 25 <input type="checkbox"/>	P 26 <input type="checkbox"/>	P 27 <input type="checkbox"/>	P 28 <input type="checkbox"/>	P 29 <input type="checkbox"/>	P 30 <input type="checkbox"/>
P 31 <input type="checkbox"/>	P 32 <input type="checkbox"/>	E 33 <input type="checkbox"/>							

Frais de port : + 12 F par livre commandé, soit la somme totale ci-jointe, de Frs par CCP Chèque bancaire Mandat-lettre

Nom Prénom

Adresse

Ville Code Postal

LOGICIELS : DELITS ET PIRATAGE

Au cours du dernier SICOB, s'est tenue une réunion d'informaticiens du Barreau de Paris intitulée «Informatique et Télématique», le piratage sanctionné ?

Il ne fait aucun doute que le piratage en informatique a lieu de manière courante et parfois même industrielle (Sud-Est asiatique). Or la France possède un certain nombre de textes de lois à ce sujet. Il nous a semblé intéressant de retenir quelques points de l'intervention de Maître Henri Alterman, avocat à la Cour de Paris et membre du Conseil de l'Ordre.

1. Un des premiers délits en informatique, sans doute le moins connu, concerne l'utilisation de fichiers informatiques. Toute personne (ou société) détenant des fichiers est obligée de déclarer à la CNIL l'existence des informations qu'elle détient.

La loi du 6 janvier 1978 précise les délits et les contraventions :

- 1 - *absence de déclaration en cas de traitement d'informations nominatives (ou empli erroné de la norme simplifiée) par l'Organisme qui gère les fichiers : 6 mois à 3 ans de prison et ou 2 000 à 200 000 F d'amende.*
- 2 - *collecte et conservation illicite d'informations nominatives suivant les moyens employés (art. 25, 26, 27, 28, 30 ou 31) : 1 à 5 ans de prison et ou 20 000 à 200 000 F d'amende.*
- 3 - *divulgarion d'informations nominatives (délit volontaire ou involontaire) : 2 à 6 mois de prison ou 2 000 à 20 000 F d'amende.*
- 4 - *détournement de finalité.*
C'est le délit le plus grave : 1 à 5 ans de prison et 20 000 à 2 000 000 de francs d'amende.
- 5 - *contraventions de 5ème classe (décret du 23.12.1981)*
 - *entrave à l'action de la CNIL*
 - *non-respect de l'obligation d'information*
 - *obstacle au droit d'accès*
 - *obstacle au droit de rectification.*

2. Passons au délit le plus connu : le piratage. Le piratage de logiciel est une pratique très utilisée par de très nombreux possesseurs de micro-ordinateurs, pratique également courante chez les professionnels utilisant des systèmes beaucoup plus gros.

Après un certain flou, la France s'est dotée en 1985 d'une loi protégeant les concepteurs de logiciels. Il s'agit de la loi du 3 juillet 1985. En fait, dès 1957, il existait une réelle protection du droit d'auteur, mais il aura fallu attendre 1985 pour une application concrète des textes.

LOI DU 3 JUILLET 1985 (relative aux droits d'auteur et aux droits des artistes interprètes, des producteurs de phonogrammes et de vidéogrammes et des Entreprises de communication audiovisuelle).

(Titre V et 6).

Article 47 "par dérogation au 2°/ de l'art. 41 de la Loi n° 57-298 du 11 Mars 1957 précitée, toute reproduction autre que l'établissement d'une copie de sauvegarde par l'utilisateur ainsi que toute utilisation d'un logiciel non expressément autorisée par l'auteur ou ses ayants-droits, est passible des sanctions prévues par ladite Loi".

A la vérité, le texte renvoie également à la Loi de 57 pour les incriminations (art. 3). Les logiciels sont donc protégés par la Loi de 57 elle-même et pas seulement par la Loi du 3 Juillet 1985.

Il faut se reporter aux articles 425 et suivants du Code Pénal.

"Constitue une contrefaçon la reproduction du logiciel sans autorisation de l'auteur quel que soit le logiciel et quel que soit le support et même s'il s'agit d'une copie réservée à l'usage privé du copiste ; seule la copie de sauvegarde est autorisée (ça peut être deux copies...)."

"Constitue également une contrefaçon l'adaptation ou l'élaboration d'un programme dérivé sans l'accord de l'auteur".

"Constitue encore une contrefaçon l'utilisation non autorisée du programme."

"Constitue enfin une contrefaçon le débit, l'importation, l'exportation, la division, la diffusion d'un logiciel contrefait."

La sanction est prévue à l'article 425 du Code Pénal : c'est une amende de 6 000 F à 120 000 F et ou une peine de prison de 3 mois à deux ans ; ladite peine est doublée en cas de récidive et le Tribunal peut prononcer la fermeture de l'établissement exploité pour une durée pouvant aller jusqu'à 5 ans.

Enfin, même hors le cas de récidive, le Tribunal peut ordonner la confiscation des recettes procurées par la contrefaçon ainsi que la confiscation des objets contrefaisants et les différentes mesures de publicité.

3. Le troisième volet concerne l'accès à des systèmes sans autorisation.

Une des données les plus floues concerne le vol de temps machine. Il est difficile de parler de vol du fait qu'il n'y a aucune «soustraction». En effet, les informations restent dans la machine, il n'y a donc pas vol. Toutefois, lors d'une discussion récente avec des spécialistes, nous avons appris que, dans ce cas, il était toujours possible de détourner le problème en cas de procès. Le plaignant remplace la notion de vol de temps machine par un détournement d'énergie. En effet, dans ce cas, le «pirate» a utilisé abusivement l'électricité du propriétaire de la machine pour en extraire des informations. Toutefois, une affaire récente a inclus dans le procès la notion de vol d'informations.

Plusieurs autres méfaits peuvent être reprochés aux «pirates» :

- l'escroquerie
- le faux
- l'abus de confiance
- le recel
- etc.

CONCLUSION

On évalue à plus de trois milliards de francs (deux fois plus selon certains spécialistes) le montant des escroqueries à l'informatique en 1986. Aujourd'hui, même si certains aspects de la loi ne sont pas encore très précis, la notion de droit d'auteur est précisée sur le plan juridique par la loi du 3 juillet 1985 modifiant la loi du 11 mars 1957. Les logiciels sont donc désormais protégés.

Il y a eu, ces derniers mois, plusieurs descentes de police et plusieurs procès. Il est évident que chaque conducteur automobile connaît le code de la route. Il sait ce qu'il risque en roulant à 150 km/h sur une route normale. Les «pirates» en informatique doivent eux aussi faire attention, la chance a ses limites.

* Extraits de la conférence de Maître Alterman : réunion d'information du Barreau de Paris dans le cadre du 38^e SICOB.

Logiciels scientifiques pour électroniciens

Charles-Henry Delaleu

Depuis quelques numéros, nous avons abordé des logiciels généraux. Ce mois nous allons faire plaisir aux scientifiques et plus particulièrement aux électroniciens. En effet, les progiciels Hewlett-Packard leur sont particulièrement destinés. Il s'agit de trois programmes exécutable sur les séries 200 et 300 de la marque.

Les trois applications sont :

- un progiciel d'analyse de circuit alternatif,
- un progiciel d'analyse de fonctions de transfert,
- un progiciel d'analyse de signaux.

Ces programmes, écrits en Basic, ne sont pas protégés afin de subir d'éventuelles modifications. Dans certains cas, cette possibilité peut être très intéressante. Nous en donnerons un exemple pour le troisième des progiciels testés. Le Basic utilisé est le Basic Hewlett-Packard qui a peu de chance de pouvoir rapidement être traduit en un autre Basic style MS-GWBASIC. Le Basic HP est en effet un super-Basic qui reprend les bases du Basic, du Fortran, du PL/I et de l'Algol. Pour certaines applications répétitives, il sera possible de supprimer quelques sous-programmes et de chaîner ceux que l'on désire garder : ceci permettra de gagner en vitesse. Ces programmes sont remarquablement écrits. Ils tiennent compte de toutes les erreurs possibles, ainsi que des problèmes d'entrées-sorties. Bref, ce sont des progiciels PRO pour des PROS.

L'arrivée des micro-ordinateurs et des ordinateurs de table est en train de profondément modifier le travail des laboratoires et des bureaux d'études en électronique. En effet, le rapport qualité/prix de ces types d'appareils est de plus en plus surprenant. Si ces dernières années, seules les grandes et riches entreprises pouvaient investir dans des systèmes d'informatique techniques et scientifiques, il ne fait aucun doute qu'une démocratisation profonde se profile.

