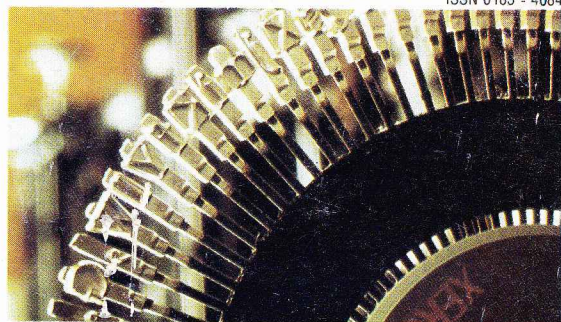
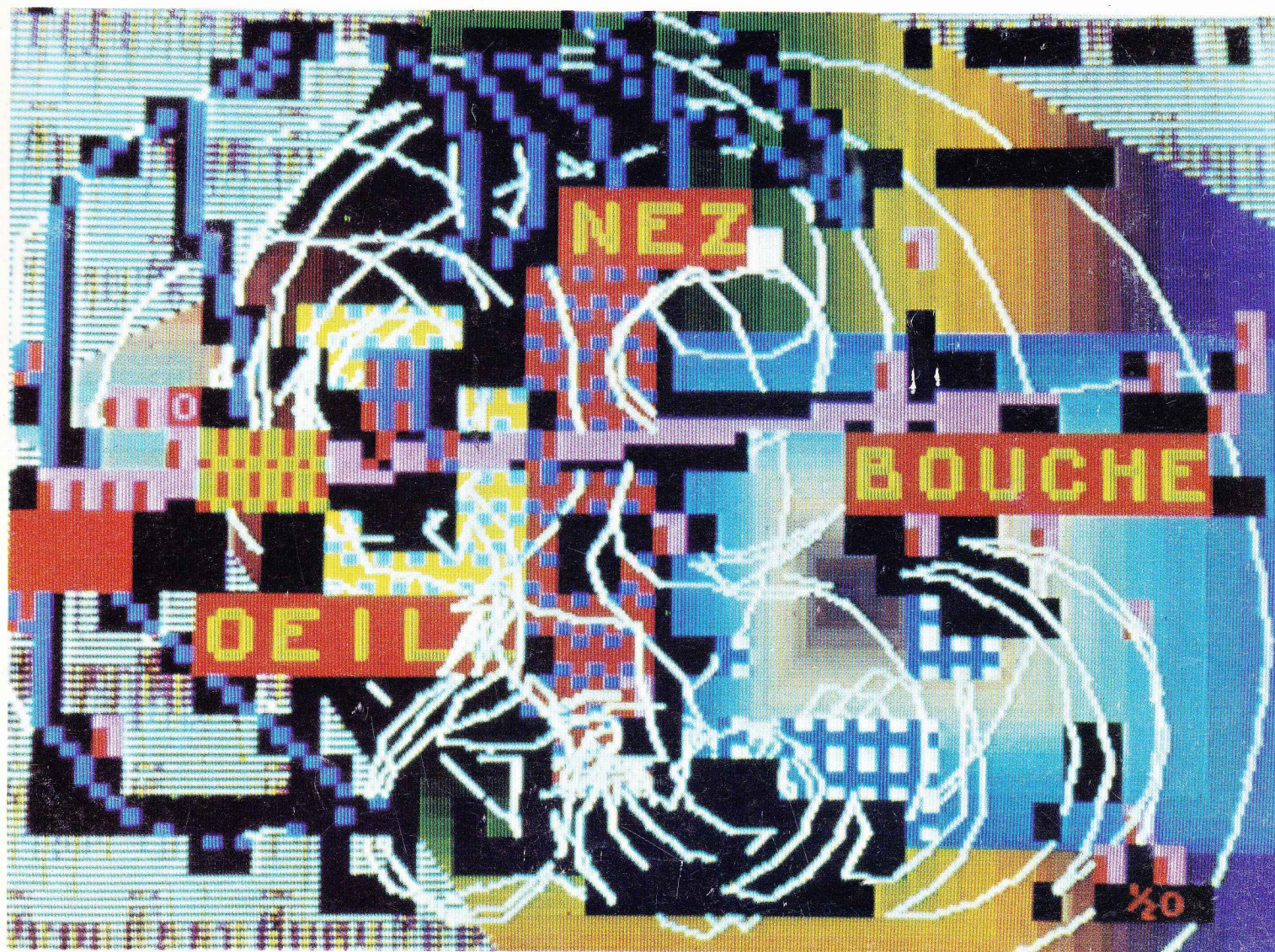


MICRO SYSTEMES



MICROPROCESSEURS/MICRO-ORDINATEURS/INFORMATIQUE APPLIQUÉE
N° 17 Bimestriel — Mai/Juin 1981 15^F



LE SON, LA COULEUR, L'INTELLIGENCE

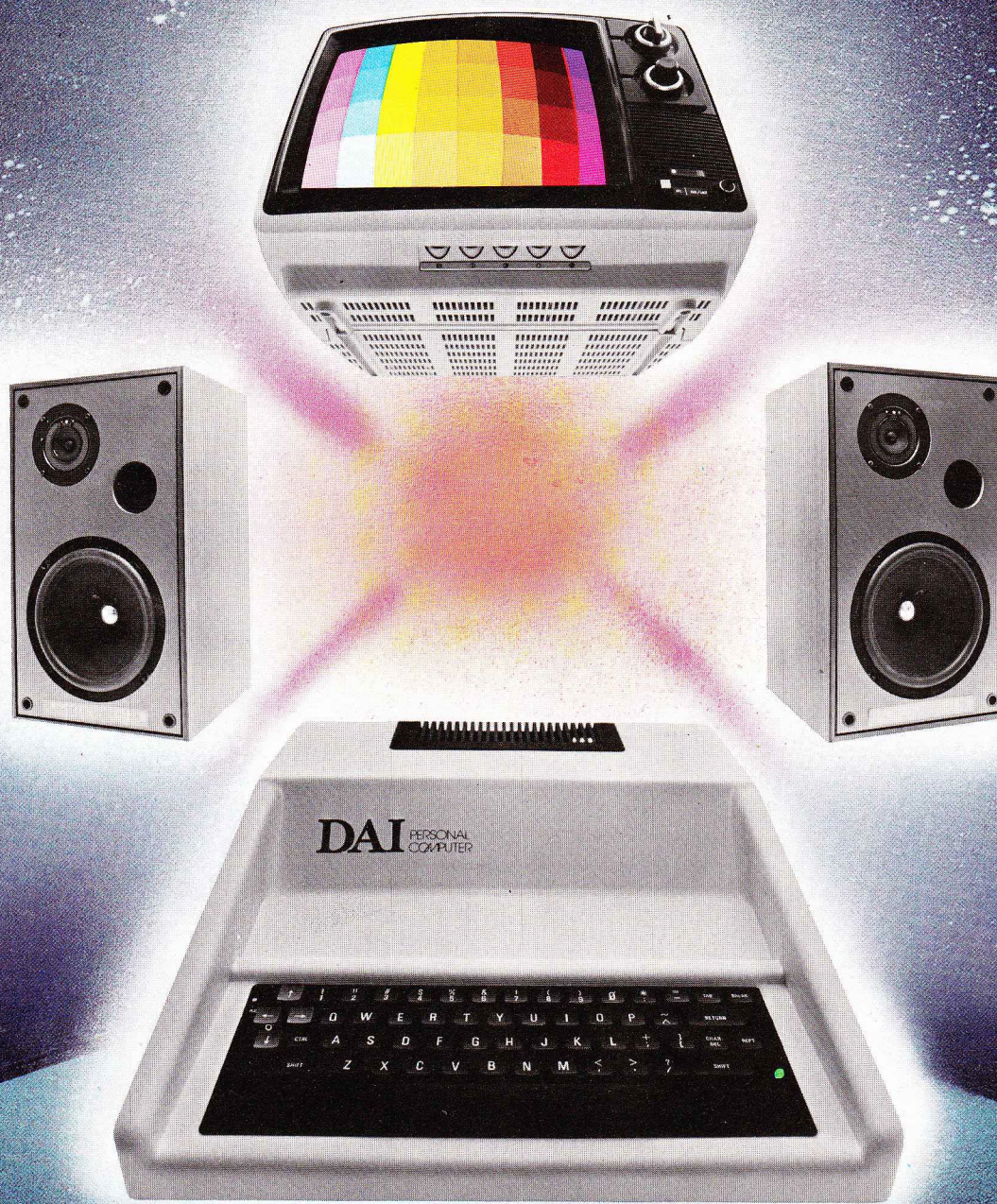


photo M.Y. Brandilly, réalisation AGRAPH

DAI Version standard intégrant :

- BASIC très puissant - semi-compilé - ultra-rapide sur 24 K ROM.
- 13 Modes graphiques dont la Haute Résolution 256X336 en 16 couleurs. (Fct. DRAW - DOT - FILL)
- Affichage de 24 Lignes - 60 Caractères (MAJ/Min.).
- Moniteur Langage Machine 8080.
- Editeur puissant.
- Synthèse Musicale : 4 Générateurs programmables, sorties en stéréophonie. (Fct. ENVELOPE - SOUND - FREQ - TREMOLO - GLISSANDO - NOISE)
- Synthèse Vocale. (Fct. TALK).
- 48 K RAM Utilisateur.
- Interface série RS 232 - 2 interfaces cassettes.
- Interface parallèle (3 ports programmables).
- Interface TV couleur.

● Nombreuses Options : Floppy, Process. Arith., Imprimante, Paddles, etc.



multi/oft Importateur exclusif pour la France, 25, rue BARGUE - 75015 PARIS - Tél. : 783 88 37

DAI SA : 60, rue de la Fusée - 1130 BRUXELLES - BELGIQUE - Tél. : 02/216 60 10

USA - ALLEMAGNE - HOLLANDE - AUTRICHE - Gde BRETAGNE

Nous invitons les revendeurs à nous contacter.

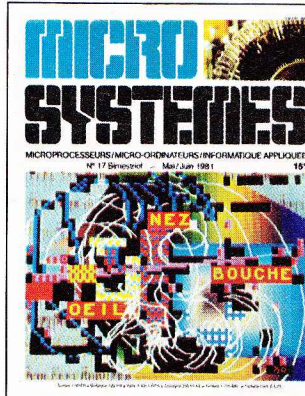
Sommaire n° 17

	Pages
Calendrier :	
Conférences, expositions, manifestations internationales 1981	45
Initiation :	
Le microprocesseur et son environnement : « Le clavier et son interface »	52
Communications :	
La transmission de données	61
Le Basic :	
Basic et mathématiques : Résolution numérique des équations différentielles	71
Gestion :	
Analyse budgétaire : programme Basic	80
Composants :	
L'IAPX 432 : Un microprocesseur 32 bits	76
Les microprocesseurs 16 bits	111
Périphériques :	
Les imprimantes	85
Couverture :	
Le principe d'émergence	106
Formule micro :	
La voiture robot du Microtel Club de Bordeaux	121
Informatique :	
Le langage Pascal	129
● Micro-Systèmes Magazine	46
● Compact disc digital audio	135
● Dames Challenger	137
● 2164 : Une mémoire RAM de 64 K × 1 bit	141
● Livre : Un microprocesseur « pas à pas »	145
● Presse internationale... les tendances	147
Divers	
Courrier des lecteurs	150
Informations	155
Petites annonces	203
Bonus « Micro-Systèmes »	209
Pour commander vos numéros manquant	208
Index des annonceurs	210
Coupons : Service lecteurs, petites annonces, abonnement ..	211

Ce numéro a été tiré à 86 000 ex.

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

MICRO SYSTEMES



Notre couverture :
Gros plan sur une tête d'impression « marguerite »... les imprimantes (p. 85).

« Il y a émergence quand de nouvelles conséquences apparaissent, quand de nouvelles formes émergent sur l'écran. » Ce principe est illustré ici par une image à multiples niveaux de transmission réalisée par B. Demiaux sur système Module... le principe d'émergence (p. 106).

Président-Directeur général
Directeur de la publication :

Jean-Pierre Ventillard

Rédacteur en chef :
Alain Tailliar

Chefs de rubriques :
Dave Habert
J. Ferber
J.-M. Durand

Secrétariat :
Catherine Salbreux
Danielle Desmaretz

Ce numéro a été réalisé avec la participation de :
E. Adamis, S. André, D. de Beaulieu, G. Biscos, W. Debache, B. Demiaux, P. Goujon, M. Guérin, L. Guinard, P. Jaulent, B. Lang, A. Leprêtre, J.-C. Nicoletto, C. Prunet, Y. Torre.

Rédaction (nouvelle adresse) :
43, rue de Dunkerque, 75010 Paris
Tél. : 285.04.46

Maquette : Josiane Garnier

Publicité :
Advertising International Manager
M. Sabbagh
Chef de Publicité : Francine Fohrer
S.P.E. – Tél. : 200.33.05

Abonnements : 2 à 12, rue de Bellevue, 75940 Paris Cedex 19. – Tél. : 200.33.05. – 1 an (6 numéros) : 75 F (France), 105 F (Etranger).

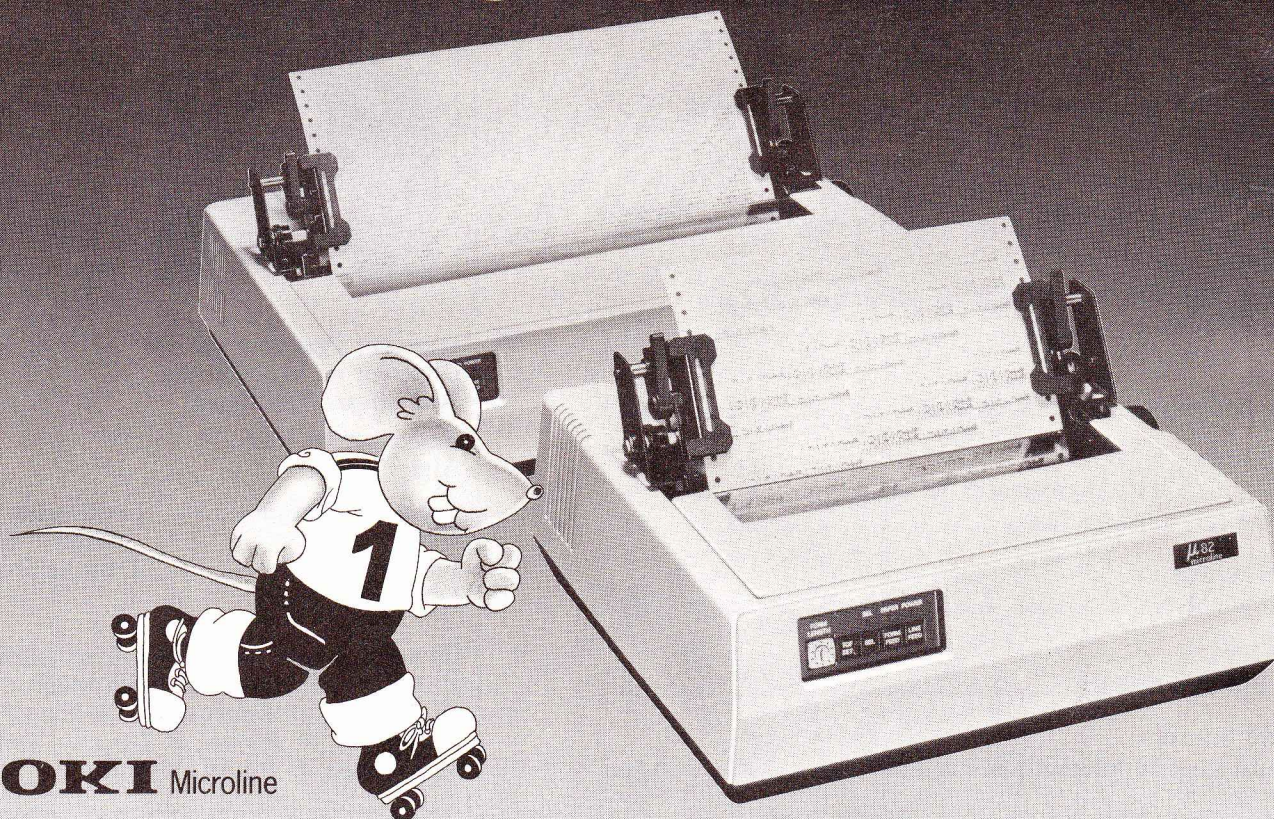
Société Parisienne d'Édition
Société anonyme au capital de 1 950 000 F
Siège social : 43, rue de Dunkerque, 75010 Paris
Direction – Administration – Ventes :
2 à 12, rue de Bellevue, 75940 Paris Cedex 19
Tél. : 200.33.05 – Télex : PGV 230472 F



Copyright 1981. – Société Parisienne d'Édition
Dépôt légal : 2^e trimestre 1981. – N° d'éditeur 900
Distribué par SAEM Transports Presse.