Généralement, les ordinateurs en technique n'ont pas besoin d'unités de stockage de masses importantes, les critères prépondérants sont : la vitesse de calcul, l'interfaçage multiple, le graphisme.

Dès lors, il est souhaitable d'utiliser pour ce genre de tâches des machines équipées de processeurs 16 bits, voire 32 bits. Or le prix d'un bon et vrai 16 bits est abordable aujourd'hui.

Dans un laboratoire, les techniciens sont souvent occupés par des tâches répétitives. Ces dernières peuvent être avantageusement effectuées par un calculateur. Il est possible, dans ces conditions, d'amortir très vite la machine.

LA SIMULATION DE CIRCUITS ELECTRONIQUES

Il existe deux manières de simuler des circuits électroniques :

- par équation mathématique,
- par création d'une série de nœuds simulés.

Pour ce faire, les deux logiciels utilisés sont :

- AC Circuit Analysis (HP 98825 A)
- Linear Systems Analysis (HP 98826 A)

AC CIRCUIT ANALYSIS

Ce titre pourrait se traduire en français par : analyse de circuit en courant alternatif. Il s'agit d'un programme pouvant analyser et simuler la réaction de circuits linéaires, en tenant compte de la tolérance des composants utilisés. Les composants utilisables sont : les résistances, les condensateurs, les inductances et les composants actifs (transistors, amplificateurs opérationnels, etc.). Il suffit d'assembler divers composants en leur donnant une valeur et en les plaçant sous forme de nœuds. Plusieurs nœuds peuvent être chaînés. L'ensemble s'échafaude sous la forme d'un tableau. On indique la nature et la valeur des composants ainsi que leur point de connexion. Chaque tableau peut être sauvegardé sur mémoire de masse.

Il est possible, une fois le tableau réalisé, de connaître les caractéristiques du circuit ainsi simulé. Pour ceci, deux choix sont prévus : soit transmettre en fréquence point par point ou plus simplement en fréquence glissante (linéaire ou logarithmique).

Pour chaque fréquence, il est possible de connaître la réponse en amplitude, en phase et le délai de chaque circuit analysé et simulé. L'impédance peut être traitée. Si une tolérance est demandée pour les composants utilisés, les résultats sont fournis en conséquence. Les valeurs minimales et maximales sont alors connues.

Si des analyses doivent être faites en continu, il est possible de s'en rapprocher par une valeur en fréquence infiniment faible. La valeur minimale acceptable est 10^{-18} Hz. La programmation de tableaux comprenant les différents composants est très facile à faire. Il suffit de répondre aux différentes questions posées par le programme.

Exemple :

Soit le filtre RC de la figure 1. Nous voulons connaître les caractéristiques de ce filtre passe-bas entre 100 kHz et 100 MHz. Nous sommes intéressés par la réponse en amplitude et la réponse en phase. Il suffit de choisir le mode d'affichage et de lancer la lecture du tableau.

L'ordre de programmation commence par les résistances, puis les condensateurs, les selfs, etc.

INPUT=NODE 1 OUTPUT=NODE 3 GROUND=NODE 0

COMPONENT	VALUE	PERCENT TOLERANCE	NODE FROM	NODE TO, (+), (-)
RESISTOR 1	90 ohm	0	1	3
RESISTOR 2	10 ohm	0	2	0
CAPACITOR 1	1.59 nF	0	3	2

Listing du circuit RC.

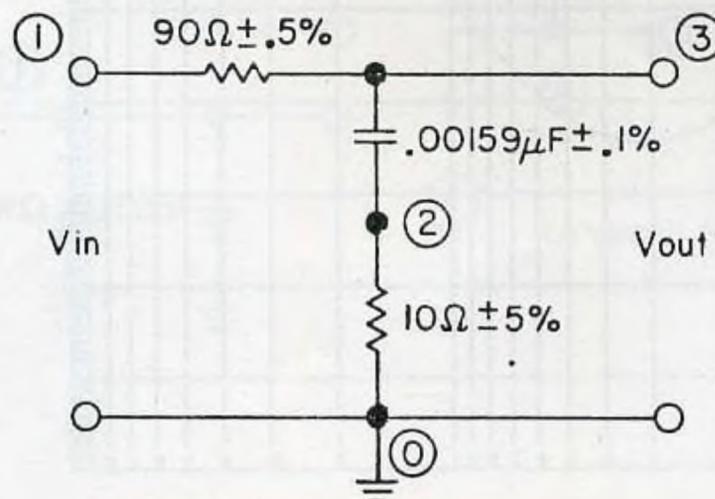


Fig. 1-A

Sur la figure 1 A, nous avons numéroté chaque nœud. Il y en a trois, le nœud 0 est, par convention, réservé à la masse. La première question à laquelle il convient de répondre est :

- Valeur de la résistance n° 1 ?

Nous tapons 90 puis la seconde question est :

- Position de la résistance n° 1 ?

Nous tapons 1,3 pour indiquer que la résistance se trouve entre les nœuds 1 et 3 de notre circuit. Puis c'est le tour de la résistance n° 2. Lorsque l'on ne désire plus entrer de résistance, il suffit de taper 0 comme valeur, et l'on passe aux condensateurs. Le même processus est répété pour chaque type de composants. Une fois le tableau enregistré, il est possible de le lister et de vérifier que tout est correct.

Sur la figure 1 B est présentée la courbe amplitude/fréquence et sur 1 C, la courbe phase/fréquence.

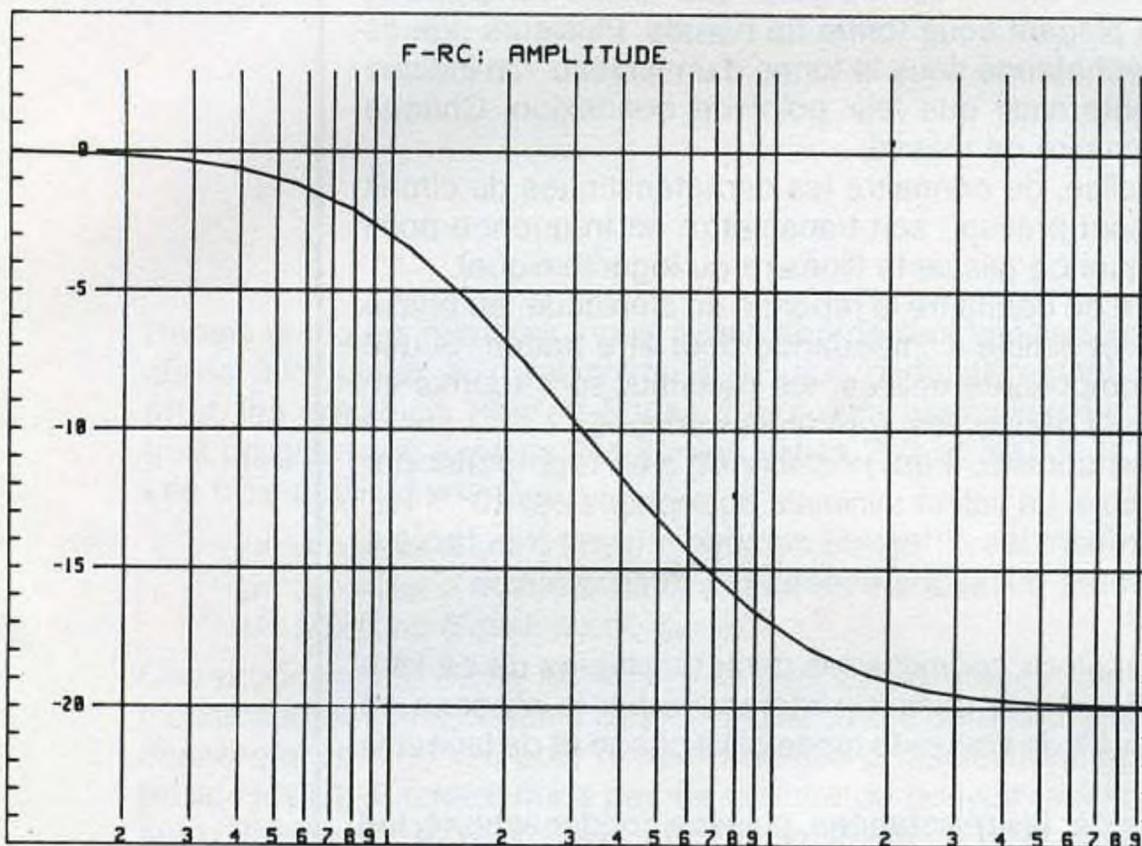


Fig. 1-B

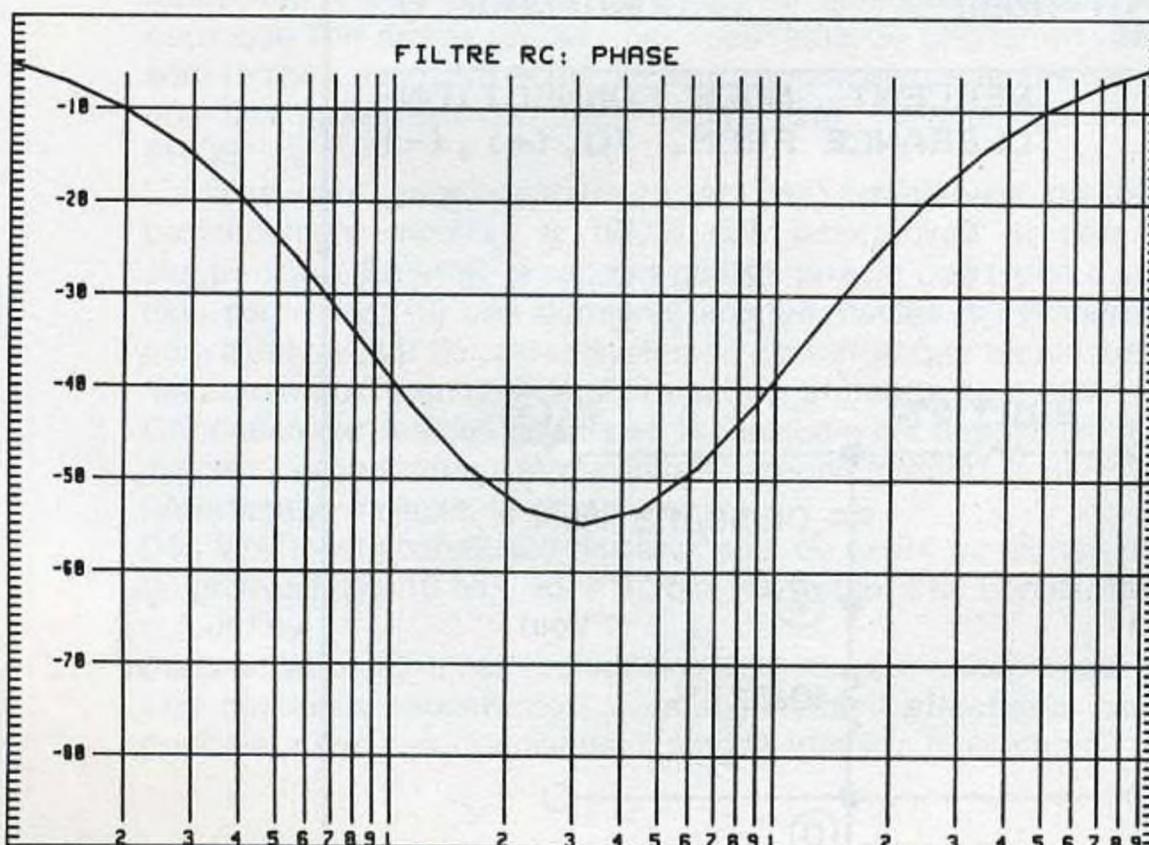


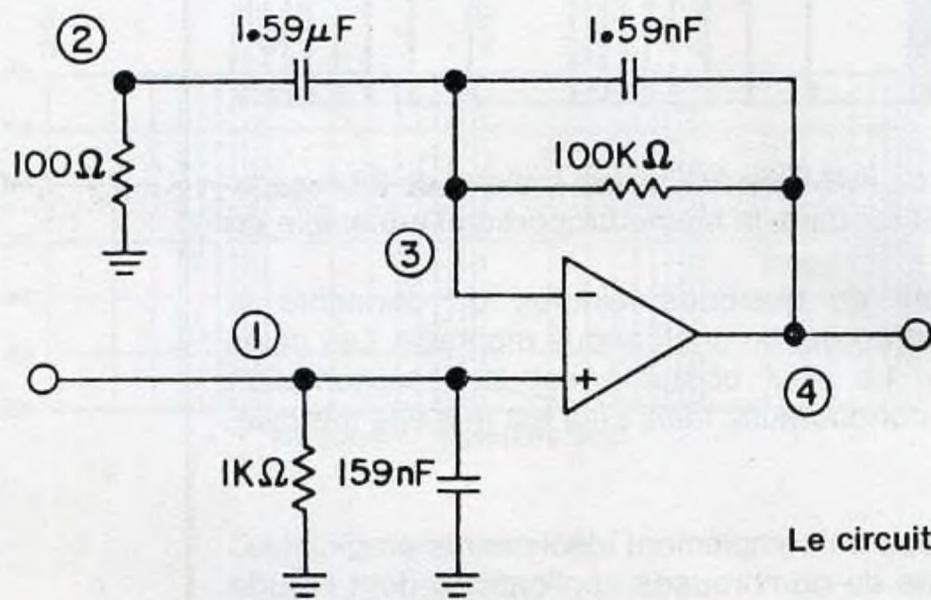
Fig. 1-C

Le second et dernier exemple concerne la simulation de la courbe amplitude/fréquence d'un filtre passe-bande réalisé autour d'un amplificateur opérationnel. Le circuit est présenté sur la figure 2 A et le résultat sur la figure 2 B.

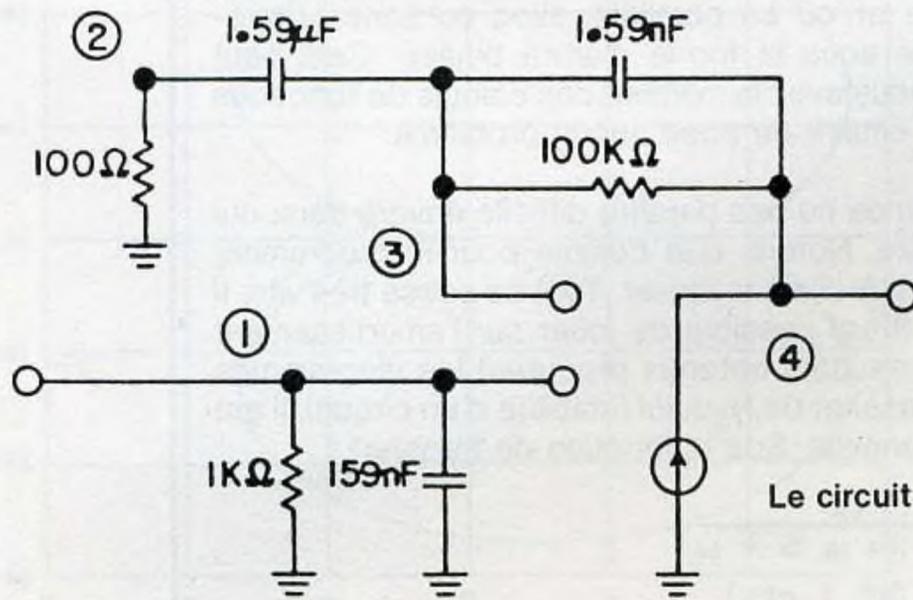
INPUT=NODE 1 OUTPUT=NODE 4 GROUND=NODE 0

COMPONENT	VALUE	PERCENT TOLERANCE	NODE FROM,	NODE TO,	CONNECTIONS (+), (-)
RESISTOR 1	100 ohm	20	2	0	
RESISTOR 2	100 kohm	20	3	4	
RESISTOR 3	1 kohm	20	1	0	
CAPACITOR 1	1.59 uF	20	2	3	
CAPACITOR 2	1.59 nF	20	3	4	
CAPACITOR 3	159 nF	20	1	0	
SOURCE 1	1 kmho	0	0	4	1 3

Listing du filtre actif.



Le circuit original.



Le circuit équivalent.