Micro-Systèmes décline toute responsabilité quant aux opinions formulées dans les articles. Celles-ci n'engagent que leurs auteurs.

Les fortes têtes accélèrent.



OKI Microline

Avec les Microline 82 et 83 à frappe bi-directionnelle, déplacement optimisé et saut de page réglable, la famille des fortes têtes (durée de vie 200 millions de caractères) s'agrandit et accélère !

Caractéristiques communes : tête balistique à aiguille 6 ou 8 lignes/pouce, entraînement friction et picots, 96 caractères ASCII + 8 alphabets + semi-graphique, interfaces parallèles et RS 232.

Microline 80 : unidirectionnelle, 80 cps, 80 col. papier jusqu'à 241 mm, matrice 9 x 7.

Microline 82 : bi-directionnelle, déplacement optimisé 120 cps, 80 col, papier jusqu'à 241 mm, matrice 9 x 9.

Microline 83 : bi-directionnelle, déplacement optimisé, 120 cps, 136 col, papier jusqu'à 406 mm, matrice 9 x 9.

Distributeur exclusif,

METRO
LOGIE
L'avance technologique, le support, le service

INT. FACILITER M. SYST.

Ce matériel est disponible chez les meilleurs spécialistes.

13007 MARSEILLE

PROVENCE SYSTEM

Le Saint James. 74, rue Sainte - Tél. : (91) 33.22.33

13770 VENELLES par Aix-en-Provence

PRADELLE INFORMATIQUE

Domaine de Fontcuberte - Tél. : (42) 57.70.01

34000 MONTPELLIER

IFI

12, rue de Castilhon - Tél. : (67) 58.58.28 - Télex 48.07.30

75008 PARIS

SIVEA S.A.

20, rue de Leningrad - Tél. : 522.70.66

75010 PARIS

ILLEL CENTER INFORMATIQUE

220, rue Lafayette - Tél. : 208.61.87

75011 PARIS

MICRO INFORMATIQUE DIFFUSION

60, avenue de la République - Tél. : 357.83.20

75015 PARIS

ILLEL CENTER INFORMATIQUE

143, avenue Félix Faure - Tél. : 554.97.48

75016 PARIS

PENTASONIC

5, rue M. Bourdel - Tél. : 524.23.16

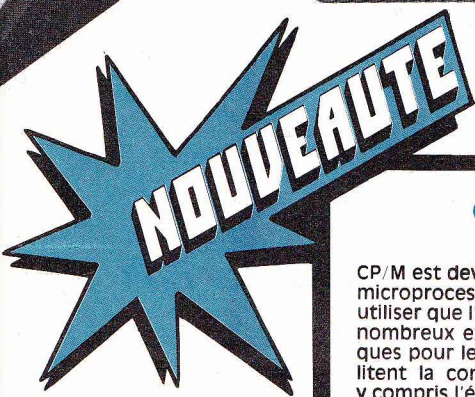
MONACO

MICROTEK

2, boulevard Rainier III - Tél. : (93) 50.43.44

Pour plus de précision cercelez la référence 105 du « Service Lecteurs »

une gamme complète de best-sellers microprocesseurs microordinateurs



GUIDE DU CP/M AVEC MP/M 420 p, Réf. C8, 98 F TTC

CP/M est devenu le système d'exploitation standard pour microprocesseurs. Sybex le rend aujourd'hui aussi facile à utiliser que l'alphabet, grâce à ce guide très progressif. De nombreux exemples de programmes, suggestions pratiques pour les opérations et des tableaux descriptifs facilitent la compréhension du fonctionnement du CP/M, y compris l'éditeur et l'assembleur. Ce livre couvre toutes les versions du CP/M jusqu'au CP/M 2.2 ainsi que le CDOS et le MP/M multiutilisateur.



M.S. 5.81.

BON DE COMMANDE RAPIDE

Nom: _____ Société: _____
 Adresse: _____
 Code Postal/Ville: _____ Pays: _____
 Tél: _____ Télex: _____

Veuillez m'envoyer les livres suivants:

____ ex. PB02 ____ ex. PB01 ____ ex. PA01 ____ ex. C8 ____ ex. C3
 ____ ex. C1B ____ ex. C4 ____ ex. C5 ____ ex. C780 ____ ex. D802

Ci-joint mon règlement de: _____ F, y compris frais d'envoi.
 (Conditions départ. 1 livre: 9,50 F, 2-4: 16 F, 5-8: 20 F)

Veuillez m'envoyer votre catalogue détaillé.

à retourner à **SYBEX** 18, rue Planchat 75020 Paris Tél. (1) 370.32.75 Télex 211801 F

ATTENTION

A partir du 15 Mai 1981, nouvelle adresse:

SYBEX
 Centre Paris Daumesnil
 4, place Félix-Éboué - 75012 Paris
 Tél. 341.71.10

Le microprocesseur et son environnement

Le clavier et son interface...

Chacun se souvient, lors du concours « Formule μ » du véhicule équipé d'un clavier qui permettait à notre concurrent d'ajuster les derniers paramètres avant le départ sur la piste.

Le clavier, ensemble d'interrupteurs disposés en « matrice », est sans doute le système le plus simple permettant à l'homme de communiquer programmes et données à la machine.

Plusieurs types coexistent actuellement sur le marché, et peuvent être classés en deux grandes catégories :

- les claviers « complètement encodés » qui délivrent pour chaque touche le code ASCII correspondant,

- les claviers « non encodés » qui restent de simples matrices de touches. C'est donc le logiciel du système qui devra délivrer le code binaire de chaque touche enfoncée.

Une fois de plus nous retrouvons, dans cet exemple d'application, la dualité « Hardware/Software » : ce qui n'est pas pris en charge par l'un doit l'être par l'autre.

Nous vous proposons de parcourir ensemble les différentes étapes permettant de « gérer » un clavier « non encodé ».

Le décodage d'un bouton-poussoir...

Puisqu'un clavier se compose d'un ensemble de touches nous allons, dans un premier temps, examiner la façon dont peut-être détectée la fermeture d'un contact relié à un système à microprocesseur.

La figure 1 représente une liaison « bouton-poussoir/unité centrale » réalisée grâce à un circuit d'interface appelé PIA (Péripheral Interface Adaptor, boîtier 6821).

Le fonctionnement est particulièrement simple : toute action pendant un temps « suffisant » sur la touche T, impose un niveau logique zéro sur la ligne PA₁, programmée en entrée (au repos PA₁ = 1).

Il suffit, pour déterminer si cette touche est enfoncée, d'effectuer un ET logique (masque) entre le contenu du PORT A et la valeur hexadécimale 02 (ce qui correspond au deuxième fil du PORT (PA₁) car \$ 02 = 00000010). S'il y a eu action sur la touche T le contenu du port A est 1111 1101 et le résultat de cette opération de « masquage » doit alors positionner à 1 l'indicateur Z du registre d'état.

C'est ce que réalise le programme de la figure 2 en mettant la variable FLAG à 0 si la touche a été enfoncée et à 1, dans le cas contraire. L'état de cette variable

nous renseignera donc sur l'appui du bouton poussoir.

Ce programme est rédigé en langage d'assemblage 6800, c'est-

certain « temps de rebondissement » (à l'ouverture comme à la fermeture) que le niveau logique correct est transmis par le

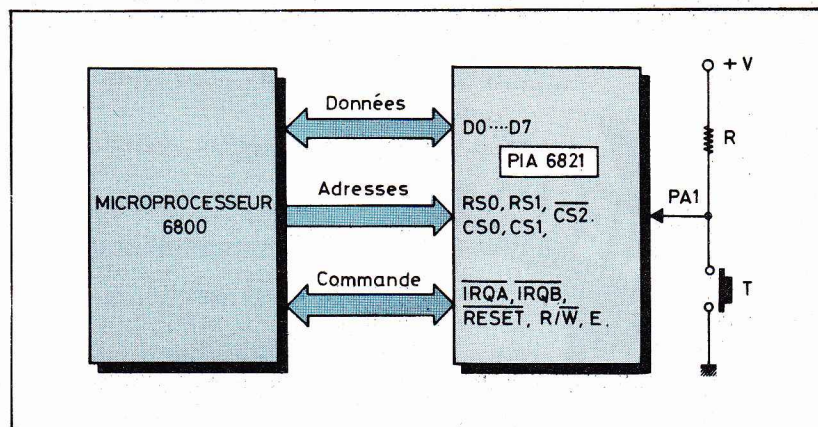


Fig. 1. - L'appui sur le bouton poussoir T impose un niveau bas sur la ligne PA₁ programmée en entrée.

à-dire qu'en plus des mnémoniques il utilise des directives d'assemblage (NAM, ORG, EQU...) qui sont des aides appréciables à la programmation. Nous incitons nos lecteurs peu familiarisés avec la programmation en langage d'assemblage (telle qu'on la pratique sur un système de développement, par exemple) à lire avec attention l'encadré 1.

L'exemple que nous venons d'envisager ne tient pas compte du problème le plus fréquemment posé par les commutateurs mécaniques : celui des rebonds des contacts.

La figure 3 illustre ce phénomène. C'est seulement après un

commutateur.

Une solution simple pour se prémunir des rebonds, consiste à attendre que l'état d'une touche soit stable, pendant un délai compris entre 0,2 ms et 10 ms suivant les qualités du clavier, avant toute interprétation.

Notons de plus que, dans notre précédent exemple, si l'enfoncement du « poussoir » intervient après l'instruction de lecture du port A (LDA A PIAD0A), l'unité centrale n'en est pas avertie.

L'organigramme de la figure 4 et sa traduction programmée de la figure 5 tiennent compte de ces dernières observations, à savoir :


```

00010 00001          NAM  CLAV1
00020 00002          OPT  NOP,S
00030 00003A 0010   ORG   $10      Adresse de Base
00040 00004          *
00050 00005          * LISTES D'EQUIVALENCES *
00060 00006          *
00070 00007          0002 A MASQUE EQU  $02
00080 00008          0050 A FLAG  EQU  $50
00090 00009          8004 A BASE  EQU  $8004
00100 00010          *
00110 00011          8004 A PIAD0A EQU  BASE
00120 00012          8005 A PIACRA EQU  BASE+1
00130 00013          *
00140 00014A 0010 7F 8005 A   CLR  PIACRA b2=0,selection DDRA
00150 00015A 0013 7F 8004 A   CLR  PIAD0A Port A en Entree
00160 00016A 0016 86 04  A   LDAA  #$04
00170 00017A 0018 B7 8005 A   STAA  PIACRA b2=1 selection ORA
00180 00018A 001B 7F 0050 A   CLR  FLAG
00190 00019A 001E B6 8004 A   LDAA  PIAD0A Lecture Port A
00200 00020A 0021 84 02  A   ANDA  #MASQUE
00210 00021A 0023 27 03 0028 BEQ  FIN
00220 00022A 0025 7C 0050 A   INC  FLAG
00230 00023A 0028 20 FE 0028 FIN  BRA  FIN
00240 00024          END
TOTAL ERRORS 00000--00000
BASE 8004 FIN 0028 FLAG 0050 MASQUE 0002 PIACRA 8005
PIAD0A 8004
    
```

Fig. 2. - Ce programme permet de déterminer si la touche T, reliée à PA1, est enfoncée. C'est la valeur de la variable FLAG qui nous renseigne sur son état.
 Ligne 140 : Mise à zéro du registre de contrôle donc de son bit 2 : sélection du DDRA.
 Ligne 150 : Mise à zéro des bits de DDRA. Les lignes du port A sont des entrées.
 Ligne 160 : Chargement de l'accumulateur avec la valeur 0000 0100 = 04.
 Ligne 170 : 04 est stockée dans le registre de contrôle : adressage ORA.
 Ligne 180 : L'indicateur FLAG est mis à 0.
 Ligne 190 : Lecture du port A (b2 à 0 signifie « touche enfoncée »).
 Ligne 200 : ET logique avec la valeur 0000 0010.
 Ligne 210 : Branchement à FIN si le résultat du ET est 0000 0000 (touche enfoncée).
 Ligne 220 : Si la touche n'a pas été enfoncée, FLAG est mise à 1.

```

00010 00001          NAM  CLAV4
00020 00002          OPT  NOP,S
00030 00003A 0010   ORG   $10      Adresse de base
00040 00004          *
00050 00005          * LISTES D'EQUIVALENCES *
00060 00006          *
00065 00007          00FF A PILE  EQU  $00FF  Adresse de la PILE
00070 00008          0050 A FLAG  EQU  $50
00080 00009          0002 A MASK  EQU  $02
00090 00010          8004 A BASE  EQU  $8004
00100 00011          8004 A PIAD0A EQU  BASE  Adresse DDRA/DRA
00110 00012          8005 A PIACRA EQU  BASE+1 Adresse CRA
00125 00014A 0010 8E 00FF A   LDS  #PILE
00130 00015A 0013 7F 8005 A   CLR  PIACRA b2=0 selection DDRA
00140 00016A 0016 7F 8004 A   CLR  PIAD0A Port A en Entree
00150 00017A 0019 86 04  A   LDAA  #%00000100
00160 00018A 001B B7 8005 A   STAA  PIACRA b2=1 selection ORA
00170 00019A 001E 7F 0050 A   CLR  FLAG
00180 00020A 0021 B6 8004 A   ATTEND LDAA  Lecture Port A
00190 00021A 0024 84 02  A   ANDA  #MASK Masque Logique
00200 00022A 0026 26 F9 0021 BNE  ATTEND
00210 00023A 0028 BD 0037 A   JSR  DELAI Appel Sous/Programme
00220 00024A 002B B6 8004 A   LDAA  PIAD0A Lecture Port A
00230 00025A 002E 84 02  A   ANDA  #MASK
00240 00026A 0030 27 EF 0021 BEQ  ATTEND
00250 00027A 0032 7C 0050 A   INC  FLAG
00260 00028A 0035 20 EA 0021 BRA  ATTEND
00270 00029          *
00280 00030A 0037 CE 5000 A DELAI LDX  #5000
00290 00031A 003A 09          LOOP  LDX  LOOP
00300 00032A 003B 26 FD 003A BNE  LOOP
00310 00033A 003D 39          RTS
00320 00034          END
TOTAL ERRORS 00000--00000
    
```

Fig. 5. - Programme correspondant à l'organigramme ci-contre. Le signe % indique que l'opérande est exprimé en binaire.

appel au sous-programme DELAI, puis exécution d'une boucle d'attente tant qu'aucune autre action sur T n'a pas été détectée.

Une ligne parmi huit...

Portons notre attention sur la figure 6 où un commutateur « fictif » relié à la masse, c'est-à-dire

au niveau logique zéro, évolue de la position 0 à la position 7 respectivement liées au PORT A du PIA programmé en entrée.

Il s'agit d'écrire un programme permettant de déterminer sur quelle ligne du PORT A un commutateur a été actionné et d'en afficher le numéro. Notons qu'il n'est pas question, dans notre

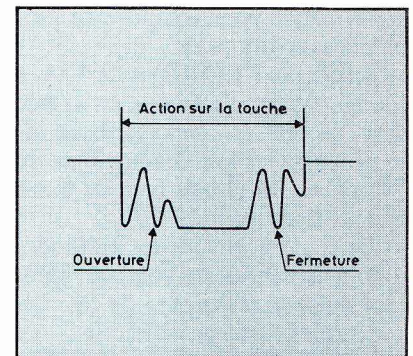
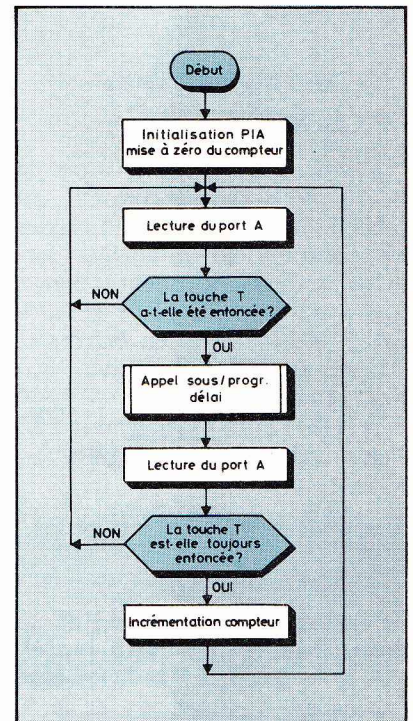


Fig. 3. - Tout commutateur mécanique présente à l'ouverture ou à la fermeture pré-sente à l'ouverture ou à la fermeture un phénomène de rebondissement. C'est seulement après un certain « temps de rebondissement » que le niveau logique correct peut être transmis au système.

Fig. 4. - Organigramme décrivant une solution « tenant compte » des rebonds. Le sous-programme DELAI est une temporisation.



cas, d'écrire la séquence permettant cet affichage. Les systèmes de développement disposant de l'Assembleur 6800 (Exorciser, MAK 68....) possèdent un sous-programme baptisé « OUT2H » dans leur bibliothèque réalisant cette fonction.

L'organigramme de la figure 7 et le programme de la figure 8 indiquent la démarche que nous

Le problème le plus fréquent posé par les commutateurs mécaniques est celui des rebondissements des contacts.

avons suivie pour résoudre ce problème sachant que :

– Après un « RESET », le PIA a toutes ses lignes fixées en entrée avec, et ceci est important, un niveau logique 1 sur chaque ligne du PORT A (un niveau logique 0 sur chaque ligne du PORT B).

De toute évidence, les méthodes que nous venons de voir sont limitées puisque le nombre de touches utilisables est proportionnel au nombre de lignes du circuit d'interface parallèle.

Il est plus judicieux d'utiliser une matrice de décodage à deux dimensions, avec n lignes et m colonnes, comme le montre la figure 9. Ainsi la frappe sur une touche T se traduit par une liaison entre la ligne et la colonne à l'intersection matérialisée par la touche enfoncée.

Du bouton-poussoir au clavier

Etudions ensemble le listing de la figure 10 correspondant au schéma de la figure 11. Le clavier est constitué de 16 touches formant une matrice dont 4 lignes sont reliées au PORT A (PA0 à PA3) programmées en entrées et 4 lignes au PORT B (PB0 à PB3) fixées en sorties.

Après l'initialisation du PIA l'unité centrale reçoit l'ordre par les instructions

```
LDAA # % 1111 1110
et
STAA PIADOB
```

de « déposer » sur la ligne PB0 un niveau logique nul. Nous rappelons que le signe # symbolise l'adressage « immédiat » et le % indique que l'opérande est exprimé en binaire (et non en hexadécimal caractérisé par \$).

Les instructions LDAA # \$04 et STAA \$50 positionnent un compteur avec la quantité 04 correspondant au nombre de colonnes et rangent cette quantité à l'adresse 50.

Le groupe d'instructions suivant :

```
LDAA PIADOA
```

```
AND A # % 0000 1111
CMP A # % 0000 1111
```

vérifie si une touche est enfoncée par la lecture du PORT A et masquage logique.

Lorsqu'une touche est actionnée le compteur de programme est chargé avec l'adresse correspondant à l'étiquette COMPT.

Analysons deux cas de figure :

● Aucune touche n'est enfoncée :

Après lecture du PORT A nous obtenons la valeur binaire 1111 1111. L'opération de « masque » avec la quantité 0000 1111 place dans l'accumulateur A la valeur 0000 1111.

L'instruction CMPA # % 0000 1111 (correspondant à une « soustraction virtuelle » entre le contenu de A et la quantité précisée par l'instruction CMPA), positionne l'indicateur Z à 1.

Le programme continue donc en séquence et exécute l'instruction de décalage logique d'une position vers la gauche (la nouvelle valeur placée sur le PORT B est égale à 1111 1100).

(La seconde colonne est maintenant au 0 volts).

Nous arrivons au groupe d'instructions (lignes 003C et 003F) :

```
DEC $50
BNE RETOUR
```

qui expriment l'idée suivante :

– « Le contenu du compteur est-il égal à zéro ? »

– « Non, il est égal à trois ».

Dans ce cas, le programme se « branche » à l'étiquette RETOUR pour exécuter une nouvelle fois ces différentes instructions.

● Une touche est enfoncée :

Supposons qu'après lecture du PORT A nous obtenons la valeur binaire 1111 1110. L'instruction AND A # % 0000 1111 affecte à l'accumulateur A la quantité 0000 1110 (résultat du ET logique entre 1111 1110 et 0000 1111).

On comprend mieux dès lors, l'instruction CMPA # % 0000 1111 qui avec la précédente valeur

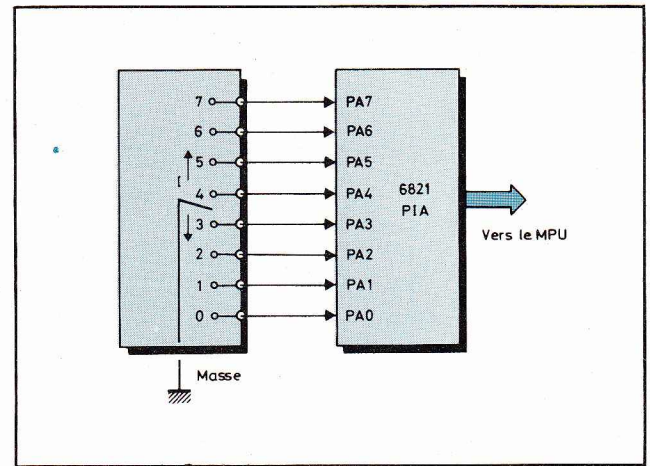


Fig. 6. – Le commutateur fictif I impose successivement un niveau « 0 » sur chacune des lignes du port A.

Fig. 7. – Organigramme illustrant la démarche que nous avons suivie pour déterminer sur quelle ligne du port A une touche est enfoncée.

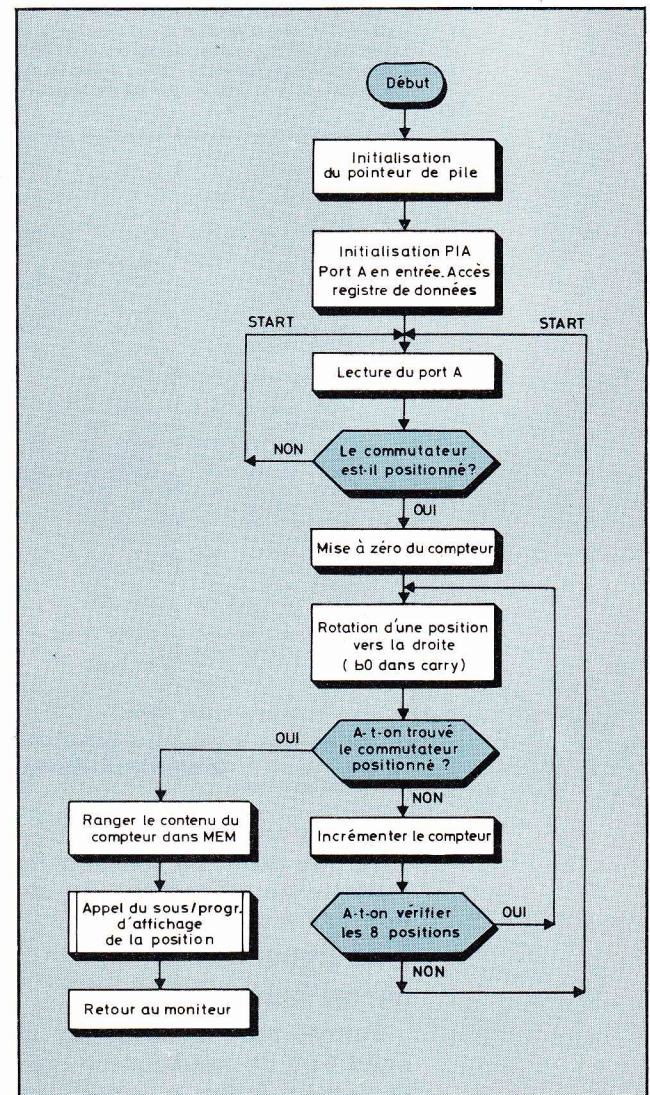


Fig. 8. - Programme correspondant à l'organigramme de la figure 7.

```

PAGE 001 TEST1C .SA:0 TEST1 LECTURE DU PORT A DU PIA 6821
00010 00001          NAM TEST1
00020 00002          TTL LECTURE DU PORT A DU PIA 6821
00030 00003          OPT LLEN=80,S
00040 00004 0000          ORG $0000

00060 00006          *****
00070 00007          * PROGRAMME DE LECTURE DU PIA(PORT A) *
00080 00008          *****

00100 00010A 8000          ORG $8000
00110 00011A 8000 0032 A RESERV RMB 50
00120 00012          8032 A STACK EQU *

00140 00014          0010 A BASE EQU $10
00150 00015          0010 A PIADA EQU BASE (DDRA/DRA)
00160 00016          0011 A PIACA EQU BASE+1 (CRA)
00170 00017          0012 A PIADB EQU BASE+2 (DDRB/ORB)
00180 00018          0013 A PIACB EQU BASE+3 (CRB)
00190 00019          *
00200 00020          F01B A OUT2H EQU $F01B
00210 00021          F000 A MAKEBUG EQU $F000
00220 00022          0000 A MEM EQU $0000

00240 00024          *****
00250 00025          * PROGRAMME PRINCIPAL *
00260 00026          *****

00280 00028A 8032 SE 8032 A          LDS #STACK
00285 00029          *
00290 00030          * INITIALISATION DU PIA EN ENTREE*****
00295 00031          *

00320 00032A 8035 7F 0011 A          CLR PIACA Bit 2 a 0 donc acces a DDRA
00330 00033A 8038 7F 0010 A          CLR PIADA Port A en entree
00340 00034A 803B 86 04 A          LDAA #04 bit b2=1 acces ORA
00345 00035A 803D 97 11 A          STAA PIACA
00350 00036A 803F 96 10 A          START LDAA PIADA
00360 00037A 8041 81 FF A          CMFA #FF
00370 00038A 8043 27 FA 803F BEQ START
00380 00039A 8045 5F CLR: Mise a zero du compteur
00390 00040A 8046 46 RETOUR RORA bit b0 dans CARRY

00400 00041A 8047 24 07 8050 BCC LUT
00410 00042A 8049 5C INCB Incrementation du compteur
00420 00043A 804A C1 08 A          CMFB #08 Tous les bits (8) sont verif
00430 00044A 804C 27 F1 803F BEQ START
00440 00045A 804E 20 F6 8046 BRA RETOUR recherche position Interrupt
00450 00046A 8050 D7 00 A          OUT STAB MEM Position INTERRUPT. dans MEM
00460 00047A 8052 CE 0000 A          LDX #MEM
00470 00048A 8055 BD F01B A          JSR OUT2H Affichage de la position
00480 00049A 8058 7E F000 A          JMP MAKEBUG Retour sous moniteur
00490 00050          END
TOTAL ERRORS 00000--00000
    
```

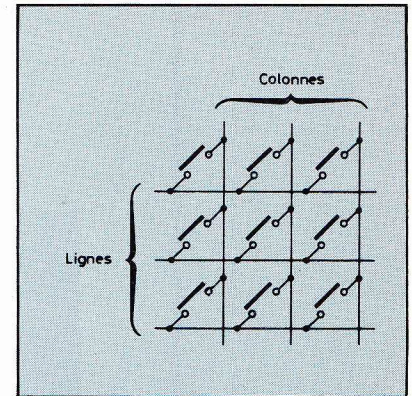


Fig. 9. - Il est judicieux de monter les touches en « matrice ».

contenue dans A impose l'indicateur d'état Z à 0 (en effet 0000 1110 - 0000 1111 donne un résultat **négatif mais non nul**) entraînant ainsi un branchement à l'étiquette COMPT.

Le décalage d'une position vers la droite (LSRA) du contenu de l'accumulateur A (bit 0 dans C) suivi du test du bit C du registre d'état, permet de connaître la position de la touche enfoncée.

Nous allons aborder, à titre de conclusion, une méthode actuellement très utilisée, dite « Polling ».

La méthode de « polling »

La méthode de « polling »* est très utilisée dans la pratique ; il est donc recommandé de bien la maîtriser. Son principe, très simple, est illustré **figure 12**.

Lorsqu'aucune touche n'est enfoncée, les quatre entrées de la porte NAND sont au niveau haut, ce qui impose un niveau bas à sa sortie, c'est-à-dire sur la ligne CA1 du PIA.