Fig. 2-A

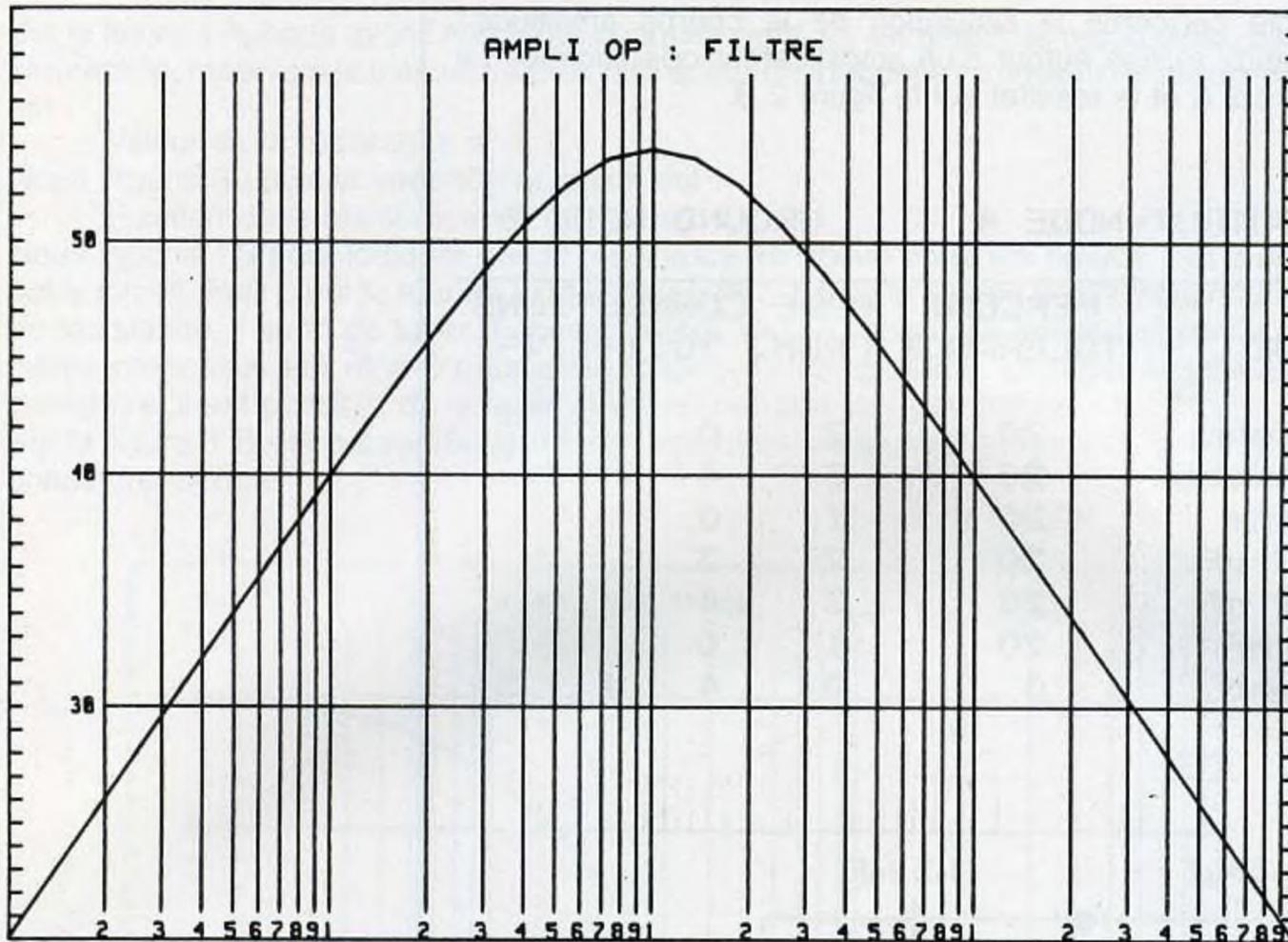


Fig. 2-B

Il est possible en version de base (256 ko de RAM) de réaliser un tableau de 99 nœuds. Chaque carte de 256 ko permet d'augmenter dans la même proportion l'ensemble du montage.

Grâce à ce programme, il est possible en quelques minutes de connaître le fonctionnement d'un circuit sans avoir à effectuer un quelconque montage. Les gains obtenus en temps sont considérables. Le seul obstacle est la représentation équivalente de circuits utilisant des semi-conducteurs, mais cela est très vite assimilé.

LINEAR SYSTEM ANALYSIS

Le progiciel Linear System Analysis peut être un complément idéal pour le progiciel AC Circuit Analysis. Il pourra aussi servir dans de nombreuses applications dont l'étude des asservissements par exemple. Ce package fait appel aux mathématiques. Grâce à cela, il est possible d'obtenir les caractéristiques techniques d'un circuit, d'un système, en partant d'une ou de plusieurs fonctions de transfert. Ces fonctions peuvent être simples, multiples, en série et/ou en parallèle, avec ou sans contre-réaction. Chaque cas est alors organisé sous la forme d'arbre binaire. Cela peut paraître compliqué pour certains, mais si vous avez la maîtrise des calculs de fonctions de transfert, le saut à l'utilisation d'arbre binaire ne pose aucun problème.

Exemple :

Nous n'entrerons pas dans les détails afin de ne pas paraître difficile envers ceux qui ont passé leur cours de maths à l'annexe. Notons que comme pour le programme précédent, l'ensemble des commandes est facile à maîtriser. Tout se passe très vite. Il suffit de suivre un ordre chronologique. Il est possible de jouer sur l'amortissement dans l'élaboration d'une simulation. Les résultats obtenus procurent les diagrammes de Bode (réponses en amplitude et en phase) et de Nyquist (stabilité d'un circuit). Il est possible de connaître la réponse impulsionnelle. Soit la fonction de transfert :

$$T_{(s)} = \frac{\omega_h^2}{s^2 + 2\varepsilon\omega_h s + \omega_h^2}$$

ε = coefficient de surtension (ex. : 0,25, 0,5, 1, etc.)

Prenons comme base $\omega_h = 4$ et $\varepsilon = 0,25$, nous avons donc :

$$T_{(s)} = \frac{16}{s^2 + 2s + 16}$$

Tout se passe très simplement. L'ordinateur commence par nous demander l'ordre du numérateur : dans notre cas, 0 ; puis sa valeur : ici 16. Puis on passe au dénominateur. L'ordre le plus élevé est 2 pour notre exemple. Une fois entré l'ordre maximum (2), l'ordinateur nous demande les valeurs de chaque ordre :

Pour l'ordre 0 \rightarrow 16

Pour l'ordre 1 \rightarrow 12

Pour l'ordre 2 \rightarrow 1.

Ici se termine la programmation de notre fonction de transfert. Comme vous pouvez le constater, c'est un jeu d'enfant. En figure 3 A sont présentés les diagrammes de Bode, en figure 3 B le diagramme de Nyquist, et en 3 C la réponse impulsionnelle, notre équation.