Supposons qu'une touche du clavier soit activée ; cela se traduit par une liaison entre la ligne et la colonne correspondante à la tou-

```

00010 00001          NAM CLAVS
00020 00002          JPI NUF
00030 00003A 0010          ORG $10 Adresse de base
00040 00004          *
00050 00005          * LISTES D'EQUIVALENCES *
00060 00006          *
00070 00007          8004 A BASE EQU $8004
00080 00008          8004 A PIADCA EQU BASE DDRA/DRA
00090 00009          8005 A PIACRA EQU BASE+1 CRA
00100 00010          8006 A PIADB EQU BASE+2 DDRB/ORB
00110 00011          8007 A PIACRB EQU BASE+3 CRB
00120 00012          *
00130 00013A 0010 7F 8005 A          CLR PIACRA b2=0 Selection DDRA
00140 00014A 0013 7F 8007 A          CLR PIACRB b2=0 Selection DDRB
00150 00015A 0016 7F 8004 A          CLR PIADCA PORT A en Entree
00160 00016A 0019 86 FF A          LDAA #FF
00170 00017A 001B B7 8006 A          STAA PIADB PORT B en Sortie
00180 00018          *
00190 00019A 001E 86 04 A          LDAA #20000100
00200 00020A 0020 B7 8005 A          STAA PIACRA Acces DRA
00210 00021A 0023 B7 8007 A          STAA PIACRB Acces ORB
00220 00022          *
00230 00023A 0026 86 FE A          ENCORE LDAA #211111110
00240 00024A 0028 B7 8006 A          STAA PIADCE P50=0
00250 00025A 002B 86 04 A          LDAA #04 Initialisation compt
00260 00026A 002D 97 50 A          STAA $50
00270 00027A 002F 5F CLR:
00280 00028A 0030 B6 8004 A          RETOUR LDAA PIADCA
00290 00029A 0033 84 0F A          ANDA #200001111
00300 00030A 0035 81 0F A          CMFA #200001111
00310 00031A 0037 26 0A 0043 BNE COMPT
00320 00032A 0039 78 8006 A          ASL PIADCE
00330 00033A 003C 7A 0050 A          DEC $50
00340 00034A 003F 26 EF 0030 BNE RETOUR
00350 00035A 0041 20 E3 0026 BRA ENCORE
00360 00036A 0043 5C          COMPT INCB
00370 00037A 0044 44          LSRA
00380 00038A 0045 25 FC 0043 BCS COMPT
00390 00039A 0047 20 DD 0026 BRA ENCORE
00400 00040          END
TOTAL ERRORS 00000--00000
    
```

Fig. 10. - Ce programme détermine quelle touche de la matrice est enfoncée.

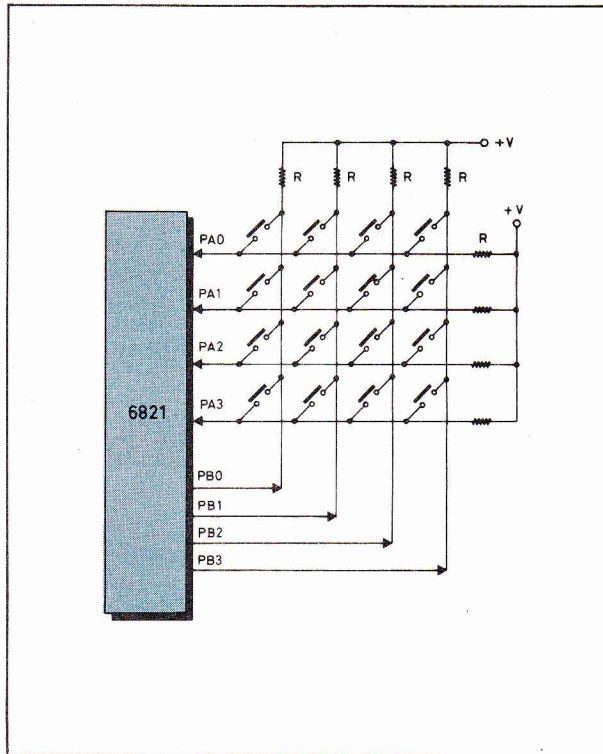


Fig. 11. - Liaison de la matrice « clavier » aux ports A et B du PIA.

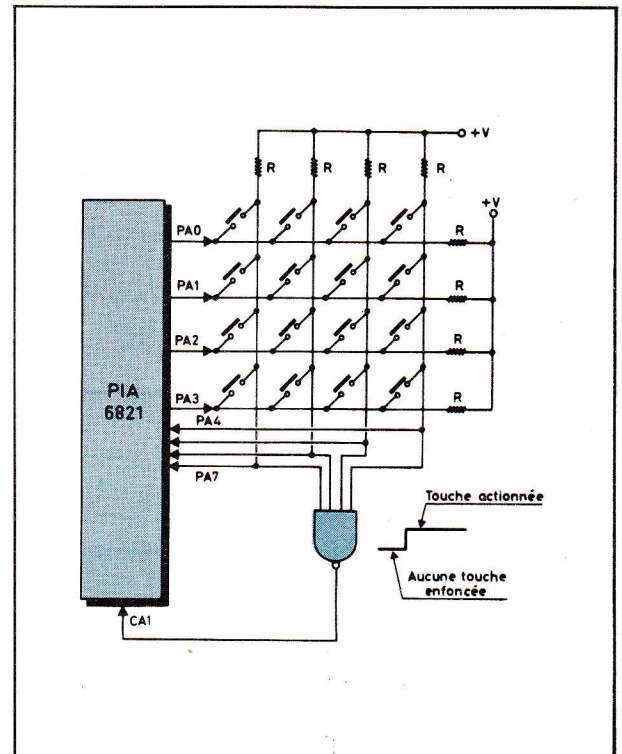


Fig. 12. - Principe de la méthode de « polling ». Dès qu'une touche est enfoncée, la sortie de la porte NAND impose un niveau logique « 1 » sur la ligne CA1. Ceci a pour effet de mettre à « 1 » le bit 7 du registre de contrôle du PIA.

LDS # \$ 00 FF	Initialisation du pointeur de pile
CLR PIACRA	b2 = 0 Sélection du DDRA
LDA A # \$ 0 F	Programmation PA ₀ à PA ₃ en sortie et de PA ₄ à PA ₇ en entrée
STA A PIADO A	Interruption masquée. Front montant
LDA B # % 0000 0110	Actif, accès au registre de donnée
STA B PIACRA	Actif, accès au registre de donnée
COM A	Complémentation du contenu de A
STA A PIADO A	Mise à zéro des lignes PA ₀ à PA ₃
LOOP LDA A PIACRA	Lecture du registre CRA
BPL LOOP	Branchement à Loop si N = 0 (pas d'impulsion sur CA1, b7 = 0)
JSR IDENT	Appel du sous-programme IDENTification de la touche sélectionnée
BRA LOOP	Retour pour une nouvelle lecture du PORT A
END	

Fig. 13. - Première partie du programme réalisant un « polling ». Vous pouvez rédiger vous-même le sous-programme qui identifiera la touche enfoncée.

* Polling : Appel sélectif. Technique consistant à interroger périodiquement chacun des terminaux qui se partagent une même ligne de communication, pour savoir s'il veut utiliser la ligne (Terminologie du traitement de l'information, IBM 80).

* La signification de chacun des bits du registre de contrôle d'un PIA apparaît de façon détaillée dans le numéro 12 (juillet-août 1980) page 86, de « Micro-Systèmes ».

* Ingénieur CNAM, Patrick Jaulent est responsable du département « Formation » de la Société Microprocess.

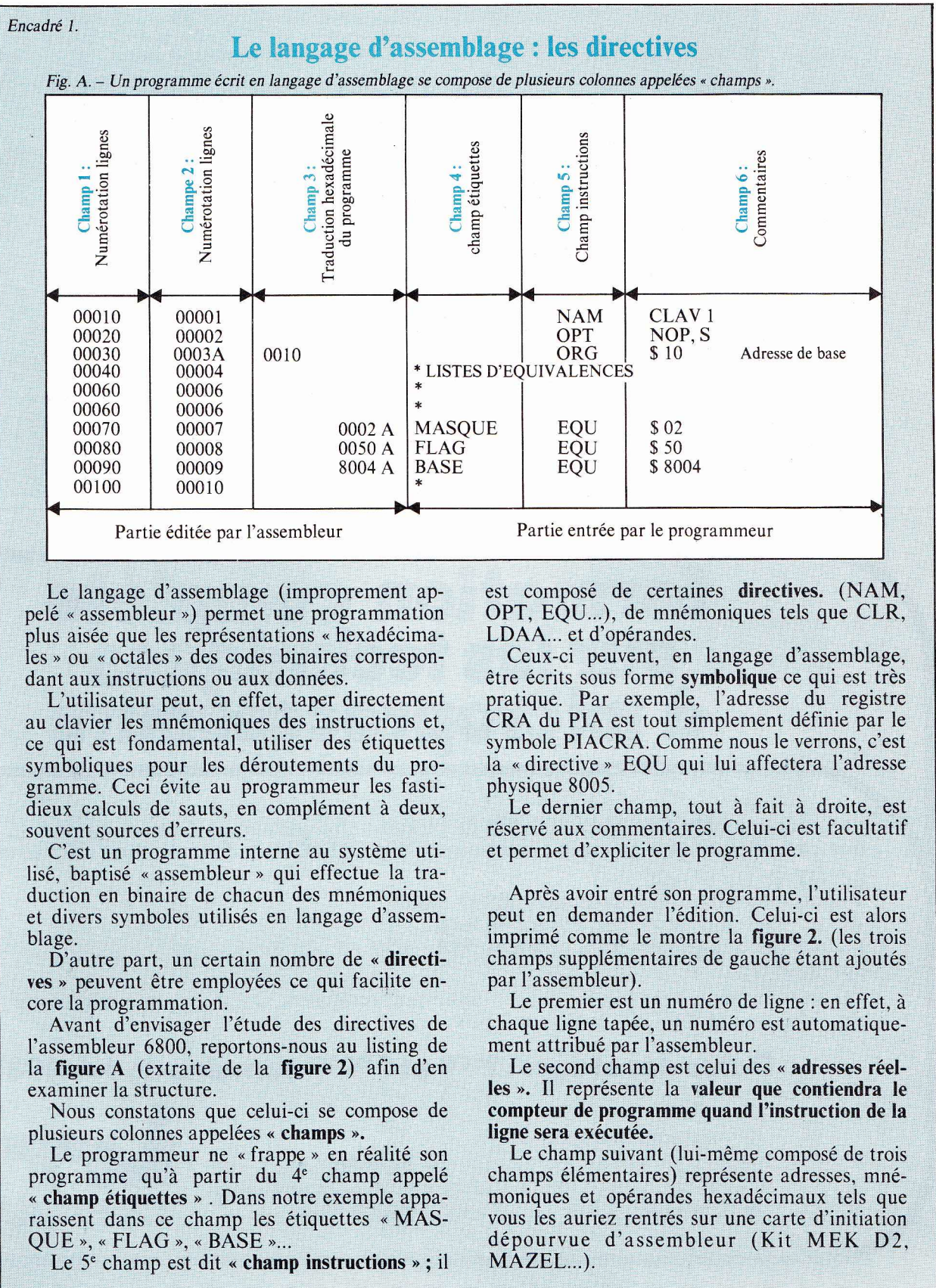
che sélectionnée, entraînant le passage à l'état haut de la ligne de commande CA1.

Nous savons que le bit 7 du registre de contrôle* est positionné à 1 sur une transition active (définie par le bit 1) arrivant sur CA1 (ou CB1).

Il est donc possible, après avoir programmé le choix du front actif, de déterminer si une impulsion est envoyée sur CA1, sans pour cela faire intervenir un programme d'interruption coûteux en cycles machine (pas de sauvegarde du contexte dans la pile avant le traitement du programme d'interruption proprement dit).

La figure 13 représente la première partie du programme de « polling » qui réalise la détection des touches enfoncées. Nous vous laissons le soin d'écrire maintenant le sous-programme d'identification de la touche (baptisée IDENT). ■

P. JAULENT *



Nous allons maintenant étudier certaines des directives de l'assembleur 6800.

Les directives d'assemblage

Une douzaine de directives peuvent être interprétées par l'assembleur 6800. Nous n'analyserons que celles figurant dans les programmes présentés ici.

EQU : Cette directive permet d'affecter une valeur à un symbole. Par exemple MASQUE prend, dans notre programme, la valeur \$ 02.

ORG : Définit l'adresse effective de début de programme. Le nôtre commence à l'adresse \$ 10 (ceci se retrouve dans le champ adresses hexadécimales du listing).

NAM : Attribue un **nom** au programme (que nous avons baptisé CLAV 1).

OPT : Permet de choisir une ou plusieurs **OPTIONS** parmi une dizaine (par exemple éditer une ligne particulière (**list**), diagnostiquer les erreurs (**error**)...

Dans notre exemple nous n'avons pas demandé d'option spéciale. En effet, ligne 20, nous retrouvons :

Le terme NOP signifie que nous ne désirons pas d'option spéciale et la virgule suivie de la lettre S ordonne l'édition de la « **table des symboles** » utilisés dans le programme. Celle-ci apparaît tout à fait au bas du listing, sur deux lignes. ■

formation complète sur les techniques à microprocesseurs

Le Laboratoire MICROPROCESSEURS de l'Institut Universitaire de Technologie de CRETEIL, organise, dans le cadre de la formation permanente DEUX STAGES de formation aux microprocesseurs.

CONTENU DU STAGE :

- * Cours matériel: le microprocesseur (6800 - 6809)
- * Cours logiciel (Assembleur, Editeur de texte...)
- * Cours entrée/sortie: les ports parallèles (PIA - ACIA)

PEDAGOGIE :

Les enseignements sont assurés par des ingénieurs enseignants. Ce stage pratique est assuré notamment par Monsieur MONTOIS - Ingénieur E.F.R.

Dates des stages: 1ère et 2ème semaine de Juin 1981

Renseignements et inscriptions :

INSTITUT UNIVERSITAIRE DE TECHNOLOGIE DE
CRETEIL - UNIVERSITE PARIS XII - Avenue du Général
de Gaulle 94010 CRETEIL Cedex - ☎ 899.80.40

IT
CRETEIL

Pour plus de précision cerchez la référence 135 du « Service Lecteurs »

La transmission de données

Si chacun d'entre nous peut définir une transmission il n'en n'est plus de même pour le mot « donnée ». Pourtant, il s'agit tout simplement d'une traduction ésotérique du mot information. Une information (nom d'une entreprise, chiffre d'affaires, numéro de compte bancaire...) prend le nom de donnée à l'instant où elle entre dans un système informatique.

D'autre part, ces données voyagent d'un ordinateur à l'autre, d'un pays à l'autre, car la gestion commerciale et financière des entreprises et des banques aiguillonnées par la concurrence, nécessite une quantité croissante d'informations (entreprise cherchant un créneau sur le marché) et de plus en plus vite (une information boursière a d'autant plus de valeur qu'elle a été transmise rapidement).

La transmission de données peut également rendre de grands services en diffusant les connaissances de l'humanité préalablement rassemblées en quelques points du globe, car les recherches deviennent longues et difficiles, voire infructueuses dans les bibliothèques et les archives qui ressemblent de plus en plus à des monstres appelés à disparaître.

Il existe actuellement plus de 1000 de ces points de stockage ou « banques de données » dans le monde. Mais que l'on ne se méprenne pas, bien que ce soit toujours des avantages d'ordre humanitaire qui sont mis en avant (données scientifiques à applications médicales par exemple), ce sont en fait des préoccupations plus prosaïques qui poussent la formidable machine d'expansion des transmissions de données ; les chiffres d'affaires américains sur les différentes banques de données en témoignent :

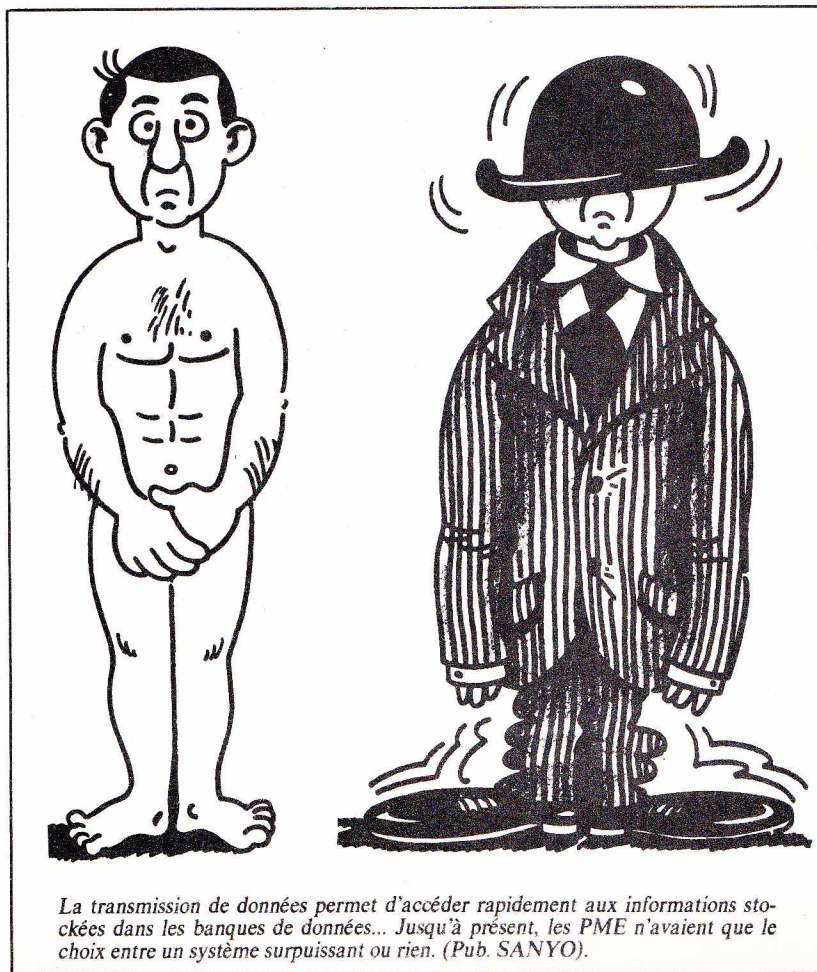
- 65 % (du chiffre d'affaires total) d'informations commerciales sur les entreprises et les marchés
- 16 % sur la solvabilité des consommateurs !...
- 8 % sur les données économiques
- 7 % sur les informations boursières
- 4 % pour le reste... dont les informations scientifiques...

Ainsi la transmission de données permet d'accéder rapidement aux informations stockées dans les banques de données mais là n'est pas sa seule utilité.

Les PME seront les premières à saisir la différence : jusqu'à présent elles n'avaient que le choix (si l'on peut dire) entre un système surpuissant ou rien, ce qui n'a pas échappé à un publiciste humoriste (il y en a).

Avec la transmission de données, il est possible de travailler en mode « conversationnel » (dit de façon impropre « temps réel »...) : plusieurs PME peuvent alors utiliser le même gros ordinateur qui travaille « en temps partagé ». Sa vitesse de travail étant très grande, il est capable de découper son temps de travail et d'organiser les temps morts pour « répondre à tout le monde ». Toutes les réponses paraissent simultanées mais il n'en est rien.

Une autre solution consistait à travailler en mode différé, c'est-à-dire que tous les jours, toutes les semaines ou tous les mois, les données sous forme de cartes, de rubans (perforés ou magnétiques) devaient être remises à des Centres informatiques.



La transmission de données permet d'accéder rapidement aux informations stockées dans les banques de données... Jusqu'à présent, les PME n'avaient que le choix entre un système surpuissant ou rien. (Pub. SANYO).

Lors d'une transmission, il faut synchroniser la fréquence d'horloge « réception » sur la fréquence d'horloge « émission ».

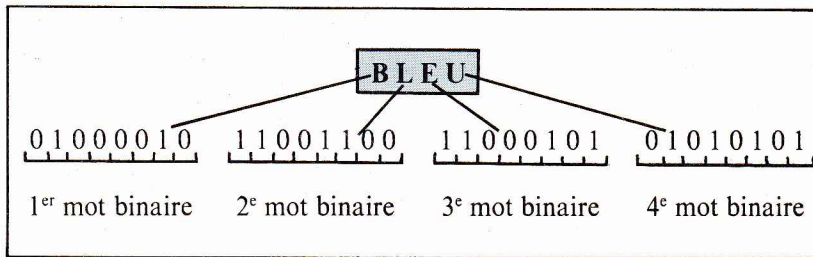
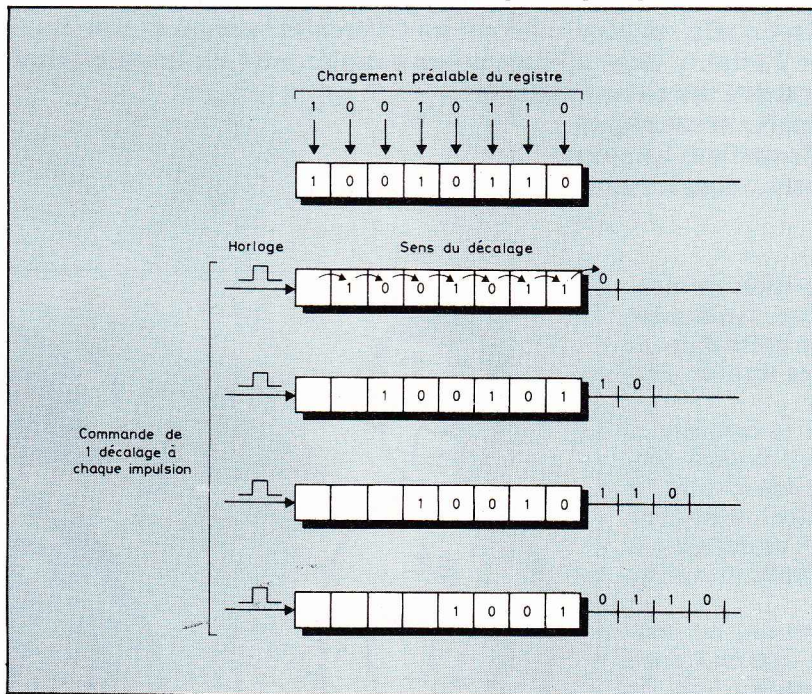


Fig. 1. - Chaque symbole de notre écriture (lettre, chiffre, signe) peut être remplacé par une succession de 0 et de 1. Dans cet exemple, on voit qu'il faut plusieurs mots binaires pour représenter le mot « BLEU » (codé en ASCII).

Fig. 2. - Il serait maladroit de vouloir transmettre simultanément les 8 bits d'un octet sur 8 fils différents. Le registre à décalage, en prenant en compte les 8 bits de l'octet et en les restituant un par un, réalise la transmission dite « série » sur un seul fil. Nous avons représenté ici les états successifs du registre à décalage après chaque impulsion.



Dans le meilleur des cas, sous quelques jours, le Centre renvoyait d'encombrants « listings » avec la courte période de validité que l'on sait. Cette méthode était loin d'être satisfaisante.

Les techniques de codage

L'ordinateur ne sait pas écrire dans ses mémoires la couleur « BLEU » par exemple, il ne sait écrire que 0 et 1.

Il est donc nécessaire de remplacer la lettre B par une succession de 1 et 0, le L par une succes-

sion différente de 1 et 0, etc.

Chaque succession de 1 et 0 constitue un mot binaire, terme peut-être mal choisi car on voit sur l'exemple de la **figure 1** qu'il faut plusieurs mots binaires pour faire le mot « BLEU ».

La correspondance entre lettres et successions de 1 à 0 est donnée par un « code ». On peut imaginer des codes à l'infini, mais n'allons pas si loin car IBM a pratiquement imposé l'usage du code ASCII*. A chaque caractère, (lettres de l'alphabet, chiffres de 0 à 9, plus un certain nombre de signes comme : +, =, /, etc.) correspond une succession de 7 bits.

Le bit étant une entité binaire pouvant prendre la valeur 1 ou 0, le nombre de combinaisons possibles est $2^7 = 128$. Le code standard ne comprend que 96 symboles et les possibilités des claviers et imprimantes limitent encore ce nombre bien souvent. Enfin aux 7 bits de codage est ajouté un bit de parité, de telle sorte que le nombre de 1 dans l'octet soit un nombre pair, par exemple.

Un autre code fréquemment utilisé est le code EBCDIC* qui utilise 8 bits de codage sans bit de parité, ce qui porte le nombre de caractères possibles à 256.

Les **tableaux 1 et 2** représentent les codes ASCII et EBCDIC.

Il existe, nous l'avons dit, de nombreux codes (BCD, baudot, réfléchi...) et le passage de l'un à l'autre ou **transcodage** n'est pas toujours possible. (En effet, si le nombre de bits de codage est différent dans les deux codes, il ne peut y avoir bijection).

Ainsi, un caractère est généralement codé par 8 bits et chacun de ces bits est matérialisé par une bascule dont la tension de sortie délivre 0 V ou + 5 V.

Huit bascules correspondent à 8 mémoires élémentaires pouvant mémoriser un octet.

Les techniques de transmission

● Mode d'échange

Pour transmettre des informations (données) d'un ordinateur à l'autre il va nous falloir transmettre les octets (groupe de 8 bits) les uns après les autres.

Nous pourrions transmettre simultanément les 8 bits de l'octet à travers 8 fils. Ceci se fait par des conducteurs à 8 fils appelés « bus » à l'intérieur de l'ordinateur ou

(*) ASCII : acronyme de « American Standard Code for Information Interchange » (code standard américain pour l'échange des informations).

* Acronyme de « Extended Binary Coded Decimal Interchange Code ».

pour de très courtes distances (de l'ordre du mètre jusqu'à 2 à 3 km). C'est la transmission dite « parallèle ». Mais si l'on veut transmettre à plusieurs kilomètres il est clair que cette méthode devient lourde et vite coûteuse. Il faut alors recourir à la transmission dite « série » qui consiste à envoyer les octets bit par bit sur un seul fil. Cette formule nécessite toutefois des circuits de conversion parallèle-série tel celui de la figure 2.

Les 8 bascules dans lesquelles est stocké notre mot de 8 bits constituent ce que l'on appelle un registre.

Il s'agit ici d'un « registre à décalage » : les 8 bits sont chargés simultanément dans le registre mais ne sortent que l'un après l'autre, à la « queue-leu-leu », au rythme d'un signal appelé « horloge », véritable métronome d'émission, qui commande le décalage.

Notons que le registre à décalage est un dispositif parfaitement réversible, ce qui permet, à la réception, de faire l'opération exactement inverse. Lorsque le registre

Col.	0	1	2	3	4	5	6	7
0	0	0	0	0	1	1	1	1
1	0	0	1	1	0	0	1	1
2	0	1	0	1	0	1	0	1
3	0	1	2	3	4	5	6	7
4	0	0	0	0	0	0	0	0
5	0	0	0	1	1	0	0	1
6	0	0	1	0	2	B	R	b
7	0	0	1	1	3	C	S	c
8	0	1	0	0	4	D	T	d
9	0	1	0	1	5	E	U	e
10	0	1	1	0	6	F	V	f
11	0	1	1	1	7	G	W	g
12	1	0	0	0	8	H	X	h
13	1	0	0	1	9	I	Y	i
14	1	0	1	0	10	J	Z	j
15	1	0	1	1	11	K		k
16	1	1	0	0	12	L		l
17	1	1	0	1	13	M		m
18	1	1	1	0	14	N		n
19	1	1	1	1	15	O		o

Tableau I. - Le Code ASCII ou CCITT N° 5 établit une relation entre un caractère et un mot binaire de 7 bits.

Tableau II. - Le code EBCDIC utilise 8 bits de codage.

La liste des caractères de commande est donnée à ce mot.

D'autres commandes ont été ajoutées :

- PF perfo hors fonction
- LC en code
- RES restauration
- IL neutre
- BYP ignorer
- LF interligne
- SM changement de mode
- PN perfo en fonction
- RS arrêt lecteur
- UC hors code
- SP espace

Col.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1
2	0	0	1	1	0	0	1	1	0	0	0	1	0	0	1	1
3	0	1	0	1	0	1	0	1	0	1	1	1	0	1	0	1
4	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5	NUL	DLE			SP	&	—						{	}	\	0
6	SOH	DC1				/		a	j				A	J		1
7	STX	DC2	FS	SYN				b	k	s			B	K	S	2
8	ETX	DC3						c	/	t			C	L	T	3
9	PF	RES	BYP	PN				d	m	u			D	M	U	4
10	HT	NL	LF	RS				e	n	v			E	N	V	5
11	LC	BS	ETB	UC				f	o	w			F	O	W	6
12	DEL	IL	ESC	EOT				g	p	x			G	P	X	7
13			CAN					h	q	y			H	Q	Y	8
14			EM					i	r	z			I	R	Z	9
15			SM	∅	!	:										
16	VT			.	\$,	#									
17	FF	IFS		DC4	<	*	%	@								
18	CR	IGS	ENQ	NAK	()	—	'								
19	SO	IRS	ACK		+	;	>	=								
20	SI	IUS	BEL	SUB		~	?									

La rapidité de modulation est l'inverse du temps le plus court existant entre deux niveaux de modulation.

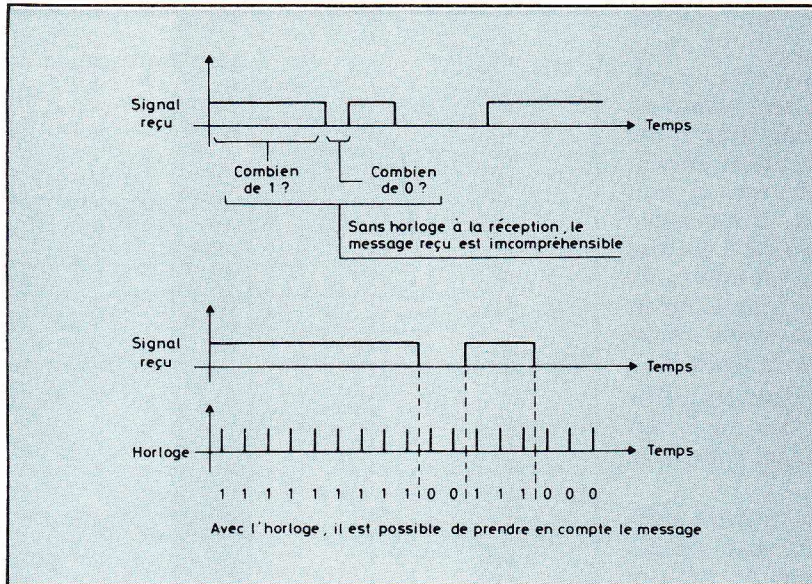
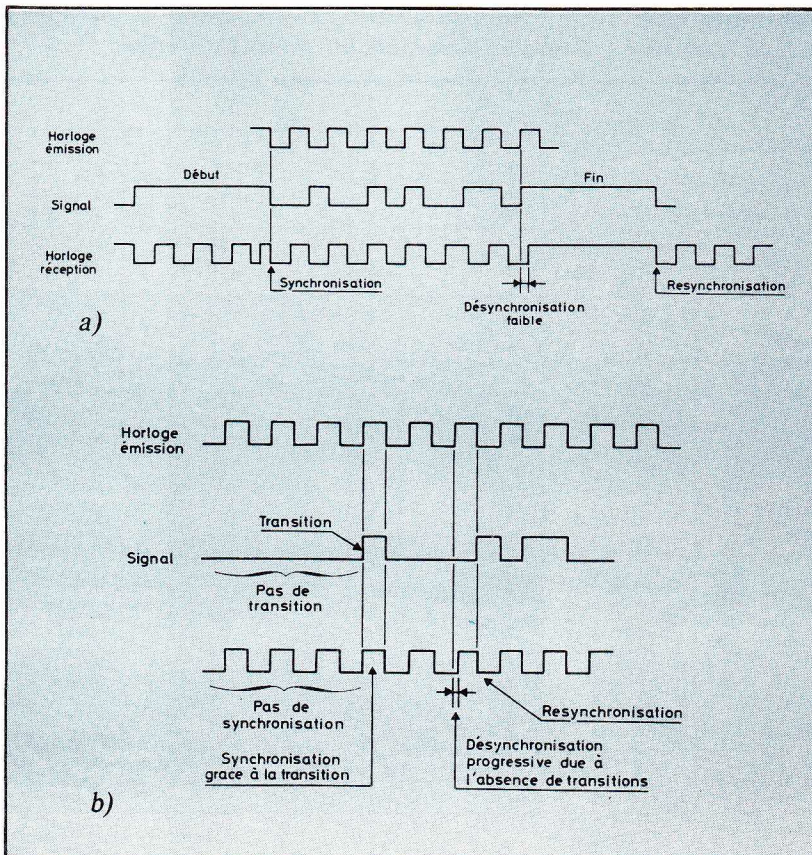


Fig. 3. - Le signal transmis est indissociable de l'horloge; sans cette dernière, il est inutilisable.