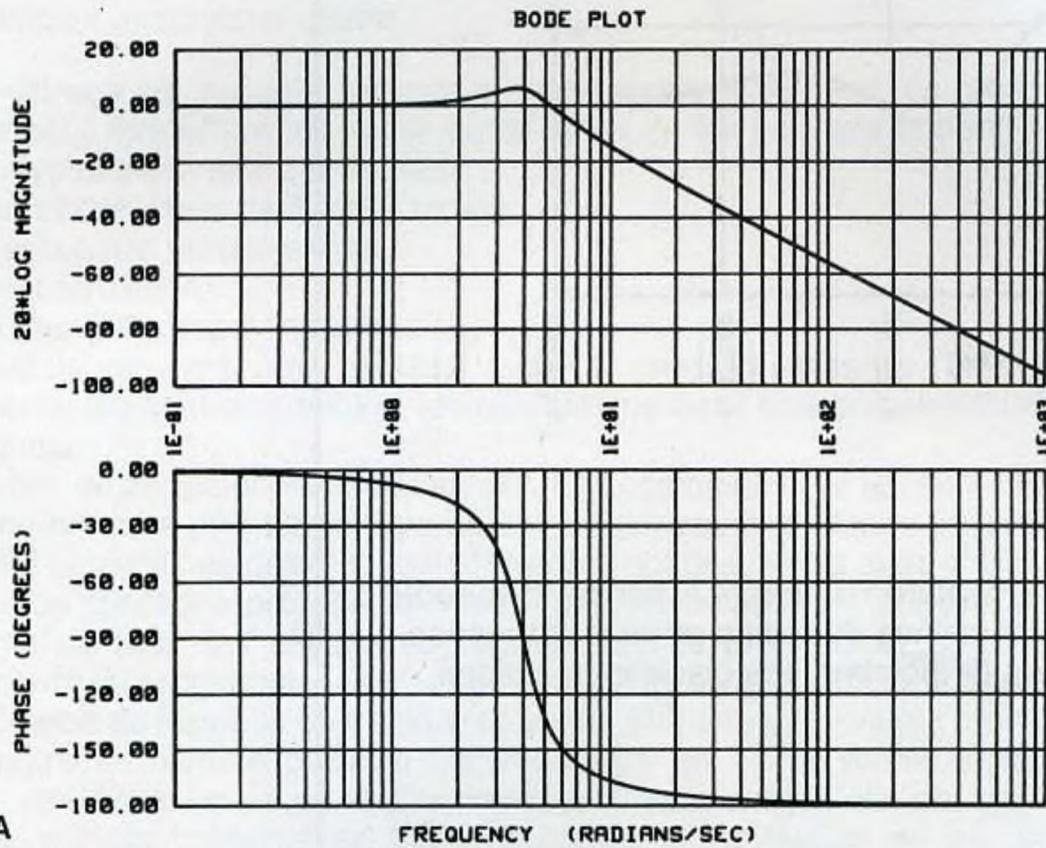


Fig. 3-A

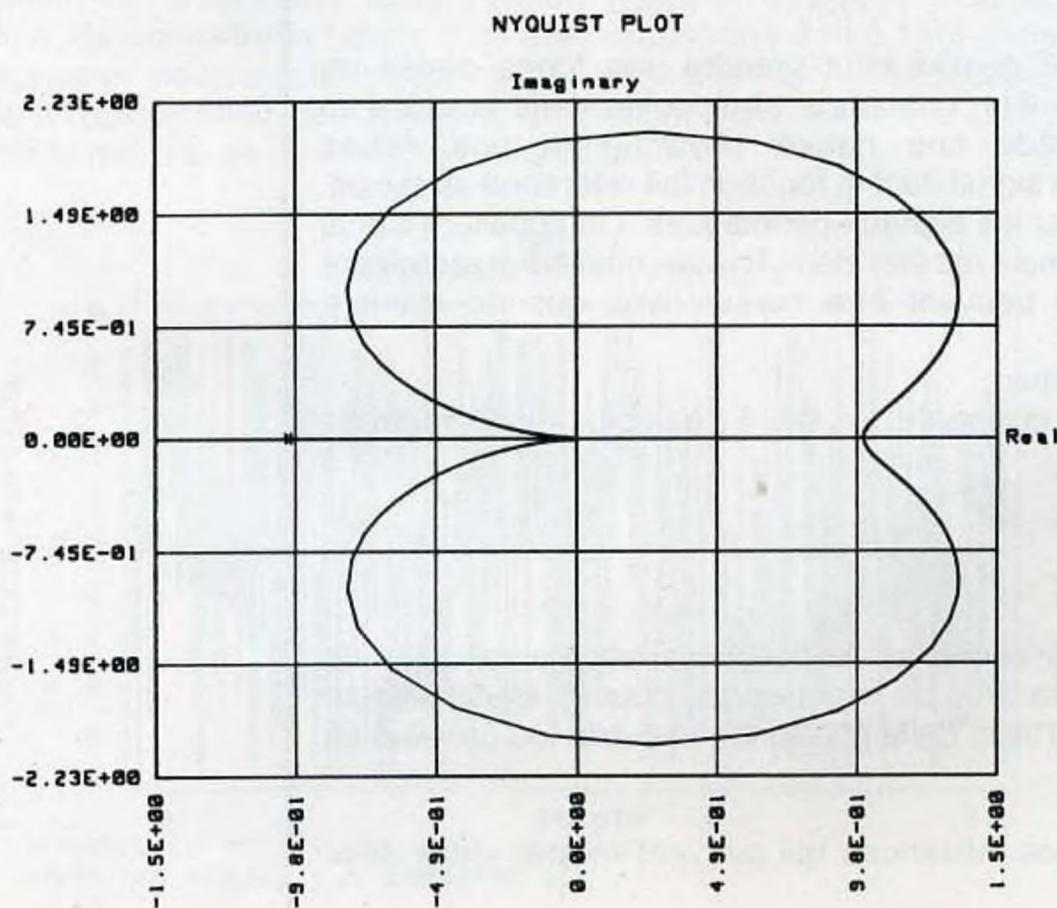


Fig. 3-B

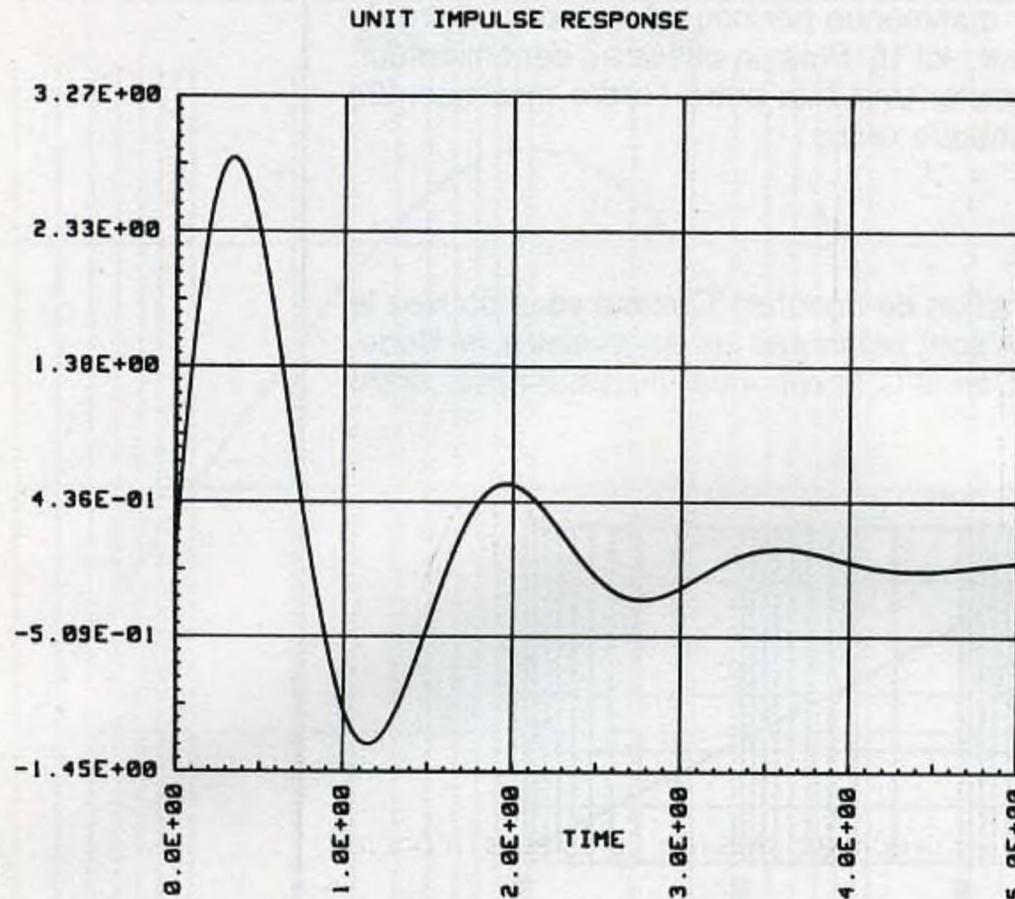


Fig. 3-C

ANALYSE DE SIGNAUX OU FFT

Le progiciel d'analyse de signaux Hewlett-Packard HP 98827 A permet d'accéder à toutes les propriétés de signaux dans les domaines du temps et de la fréquence.

Le traitement du signal peut être appliqué à de nombreuses disciplines, notons :

- l'électronique,
- l'acoustique,
- la mécanique, etc.

Les signaux :

Il existe deux types de signaux :

- les signaux certains,
- les signaux aléatoires.

Un signal transporte un message. Ce dernier peut prendre une forme codée ou modulée. Elle sera codée dans le cas d'un ordinateur et modulée dans le cas d'un circuit analogique. Un signal possède une nature physique et une nature mathématique. Le signal certain sera un signal dont la fonction fait référence au temps. Le meilleur exemple nous est donné par les signaux périodiques. On appellera signal aléatoire un signal que l'on ne peut jamais répéter deux fois de manière exactement identique. Ces signaux aléatoires ne peuvent être caractérisés que de manière statique.

Les termes utilisés en analyse de signaux :

L'analyse de signaux est devenue très puissante. En fait, il s'agit plus exactement de trois analyses principales :

- l'analyse de densité spectrale,
- la corrélation,
- la réponse en fréquence.

L'analyse de densité spectrale :

L'analyse de densité spectrale de puissance et de densité spectrale naturelle de puissance nous permet d'étudier la répartition de la puissance pour un signal unique. Pour deux canaux, nous aurons la répartition de la puissance mutuelle (ou croisée) en fonction de la fréquence.

Corrélation :

La fonction de cohérence démontre les influences qui peuvent exister entre deux signaux.