Fig. 4. - Pour recevoir correctement un signal, il faut restituer une horloge à la réception ayant une fréquence la plus proche possible de l'horloge d'émission. En mode asynchrone (a), les octets sont groupés en « messages » et sont précédés d'un signal de « DEBUT » et terminés par une « FIN ». Le DEBUT sert à synchroniser l'horloge réception pour la durée de message. En mode synchrone (b), l'horloge « réception » est synchronisée en permanence grâce aux transitions du signal.



a été rempli bit par bit, il est vidé « en parallèle » sur un bus de 8 fils restituant ainsi l'octet sous sa forme primitive.

● La synchronisation

Les difficultés commencent à la réception. En effet, si le signal reste à + 5 V * pendant dix secondes, à combien de bits de valeur 1 ceci correspond-il ? Nous n'avons que l'embarras du choix : si la durée d'un bit est de une seconde, il y en a dix, si un bit dure deux secondes, il n'y en a que cinq, etc.

Quelle est donc la durée du bit ? Nous avons vu qu'elle était définie par l'horloge à l'émission. Il est donc indispensable d'avoir cette horloge à la réception pour décoder le message reçu (fig. 3). Mais, comment obtenir cette horloge, qui n'est pas transmise parallèlement au signal de données ? Nous allons voir qu'elle est implicitement et partiellement transmise « dans les données ».

La bonne méthode consiste à créer à la réception une horloge ayant une fréquence approchant celle de l'horloge émission. Puis, à l'aide des informations contenues dans le signal reçu, les tops horloge réception sont calés sur les bits.

Cette synchronisation réalise automatiquement l'égalité des fréquences horloge émission et réception par réajustement continu comme le montre le figure 4.

En mode asynchrone

Les octets sont groupés en « messages » ou « trames » précédés d'un « DEBUT* » et terminés par une « FIN** ». Le « DEBUT » sert à mettre l'horloge réception au pas des données et ce pour la durée du message. Il ne faut donc pas que les messages soient trop longs car entre deux « DEBUT » l'horloge réception ne recevant pas

* En supposant par exemple la correspondance suivante :

1 → + 5 V
0 → 0 V.

* Traduit souvent par « Start »

** Traduit souvent par « Stop »

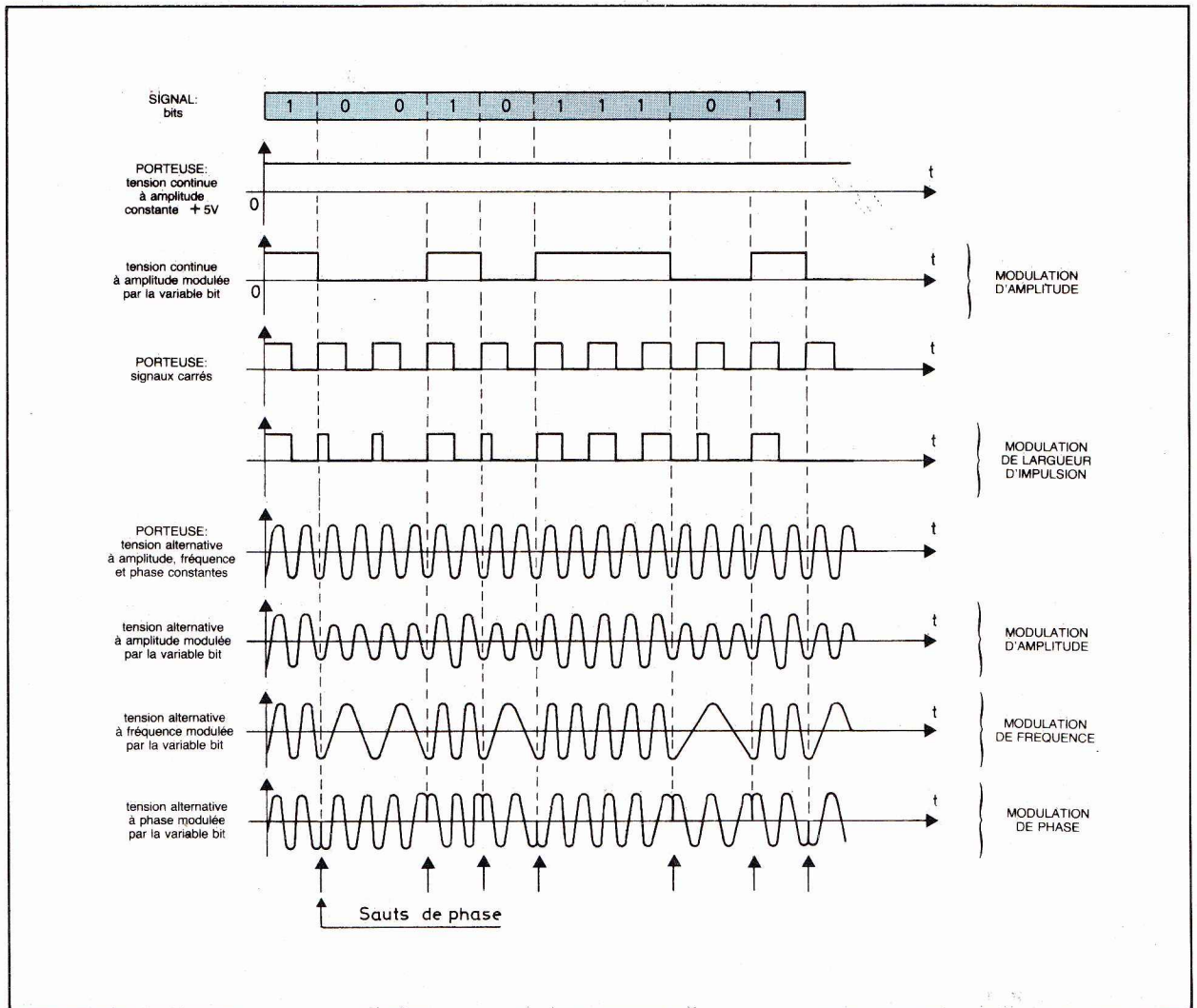


Fig. 5. - Trois porteuses différentes (tension continue, signaux carrés et tension alternative) sont modulées par un signal binaire (1001 0111...). Nous avons représenté les modulations les plus souvent rencontrées : modulation d'amplitude, de largeur d'impulsion, de fréquence et de phase. Le support de modulation (porteuse) dépend beaucoup du support de transmission : sur ligne téléphonique il est possible d'envoyer soit un courant continu, modulé ou non, soit une porteuse à fréquence relativement basse ; dans le cas d'un faisceau optique ou hertzien il s'agit impérativement d'une onde porteuse de fréquence élevée (bande des MHz).

de « calage » risque de se déphaser par rapport à l'horloge « émission » au-delà des limites permises. Il convient de remarquer que, du fait des informations supplémentaires « DEBUT » et « FIN » et du décalage d'horloge inhérent au système, il est impossible de transmettre de façon fiable à grande vitesse (à partir de 2 400 bits/s il faut utiliser la transmission synchrone).

En mode synchrone

L'horloge «réception » est « mise au pas » en permanence

grâce aux transitions du signal (1-0 ou 0-1) qui correspondent forcément à un changement de bit.

Il est donc possible à la réception de faire fonctionner un métronome sur un rythme « approximatif » dans un premier temps, puis de le « mettre au pas » à la réception des premières transitions.

Les premières données seraient alors perdues... c'est vrai, c'est pourquoi elles ne sont pas envoyées immédiatement mais précédées d'un message de synchronisation préalable.

Ce mode de transmission est in-

teressant pour les débits élevés mais comporte quelques inconvénients auxquels, rassurez-vous, l'électronique apporte une solution élégante ; en effet, si l'émetteur envoie un 1 ou 0 permanent, il n'y a pas de transition et la synchronisation de l'horloge réception ne peut s'établir.

Il est donc prévu un « limiteur d'état permanent » à l'émission qui, au n-ième bit identique inverse systématiquement tous les bits suivants. Le récepteur a lui aussi compté n bits successifs identiques, il « sait » donc qu'il y a eu inversion.

La modulation

Moduler un signal consiste à modifier une grandeur physique (fréquence, tension, brillance,...) en fonction d'une variable (le bit dans le cas qui nous intéresse). La grandeur physique est le support de modulation et, si cette grandeur physique appartient à une onde nous dirons qu'il s'agit d'une **onde porteuse** (plus familièrement « **porteuse** »).

Il est possible d'imaginer une infinité de modulations. Les plus courantes (modulations d'amplitude, de fréquence, de phase) sont représentées **figure 5**.

En outre, une modulation porte le qualificatif de « **différentielle** » lorsque c'est le changement d'amplitude, de fréquence, de phase, etc., qui caractérise l'une des deux valeurs 0 ou 1. Différentes modulations peuvent être utilisées simultanément.

Bien entendu, le choix du type de modulation est fonction du débit d'informations à transmettre.

La modulation de fréquence autorise des débits de l'ordre de 1 200 bits/s ; la modulation d'amplitude ou de phase, 9 600 bits/s et la modulation dite « **de largeur d'impulsion** » est capable de véhiculer 250 000 bits/s.

Néanmoins, les modulations les plus performantes quant au débit sont aussi les plus sophistiquées sur le plan de l'électronique. L'électronicien devra une fois de plus manipuler la balance des compromis : ainsi la modulation de largeur d'impulsion (et toutes les modulations homologues) est vite épuisée par un voyage, la modulation d'amplitude ne supporte pas le bruit et la modulation de phase ne sait pas danser la « gigue * » au-delà d'une certaine valeur. Dans certains cas, il faudra bien, si l'on désire une transmission fiable, se contenter de la modulation de fréquence avec un faible débit.

* La gigue est le déplacement parasite, variable de phase.

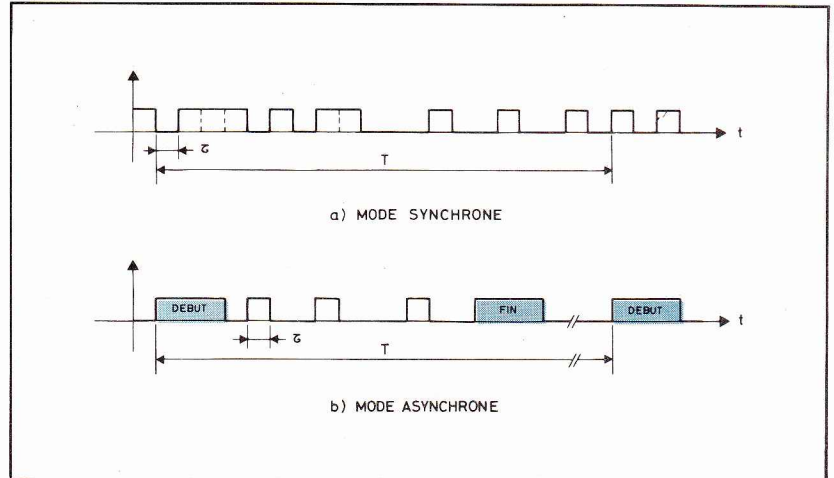


Fig. 6. - La rapidité de modulation. En mode synchrone (a), l'intervalle de temps le plus petit τ (ou temps élémentaire) permet d'exprimer la rapidité de modulation (ou fréquence baud) par :

$$R = \frac{1}{\tau} \text{ en bauds.}$$

Dans notre exemple, la fréquence bit (20 bits/T) est égale à la fréquence baud.

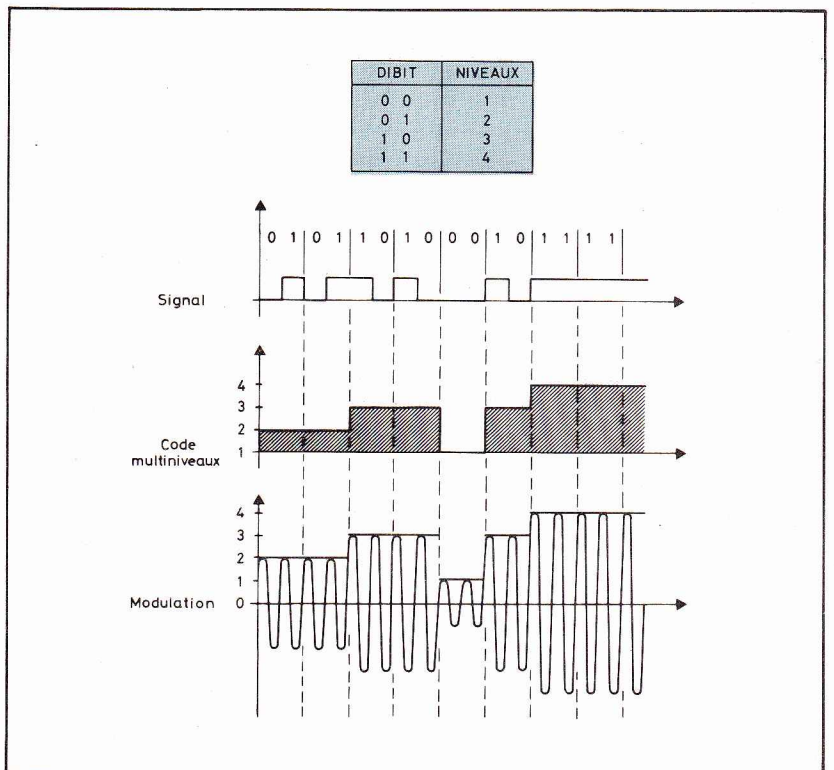
En mode asynchrone (b), la fréquence bit est altérée par la présence des DEBUT et FIN et éventuellement par des « trous de transmission » entre une FIN et un DEBUT : elle vaut ici 11 bits/T alors que la fréquence baud exprimée sur la même période

$$R = \frac{20}{T} = \frac{1}{\tau} \text{ ne change pas}$$

Fig. 7. - Code et modulation multi-niveaux, nous avons représenté un exemple de code et de modulation d'amplitude à 4 niveaux.

Il suffit de grouper les bits (ici 2 par 2) pour obtenir 4 combinaisons possibles donc 4 niveaux distincts.

L'écart entre deux bits est deux fois plus petit que l'écart entre deux niveaux de modulation, la fréquence baud est donc (en mode synchrone) deux fois plus faible.



Rapidité de modulation

La rapidité de modulation est l'inverse du temps le plus court existant entre deux niveaux de modulation ; cette grandeur est donc homogène à une fréquence et s'exprime en **baud**. Cependant, il faut faire très attention (**fig. 6**) ; le nombre de baud ne correspond pas forcément au nombre de bits par seconde : ainsi en mode asynchrone, les « DEBUT » et « FIN » modifient la « fréquence bit » mais n'affectent pas la « fréquence baud » et, en code multi-niveaux nous verrons que les fréquences baud et bit sont différentes... pour une autre raison.

Le débit binaire

Le débit binaire est directement lié aux caractéristiques du **support de transmission**, et s'exprime en bits par seconde.

Le débit binaire maximum envisageable pour un support est donné par la formule de Shannon :

$$C = W \lg_2 (1 + S/B)$$

où :

C est le débit binaire maximum
W est la largeur de bande de fréquence utilisée

\lg_2 est le logarithme à base 2

S est la puissance du signal

B est la puissance du bruit.

Le débit binaire prend aujourd'hui une importance capitale car le flot d'informations délivré par les ordinateurs est souvent considérable et la ligne téléphonique ne peut pas suivre...

C'est alors que les codes multi-niveaux vont paraître très intéressants.

Codes et modulations multi-niveaux

Code et modulation sont étroitement liés. Ainsi, il est toujours possible de remplacer nos deux niveaux binaires habituels par un certain nombre de niveaux mais il

nous faudra aussi le même nombre de niveaux de modulation.

Comment obtenir un code multi-niveaux alors que l'ordinateur ne travaille qu'en binaire ? En prenant les bits par groupe de 2, 3, 4, etc.

Examinons le cas le plus simple et supposons que nous ayons groupé les bits deux par deux. Les combinaisons possibles sont (0,0) ; (0,1) ; (1,0) ; (1,1) et la modulation d'amplitude correspondante s'effectuera par conséquent comme l'indique la **figure 7** avec 4 niveaux.

Dans notre exemple, nous avons choisi la modulation d'amplitudes pour faciliter la compréhension bien qu'en pratique l'on utilise la modulation de phase ou de fréquence (avec des niveaux de phases ou de fréquences).

Quel que soit le type de modulation envisagé, le nombre de transitions est deux fois plus faible. Il est donc possible de doubler le débit. On serait alors tenté d'augmenter le nombre de niveaux pour accroître le débit mais l'écart entre deux niveaux étant plus faible, si des signaux parasites du même ordre de grandeur que cet écart s'insèrent dans le signal, on imagine sans peine le résultat *. Ainsi, les codes multi-niveaux, s'ils autorisent des débits plus élevés, comportent aussi un risque d'erreur plus important et la fiabilité de la transmission dépend de ce compromis.

A la limite, si l'on augmente le nombre de niveaux indéfiniment, nous obtenons un signal analogique, il s'agit même d'une conversion numérique-analogique. Les codes multi-niveaux n'étant en fin de compte qu'un aspect particulier de la conversion numérique-analogique.

Procédures de dialogue

Il est évident que le terminal * ne peut envoyer des données sans se soucier de ce qu'il en advient. Un dialogue terminal-modem doit s'établir avant toute émission de

données. Ce dialogue étant plus ou moins important suivant les liaisons.

Les modems émetteur et récepteur peuvent être reliés soit par le réseau téléphonique normal (ce qui nécessite la connexion à la ligne et l'appel du correspondant), soit par des liaisons spécialisées (qui sont le plus souvent des lignes téléphoniques désolidarisées du réseau afin de les rendre spécifiques à la transmission de données). Dans le premier cas nous dirons que le modem est sur « **réseau commuté** » et dans le deuxième cas, sur « **réseau spécialisé** ».

Écoutons maintenant quel genre de dialogue est échangé entre modem et terminal.

Dans le cas du réseau commuté, le terminal « dit » au modem :

- « **Connectez-vous à la ligne.** »

Si la ligne est libre (pas de conversation téléphonique en cours), le modem appelle son correspondant et répond :

- « **Je suis connecté.** »

Puis le dialogue se poursuit, quel que soit le type de réseau cette fois, de la façon suivante :

S'il désire émettre des données, le terminal questionne le modem :

- « **Etes-vous prêt à émettre ?** »

Lorsque la réponse est affirmative, le modem émetteur transmet la porteuse de façon à ce que le modem récepteur puisse se synchroniser.

Après une temporisation jugée suffisante pour que la synchronisation s'effectue, le modem répond :

- « **Je suis prêt.** »

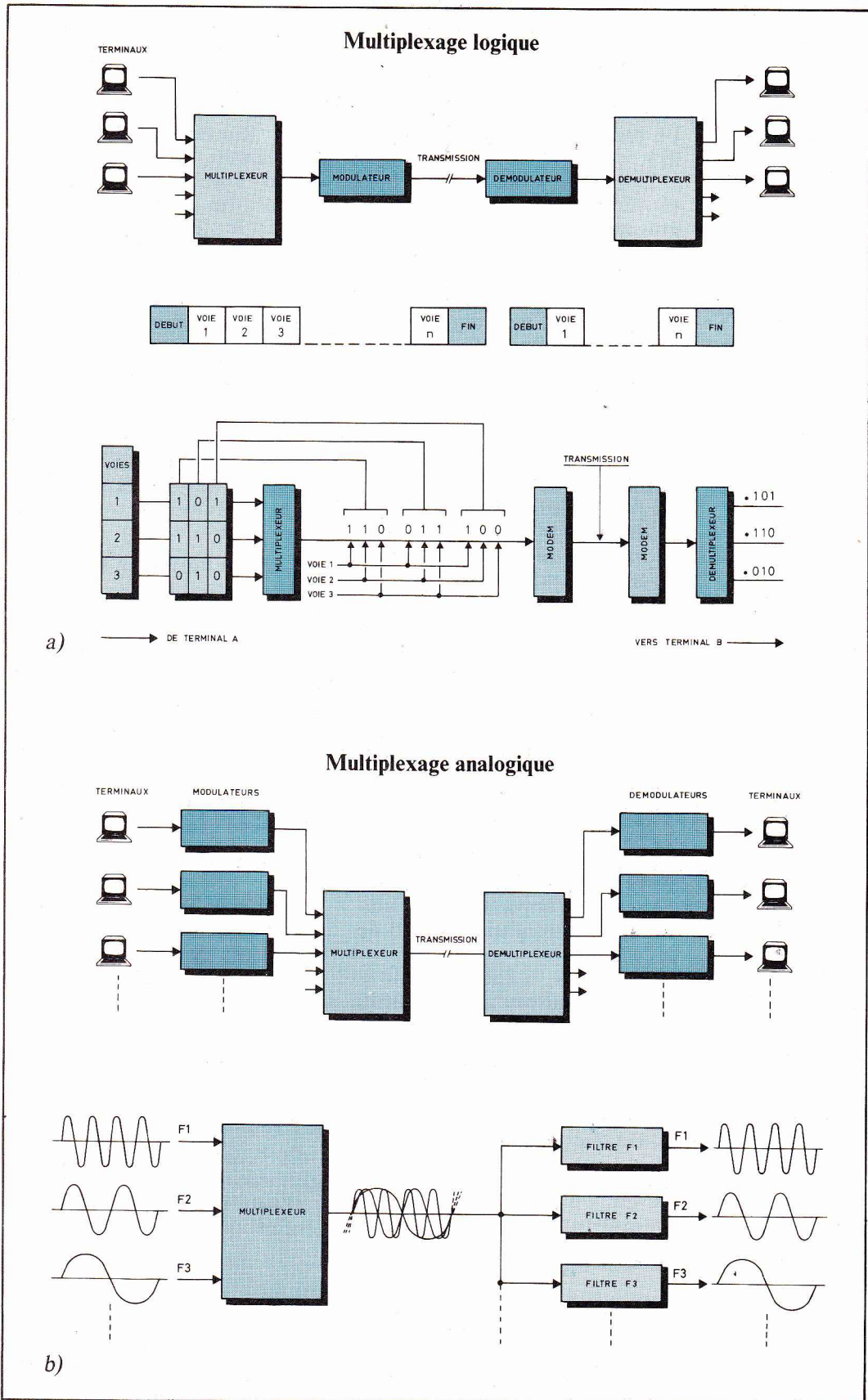
Et l'échange de données peut commencer.

D'autres échanges, à des fins de vérification de la transmission,

* Ceci est aussi valable pour les modulations de fréquence (dérive parasite) ou de phase (gigue parasite).

* Le mot terminal est ici pris au sens large, il peut s'agir d'un ordinateur ou d'un périphérique (console par exemple).

Si l'on augmente le nombre de niveau indéfiniment, nous obtenons un signal analogique...



sont réalisés dans le cadre de tests très utiles et dont la complexité au niveau des circuits ou des logiciels dépasse parfois celle des circuits de modulation et de démodulation.

Le multiplexage

Le multiplexage est un procédé offrant la possibilité de transmettre plusieurs voies simultanément en les regroupant en une seule.

Le démultiplexage est l'opération inverse : la voie unique redonne les n voies de départ.

Il existe deux types de multiplexage : le multiplexage numérique et le multiplexage analogique.

La figure 8 montre comment les multiplexeurs s'insèrent dans un réseau de transmission.

Le multiplexage numérique est un multiplexage « bit à bit » dans lequel chaque trame comprend un bit de la voie 1, puis un bit de la voie 2, puis un bit de la voie 3, etc. Afin de repérer les débuts et fins de trames des codes spécifiques sont utilisés. Ce sont les trames ainsi obtenues qui serviront ensuite à moduler éventuellement une onde porteuse.

En ce qui concerne le multiplexage analogique, les données émises sur chaque voie sont utilisées pour moduler une porteuse, chaque porteuse ayant une fréquence différente des autres. Toutes les porteuses sont transmises simultanément.

A la réception, des filtres sélectionnent les porteuses en fonction de leur fréquence, opérant ainsi le démultiplexage. ■

D. de BEAULIEU

Fig. 8. - L'insertion d'un multiplexage logique (a) ou analogique (b) dans un réseau de transmission.

BASIC et mathématiques

La résolution des équations différentielles

Les mathématiques sont un outil précieux pour le scientifique qui cherche à coordonner entre elles des observations afin d'appréhender les lois de l'univers et d'en tirer ainsi la « substantifique moëlle ».

L'idée n'est pas nouvelle. Les pythagoriciens déjà, quelque cinq siècles avant J.C. prétendaient que le nombre représentait l'ultime réalité accessible à l'esprit humain.

Pour cette raison, l'étude des phénomènes de la nature, de la chute d'une pierre du sixième étage à l'évolution délicate d'une souche bactérienne, font apparaître des fonctions mathématiques pour décrire leur évolution dans le temps et l'espace.

De très nombreux phénomènes naturels peuvent s'exprimer au moyen d'équations, appelées équations différentielles.

Tous ceux d'entre vous qui ont peiné de longues heures à tenter la résolution d'une équation du type :

$$\frac{dy}{dx} = y^2 - x^2 + \sin [\log (x^2)] - \frac{y}{x} \text{Arc tg} (y^2 - x^2)$$

apprécieront pleinement les résultats obtenus, à la fin de cet article, en quelques secondes...

De l'équation la plus simple à la plus complexe...

L'expérience montre que la plupart des lois physiques qu'on rencontre dans la nature peuvent se mettre sous la forme d'une fonction $Y = F(x)$.

La plus simple de ces fonctions correspond à l'équation d'une droite dont l'expression de la forme :

$$e = vt$$

exprime par exemple la proportionnalité entre le temps et l'espace parcouru, e , pour un mobile se déplaçant à vitesse constante. La plus complexe reste certainement l'équation de Schrödinger décrivant la probabilité de présence d'un électron autour du noyau atomique.

Entre ces deux extrêmes on rencontre un grand nombre de fonctions mathématiques explicitant la plupart des phénomènes physiques, tels la sinusoïde ($y = a \sin x$), l'exponentielle ($y = a \exp(x)$) et le logarithme ($y = \log(x)$).

Un exemple élémentaire

« Un automobiliste se déplaçant sur une route désire connaître sa vitesse précise. Pour ce faire, il décide de se chronométrer entre deux bornes kilométriques. »

Il note l'écart de temps écoulé entre les deux bornes Δt . Si Δx

représente la distance séparant ces deux bornes, le conducteur calcule sa vitesse en effectuant l'opération très simple :

$$v = \frac{\Delta x}{\Delta t}$$

Mais cette expression ne donne pas une bonne valeur de la vitesse du véhicule à un instant précis (sauf si la vitesse est constante).

En fait, V représente ici une valeur moyenne de la vitesse durant l'intervalle de temps Δt .

Ainsi, pour obtenir une meilleure approximation de V il faudra réduire le plus possible Δt . A la limite, la valeur exacte de la vitesse à l'instant considéré t sera la plus précise lorsque Δt tendra vers 0.

Δx et Δt peuvent en réalité être pris aussi petit que l'on désire, et l'écriture Δx et Δt devient alors dx et dt .

Cette opération donne une bonne idée du calcul différentiel. En effet, V la vitesse introduite par la division $\Delta x/\Delta t$, lorsque Δt tend vers 0, devient la dérivée première de la distance (ou espace) par rapport au temps.

La vitesse vraie à l'instant t :

$$v = \frac{dx}{dt}$$

s'exprime alors comme la limite du rapport $\Delta x/\Delta t$ lorsque les deux nombres tendent vers 0. Dès lors que les distances et les temps sont infinitésimaux, il devient possible, mathématiquement du moins, de définir des mouvements instantanés.

Dans notre exemple, si la vitesse est constante (ce qui n'est pas nécessairement le cas), alors $x = vt + x_0$ est la solution de l'équation différentielle.

Trouver une solution à une équation différentielle s'appelle intégrer cette équation.

Examinons maintenant un exemple un peu plus compliqué « d'intégration » d'équations différentielles.

Les équations différentielles du 2^e ordre

Une tuile tombant du toit d'un immeuble de six étages, parce qu'attriée par l'énorme masse de la Terre, est soumise à une force que lui communique une accélération g . Ce qui se traduit mathématiquement par la célèbre relation de Newton :

$$F = mg = GmM/R^2 \quad (1)$$

m est la masse de la tuile exprimée en kg

g est l'accélération par la pesanteur : 9,81 m/s²

M est la masse de la Terre en kg

R est le rayon de la Terre en m

G est la constante universelle de la gravitation 6,62 10⁻¹¹ m³ · kg/s²

F la force qui résulte du potentiel lié aux masses : en Newton.

La hauteur d'un sixième étage

est ridiculement petite à côté du rayon de la Terre de sorte que :

$$g = GM/R^2$$

peut être considérée comme une constante...

Ce raisonnement montre, entre autre, que l'allure du mouvement d'un objet en chute libre, abstraction faite des frottements de l'air, est indépendante de sa masse, ce qui n'était pas évident a priori.

Or, l'accélération (la vitesse à laquelle varie la vitesse), est définie comme la **dérivée de la vitesse par rapport au temps** que l'on note :

$$g = \frac{dv}{dt}$$

Elle peut se déterminer de la même façon que la vitesse de notre automobiliste en mesurant v_2 et v_1 entre deux instants t_2 et t_1 . La vitesse, comme on l'a déjà dit, est la dérivée de l'espace par rapport au temps, l'accélération, la dérivée de la vitesse par rapport à ce même temps. Or la dérivée d'une dérivée s'appelle une DERIVÉE SECONDE. L'accélération est donc définie comme la dérivée seconde de l'espace par rapport au temps.

Ceci se note :

$$V = \frac{dx}{dt} = x' \quad (1)$$

$$g = \frac{dv}{dt} = \frac{d^2x}{dt^2} = x'' \quad (2)$$

g étant une constante, l'intégration de (2) conduit à :

$$v = g \int dt = gt + v_0 \quad (3)$$

où v_0 est la vitesse à l'instant initial $t = 0$. Celle-ci pouvant être nulle.