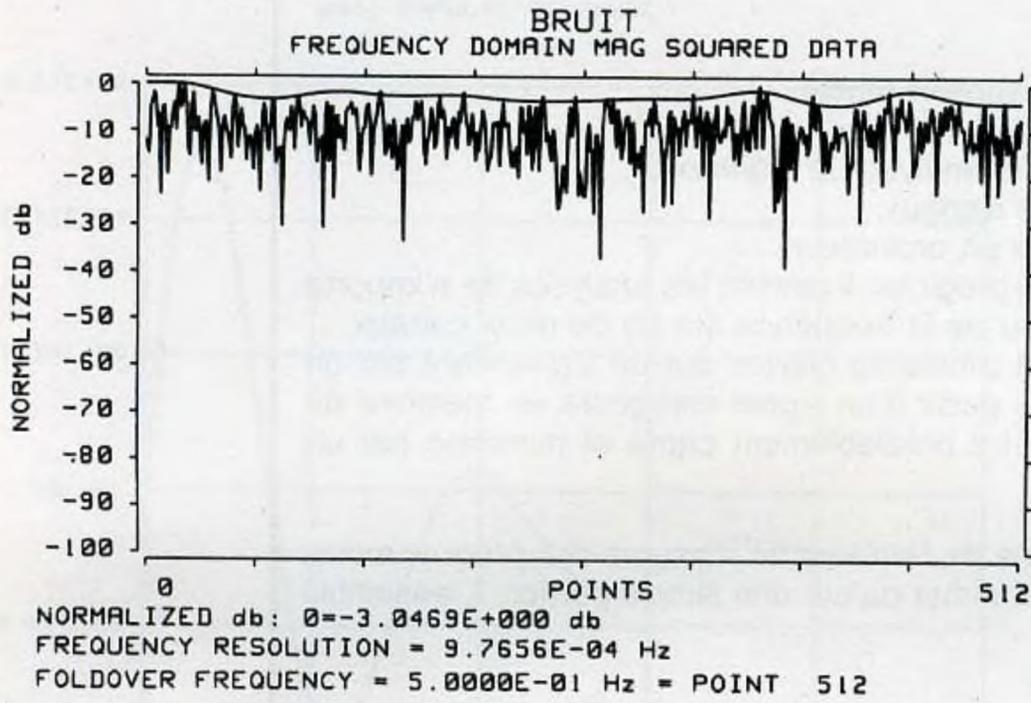


Fig. 4-B

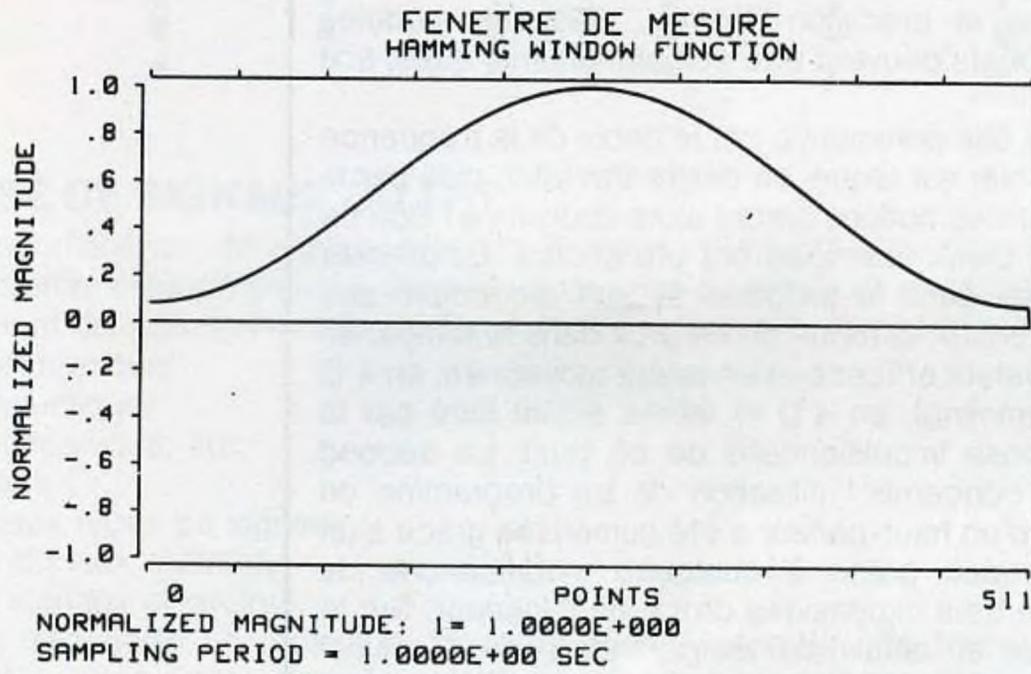


Fig. 4-C

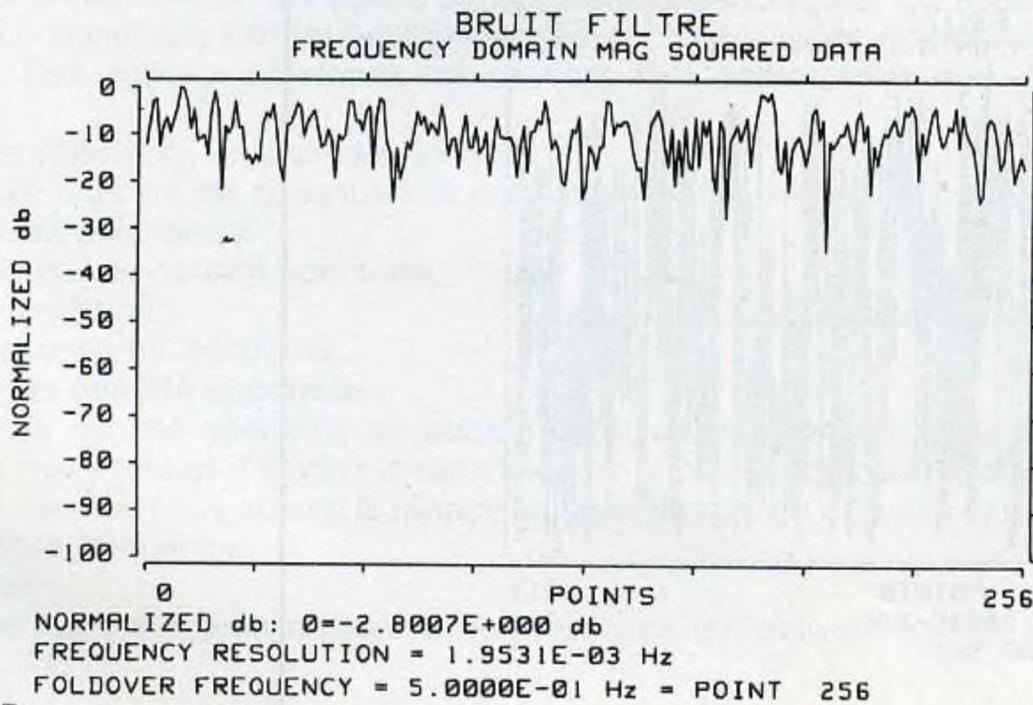
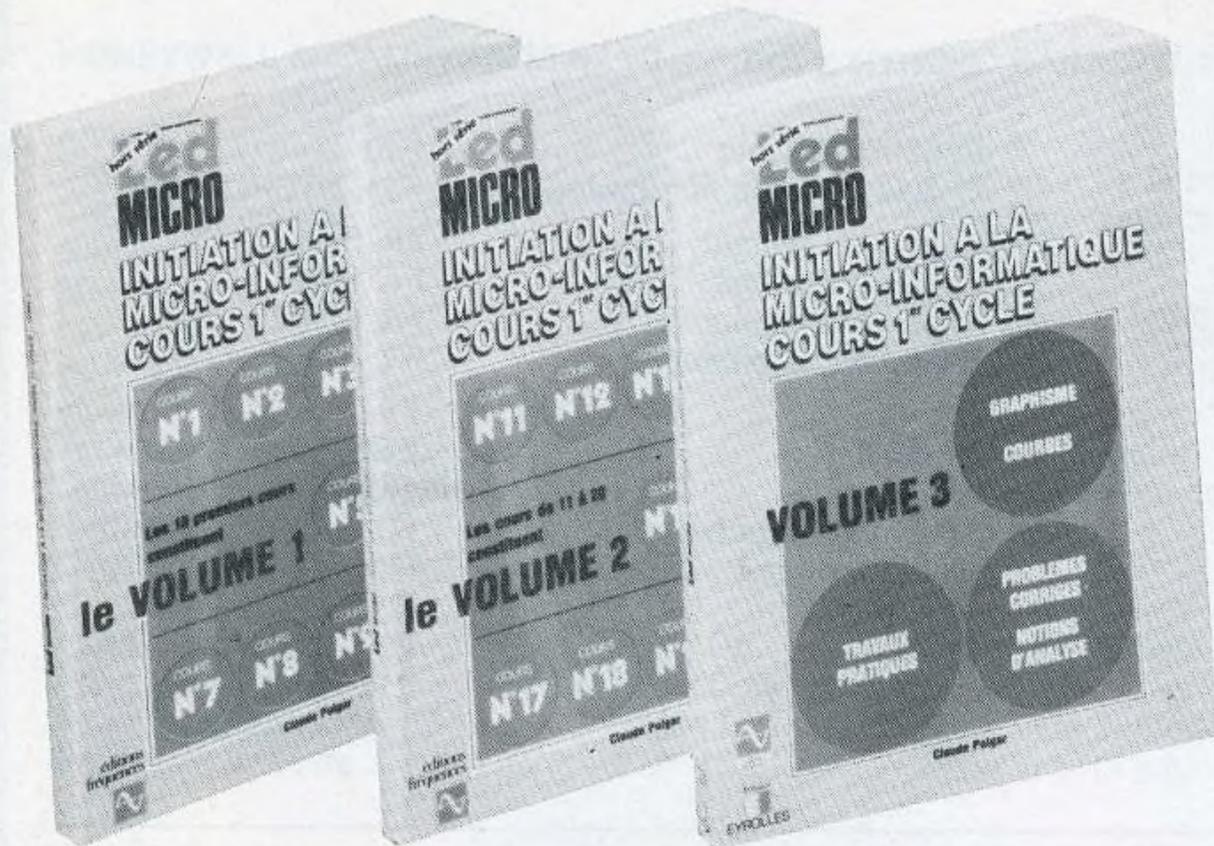


Fig. 4-D



**Le cours
d'initiation
le plus
complet
+ de 700 pages**

Non, on ne s'initie pas à la micro-informatique en 5 leçons !

Si vous croyez au Père Noël vous pouvez espérer apprendre l'Informatique en lisant les innombrables «Cours de BASIC pour débutants» qui ont poussé comme des champignons dans les années 1980. Votre ordinateur risque de finir ses jours au-dessus de votre armoire.

Mais si vous voulez vraiment apprendre à programmer il faut avoir le courage de commencer par A pour arriver à Z. Programmer est un loisir intelligent et peut devenir un métier passionnant, mais l'étude de la programmation nécessite un minimum de travail et de méthode.

Etre sérieux – c'est le pari que fit la revue LED-MICRO en publiant à partir de 1985 les 20 premiers cours de C. Polgar. Plus de 40 000 lecteurs les ont suivis. Ce succès nous a conduit à demander à C. Polgar de remettre son cours à jour et de le compléter. Le résultat : un ouvrage épais (3 tomes, plus de 700 pages format 21 x 27), permettant d'acquérir agréablement des connaissances solides.



Diffusion auprès des libraires assurée exclusivement par les Editions Eyrolles.

Initiation à la micro-informatique C. Polgar

Bon de commande à retourner aux Editions Fréquences
1, boulevard Ney 75018 Paris.

Je désire recevoir le tome 1 140 F (130 F + 10 F de frais de port)
le tome 2 140 F (130 F + 10 F de frais de port)
le tome 3 200 F (190 F + 10 F de frais de port)

Ci-joint mon règlement par :

CCP Chèque bancaire Mandat

Nom

Prénom

Adresse

Code postal Ville

Une seule
parmi près de 600 lettres
de lecteurs :

J'enseigne les mathématiques dans une Université de Sciences Humaines et j'ai été amenée, alors que n'avais moi-même reçu aucune formation à la micro-informatique, à initier des étudiants de 1^{re} année de Mathématiques et Sciences Sociales (MASS) à la programmation en S-BASIC (sur Goupil-3), dans le but de faire avec eux de l'analyse numérique élémentaire. Ce que j'ai fait, tant bien que mal, cette année, en collaboration avec deux autres collègues. Nous sommes conscientes d'avoir commis un certain nombre d'erreurs pédagogiques et nous souhaitons tenter d'y remédier l'an prochain.
J'ai découvert votre revue tout récemment, alors que j'arrivais quasiment au bout de mon enseignement. J'ai été très sensible à votre démarche pédagogique et je me sens personnellement tout à fait en accord avec votre manière de procéder. Je me suis procurée l'ensemble des n^{os} de la revue et me permettrai de puiser dans votre cours certains exemples ou certaines façons de présenter les choses l'an prochain. Donc merci à vous...
C.L. St Cloud, le 22/5/85

Initiation à la Micro-Informatique 1^{er} Cycle Tome 3 (enfin paru !)

3.16 (Suite et fin) L'affichage

- ★ Etude des instructions permettant d'effectuer des présentations « évoluées » : PRINT TAB - PRINT USING - LOCATE - COLOR en mode texte.
- ★ Présentation en tableaux de toutes sortes grâce à la pratique des opérateurs MODULO et DIVISION ENTIERE.
- ★ Beaucoup de programmes utilisent des assemblages de ces instructions et opérateurs... dont la combinaison n'est pas toujours facile.

3.17 Compléments

- ★ Etude des dernières instructions, fonctions et variables du cycle 1 : FILES, KILL, AUTO, ON ERROR GOTO, RESUME, ERR, ERL, DELETE, EDIT, RENUM TRON, TROFF, STOP, CONT, KEY ON, KEY OFF, FIX, BEEP.
- ★ Compléments de cycle 1 qui sont maintenant accessibles aux élèves : sur la précision et les erreurs dues à l'arrondi, sur la sélection, les boucles.

3.18 Graphisme

- ★ Une étude complète et détaillée sur les instructions graphiques en haute résolution : SCREEN, PSET, PRESET, STEP, LINE, CIRCLE, COLOR, POINT, PAINT, sans éluder aucune des difficultés et « pièges » classiques : l'incrustation de texte dans le dessin, les « bavures » dues au PAINT mal utilisé.
- ★ Une étude détaillée du langage graphique DRAW, avec ses subtilités et ses pièges (sous-chaînes X, paramètres variables dans le DRAW, etc.).
- ★ De nombreux exercices avec leurs solutions (80) et leurs illustrations sur des photos d'écran en couleur (48 photos).

3.19. Dessin des courbes

- ★ Un chapitre séparé du graphisme général (chapitre 3.18) de façon à ce que les « non matheux » puissent le sauter sans remords : ils ne seront pas punis !
- ★ Pour les matheux : une excellente révision et illustration des courbes de toutes sortes : $Y = f(x)$, courbes paramétrées, courbes en coordonnées polaires, avec des exemples utiles : courbes d'amortissement, astroïde, cardioïde, décomposition d'une fonction périodique par une série de Fourier.

3.20. Révision générale

- ★ L'enchaînement des notions selon l'ordre « pédagogique » qui a été utilisé jusqu'ici est bien différent de l'ordre « logique ». Autant qu'un cours d'anglais suit un ordre différent de celui (plus logique !) d'une grammaire anglaise.
- ★ Tout ce qui a été enseigné jusqu'ici résumé en 30 pages. Une référence pour retrouver la notion dont on a besoin à travers le cours et ses exercices. Mais aussi une réflexion sur la structure d'un langage informatique, d'où une préparation à la lecture des cours de PASCAL (par exemple !).

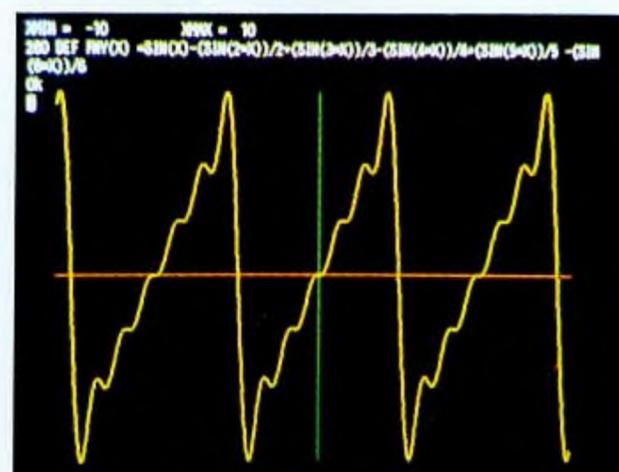
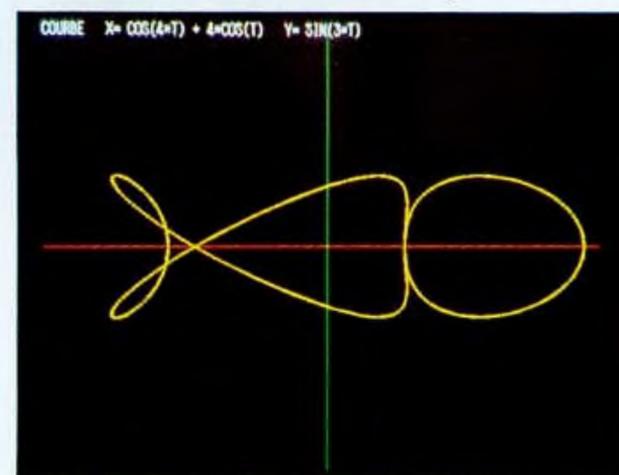
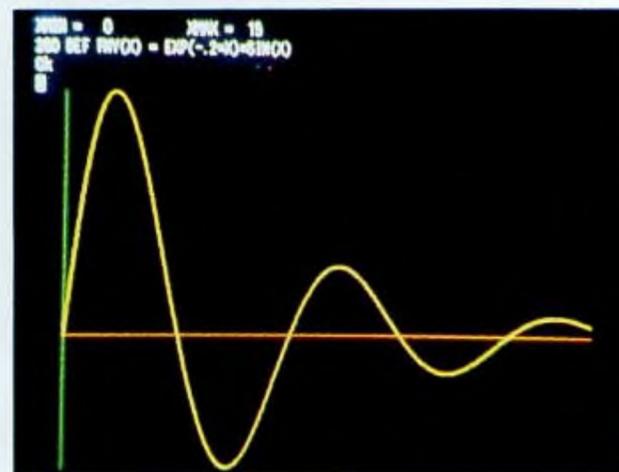
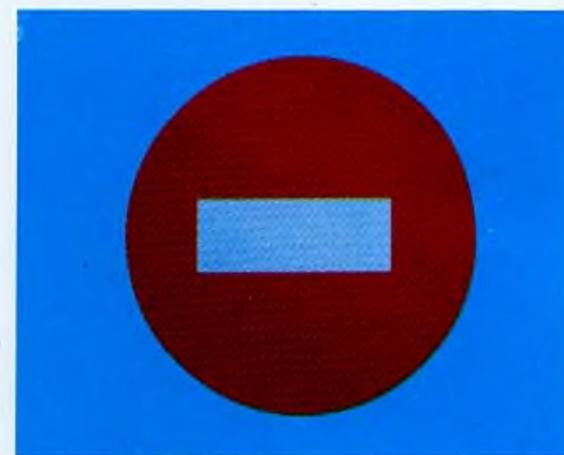
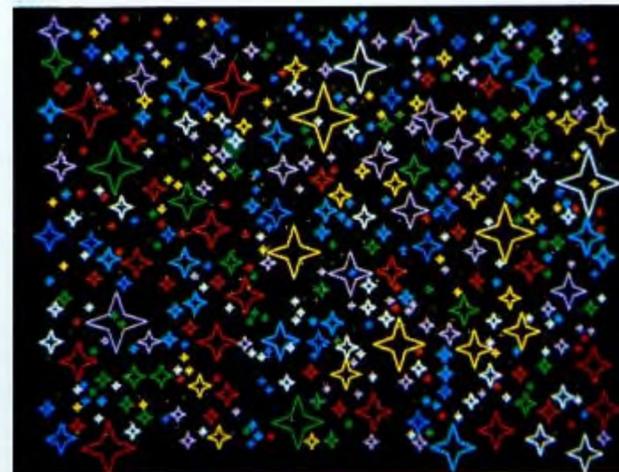
3.21. Techniques de mise au point

- ★ Les outils de base : étude des éditeurs de texte, connaissance et interprétation des messages d'erreur.
- ★ Comment rechercher et corriger ses erreurs.
- ★ La représentation du dialogue homme-machine, pour noter l'expérience que vous acquérez par la pratique.

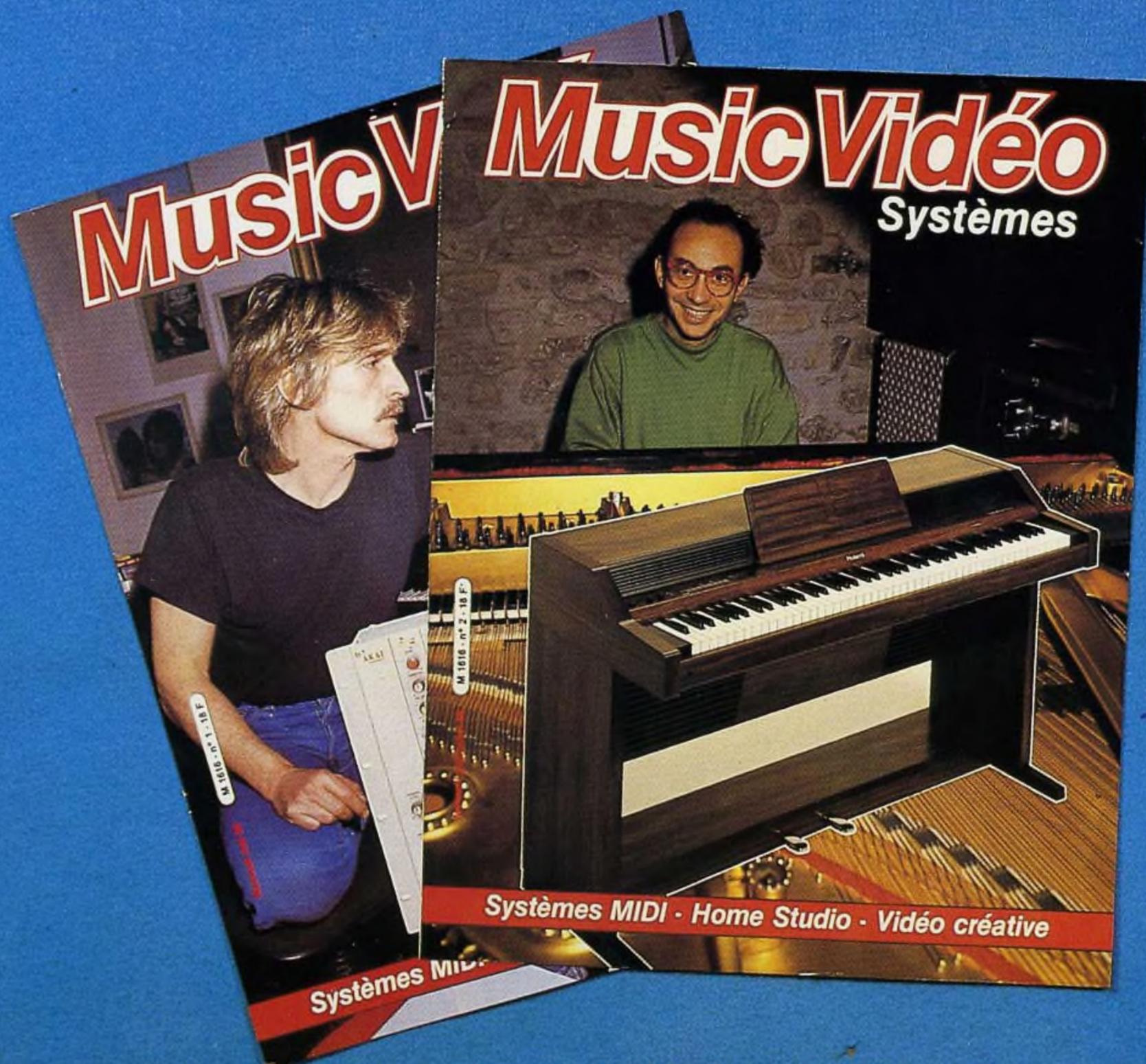
3.22. Problèmes de synthèse - Notions d'analyse

C'est à la fois la conclusion, la partie la plus originale et la plus utile de ce cours. L'auteur ne se contente pas de fournir une liste de problèmes avec leur solution : il se met à la place du programmeur débutant en essayant de décortiquer le « processus de réflexion » qui fait passer de l'énoncé d'un problème à sa solution : une initiation pratique à l'analyse.

1 livre broché de 248 pages pages 21 x 27, dont 8 pages en couleur



nouveau!



- *exploiter toutes les possibilités des systèmes MIDI*
- *réaliser vous-mêmes un clip vidéo*
- *tirer le maximum de vos synthétiseurs*
- *installer chez vous votre studio d'enregistrement*
- *tout savoir sur les nouveautés musique et vidéo créatives*

**Tout cela chaque mois
dans Music Vidéo Systèmes**

une publication des Editions Fréquences chez votre marchand de journaux

Editions Fréquences 1, boulevard Ney 75018 Paris - Tél. 46.07.01.97