En intégrant (3) on obtient

$$x = \int v dt = \int (gt + v_0) dt$$

ce qui donne

$$x = \frac{1}{2}gt^2 + v_0 t + x_0$$

x_0 étant la position du mobile à $t = 0$.

C'est par ce calcul, que l'on démontre la justesse des observations de Galilée qui mesurait le temps de chute de billes de plomb à différents étages de la Tour de Pise et où il observait que l'espace

parcouru est proportionnel au carré du temps de chute.

Nombreuses sont les branches de l'activité humaine où il est fait appel à la mathématisation d'une quantité afin d'en déterminer, toutes les fois que c'est possible, l'évolution quantitative d'un phénomène.

Souvent les équations différentielles, d'ordre 1 ou 2 ont des solutions analytiques exactes. On entend par là, que l'équation différentielle est directement intégrable. Mais il arrive parfois qu'il soit impossible d'exprimer directement la solution par des fonctions mathématiques connues. Par exemple, l'équation différentielle complète, régissant le mouvement du pendule qui s'exprime en fonction de l'angle θ par la relation :

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l} \cdot \sin \theta$$

n'est pas directement intégrable, mais on peut évidemment trouver un développement qui permet de calculer avec une bonne approximation la période d'oscillation du pendule.

Cependant, si vous voulez vérifier que votre solution est exacte il est intéressant d'utiliser le calcul numérique offrant la possibilité, par itérations, d'approcher avec une grande précision la **fonction** que vous étudiez.

Pour cette raison le calcul numérique approché des équations différentielles a été mis au point afin de s'adapter aux puissants outils que sont les ordinateurs.

Un grand nombre de techniques différentes permettent de résoudre **numériquement** les équations différentielles. La plus connue d'entre elles s'appelle méthode de **Runge-Kutta**.

C'est cette dernière que nous nous proposons de développer plus en détail.

La méthode de Runge-Kutta

Pour simplifier l'exposé et prendre modèle sur les Anglo-Saxons nous allons décrire cette méthode

en utilisant l'exemple très simple de l'équation différentielle :

$$\frac{dy}{dx} + y = e^x$$

ou

$$y' = e^x - y$$

L'utilisation d'une méthode numérique impose la spécification des conditions initiales d'intégration. Ainsi pour $Y_0 = 0$ nous posons $X_0 = 0$.

La méthode employée consiste à faire correspondre à un incrément de la variable x , une **moyenne pondérée** de n évaluations. L'équation régissant ce processus s'exprime de la manière suivante :

$$y_{i+1} = y_i (k_1 + 2k_2 + 2k_3 + k_4)/6$$

les valeurs y_i et y_{i+1} correspondent à deux valeurs successives de la fonction $Y = f(x)$ avec :

$$y_i = f(x_i) \text{ et } y_{i+1} = f(x_{i+h})$$

les coefficients k_1, k_2, k_3 et k_4 sont déterminés par :

$$k_1 = h \cdot f(x_i, y_i)$$

$$k_2 = h \cdot f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1)$$

$$k_3 = h \cdot f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2)$$

$$k_4 = h \cdot f(x_i + h, y_i + k_3)$$

Dès lors, pour $x_1 = x_0 + h$:

$$y_1 = y_0 + (k_1 + 2k_2 + 2k_3 + k_4)/6$$

Lorsqu'un y_1 est évalué il est ensuite possible de répéter ce procédé pour une nouvelle valeur $x_2 = x_1 + h$ et obtenir y_2 et ainsi de suite...

De ce fait, vous pouvez calculer $y = f(x)$ pour un grand nombre de points, le pas d'intégration h pouvant être pris aussi petit que l'on veut.

Le calcul de la fonction $y = f(x)$ est d'autant plus précis que h est petit, mais le temps de calcul s'accroît en proportion.

A titre de vérification, précisons que la solution analytique de l'équation différentielle

$$y' = \exp(x) - y$$

$$\text{est : } y = (\exp(x) - \exp(-x))/2$$

soit sinus hyperbolique de x : $Sh(x)$ qui permet pour $x_1 = 0,5$ et $x_2 = 1$ de retrouver $Y_1 = 0,521$ et $Y_2 = 1,175$.

Cette vérification nous donne l'indication que notre procédé de calcul se révèle exact.

L'organigramme de la **figure 1** montre l'algorithme employé. Après une phase d'initialisation, le programme exécute une boucle de calcul. A chaque passage dans cette boucle, un couple de valeur x, y ou $y = f(x)$ est calculé, ce qui implique, au préalable, la recherche de la valeur des coefficients k_1, k_2, k_3 et k_4 .

Le détail des opérations à effectuer sur l'exemple $y' = e^x - y$ permettra de mieux comprendre la méthode et facilitera la programmation en langage BASIC.

Posons les conditions initiales d'intégration :

$$X_0 = 0; Y_0 = 0; h = 0,5$$

$$k_1 = h \cdot f(X_0, Y_0) = 0,5 \times \{ \exp(0) - 0 \} = 0,5$$

$$k_2 = h \cdot f(X_0 + \frac{1}{2}h, Y_0 + \frac{1}{2}k_1) = 0,5 \times \{ \exp(0 + \frac{0,5}{2}) - (0 + \frac{0,5}{2}) \} = 0,517$$

$$k_3 = h \cdot f(X_0 + \frac{1}{2}h, Y_0 + \frac{1}{2}k_2) = 0,5 \times \{ \exp(0 + \frac{0,5}{2}) - (0 + \frac{0,517}{2}) \} = 0,513$$

$$k_4 = h \cdot f(X_0 + h, Y_0 + k_3) = 0,5 \times \{ \exp(0 + 0,5) - (0 + 0,513) \} = 0,568$$

Ainsi :

$$Y_1 = Y_0 + \frac{1}{6} \{ k_1 + 2k_2 + 2k_3 + k_4 \} = 0 + \frac{1}{6} \{ 0,5 + 2 \times 0,517 + 2 \times 0,513 + 0,568 \}$$

et

$$Y_1 = 0,521;$$

$$X_1 = 0,5;$$

$$h = 0,5.$$

En utilisant la valeur de Y_1 trouvée, on peut à nouveau calculer une seconde valeur Y_2 correspondant à $X_1 + h$. Cela revient au fond à changer les conditions initiales

$$Y_1 = 0,521;$$

$$X_1 = 0,5;$$

$$h = 0,5$$

$$k_1 = h \cdot f(X_1, Y_1) = 0,5 \times \{ \exp(0,5) - 0,521 \} = 0,564$$

$$k_2 = h \cdot f(X_1 + \frac{1}{2}h, Y_1 + \frac{1}{2}k_1) = 0,5 \times \{ \exp(0,5 + \frac{1}{2} \cdot 0,5) - (0,521 + \frac{1}{2} \cdot 0,564) \} = 0,657$$

$$k_3 = h \cdot f(X_1 + \frac{1}{2}h, Y_1 + \frac{1}{2}k_2) = 0,5 \times \{ \exp(0,5 + \frac{1}{2} \cdot 0,5) - (0,521 + \frac{1}{2} \cdot 0,657) \} = 0,634$$

$$k_4 = h \cdot f(X_1 + h, Y_1 + k_3) = 0,5 \times \{ \exp(0,5 + 0,5) - (0,521 + 0,634) \} = 0,782$$

$$Y_2 = Y_1 + \frac{1}{6} \{ k_1 + 2k_2 + 2k_3 + k_4 \} = 0,521 + \frac{1}{6} \{ 0,564 + 2 \times 0,657 + 2 \times 0,634 + 0,782 \}$$

$$Y_2 = 1,176$$

$$X_2 = 1$$

$$h = 0,5$$

... et ainsi de suite.

Dans le programme présenté **figure 2**, l'équation différentielle à intégrer est placée dans le sous-programme implémenté aux lignes 1000 et suivantes.

Fig. 2. - Listing du programme permettant la résolution d'équations différentielles de la forme $Y' = f(X, Y)$. L'équation proprement dite sera toujours placée en 1000: $y' = e^x - y$ dans cet exemple.

```

5  ' RESOLUTION D'EQUATIONS DIFFERENTIELLES
6  ' COPYRIGHT Y. TORRE ET MICRO-SYSTEMES
7  '
10 PRINT "PROGRAMME BASIC POUR RESOUDRE DES EQUATIONS"
20 PRINT "DIFFERENTIELLES DE LA FORME Y'=F(X,Y)"
30 INPUT "CONDITIONS INITIALES D'INTEGRATION X0,Y0 :";X0,Y0
40 INPUT "INCREMENT DE LA VARIABLE X -> H :";H
45 PRINT
50 PRINT " *** X", " *** Y=F(X) ":PRINT X0,Y0
60 FOR N=0 TO 11:REM NOMBRE DE COUPLES
70 X=X0:Y=Y0:GOSUB 1000
80 K1=H*Y1
90 X=X0+H/2:Y=Y0+K1/2:GOSUB 1000
100 K2=H*Y1
110 Y=Y0+K2/2:GOSUB 1000
120 K3=H*Y1
130 X=X0+H:Y=Y0+K3:GOSUB 1000
140 K4=H*Y1
150 YX=Y0+(K1+2*K2+2*K3+K4)/6
160 PRINT X, YX
170 X0=X:Y0=YX
180 NEXT
190 END
1000 Y1=EXP(X)-Y
1010 RETURN
    
```

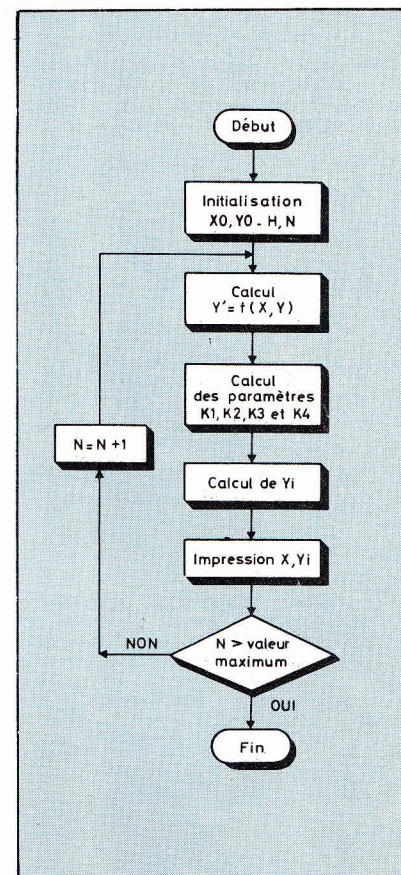


Fig. 1. - Organigramme de la résolution d'équations différentielles par la méthode de Runge-Kutta.


```
PROGRAMME BASIC POUR RESOUDRE DES EQUATIONS
DIFFERENTIELLES DE LA FORME Y'=F(X,Y)
CONDITIONS INITIALES D'INTEGRATION X0,Y0 : 0 , 0
INCREMENT DE LA VARIABLE X -> H : .5
```

```
*** X      *** Y=F(X)
0          0
.5        .521254
1         1.17568
1.5       2.13029
2         3.6287
2.5       6.05336
3         10.0232
3.5       16.5514
4         27.3045
4.5       45.0271
5         74.2429
5.5       122.409
6         201.821
```

Fig. 3. - L'exécution de l'intégration de la fonction : $y' = \exp(x) - y$.

```
PROGRAMME BASIC POUR RESOUDRE DES EQUATIONS
DIFFERENTIELLES DE LA FORME Y'=F(X,Y)
CONDITIONS INITIALES D'INTEGRATION X0,Y0 : 0 , 1
INCREMENT DE LA VARIABLE X -> H : .02
```

```
*** X      *** Y=F(X)
0          1
.02       1.02102
.04       1.04413
.05       1.06946
.08       1.09713
.1        1.12726
.12       1.16001
.14       1.19554
.16       1.23405
.18       1.27573
.2        1.32083
.22       1.36961
.24       1.42237
```

Fig. 4. - L'exemple présenté ici donne les valeurs de la résolution de l'équation : $y' = 3x + y^2$.

```
PROGRAMME BASIC POUR RESOUDRE DES EQUATIONS
DIFFERENTIELLES DE LA FORME Y'=F(X,Y)
CONDITIONS INITIALES D'INTEGRATION X0,Y0 : .5 , 0
INCREMENT DE LA VARIABLE X -> H : .1
```

```
*** X      *** Y=F(X)
.5         0
.6        -.125795
.7        -.250869
.8        -.369849
.9        -.480742
1         -.583959
1.1       -.681367
1.2       -.775464
1.3       -.868771
1.4       -.96346
1.5       -1.06115
1.6       -1.16287
1.7       -1.26904
```

Fig. 5. - L'intégration de l'équation différentielle : $y' = y^2 - x^2 + \sin[\log(x^2)] - \frac{y}{x} \text{Arc tg}(y^2 - x^2)$

Trois exemples...

Nous vous présentons trois exemples d'équations différentielles (toujours placées en 1000) à résoudre.

L'exemple 1 traite l'équation que nous venons d'étudier :

$$y' = e^x - y$$

et les résultats sont présentés figure 3.

Le deuxième exemple concerne la fonction :

$$y' = 3x + y^2$$

qui s'implémente ainsi dans le programme :

```
1000 Y1=3*X+Y*Y
1010 RETURN
```

et dont les résultats sont donnés figure 4.

L'équation différentielle du troisième exemple est nettement plus compliquée et sa solution analytique n'est pas, c'est le moins que l'on puisse dire, évidente :

$$y' = y^2 - x^2 + \sin[\log(x^2)] - \frac{y}{x} \text{Arc tg}(y^2 - x^2)$$

Et pourtant la solution numérique peut être trouvée en quelques secondes. Il suffit pour cela d'introduire dans le programme les lignes suivantes :

```
1000 Y1=Y*Y-X*X+SIN(LOG
(X*X))-Y/X*ATN(Y*Y-X*X)
1010 RETURN
```

Les résultats de ce calcul sont présentés figure 5. Il faut cependant éviter ici de poser $x_0 = 0$.

La précision des résultats dépend surtout du choix de l'incrément de la variable x. Aussi faut-il prendre soin de vérifier leurs exactitudes pour des valeurs différentes de h.

Dans notre prochain numéro nous vous présenterons des résolutions d'équations différentielles encore plus compliquées (dont les

équations différentielles du 2^e ordre) et nous développerons le cas pratique du mouvement réel du pendule. ■

Y. TORRE

Les méthodes de Runge-Kutta

Les méthodes de Runge-Kutta sont des méthodes largement utilisées en analyse numérique dans le cadre des résolutions (ou intégration) d'équations différentielles.

Ces méthodes sont fondées sur le développement d'une fonction en série de Taylor au moyen de combinaisons des dérivées de cette fonction. En ce sens, elles sont donc limitées, c'est-à-dire que si la fonction intégrée n'est pas développable en série de Taylor ou si l'intervalle de calcul contient une discontinuité, le point de discontinuité doit être déterminé et la solution calculée jusqu'à ce point puis repris à partir de ce point.

Il existe ainsi plusieurs méthodes de Runge-Kutta qui se distinguent par leur ordre de développement.

Au premier ordre la formule d'intégration devient :

$$y_{i+1} = y_i + h \cdot f(x_i, y_i)$$

Cette formule n'est pas intéressante car trop peu précise.

Au deuxième ordre nous obtenons :

$$y_{i+1} = y_i + \frac{1}{2}(k_1 + k_2)$$

avec $k_1 = h \cdot f(x_i, y_i)$

$$k_2 = h \cdot f(x_i + h, y_i + k_1)$$

Et ainsi de suite jusqu'à un ordre de degré plus élevé. La formule d'ordre 4 que nous développons dans cet article est la plus commune et la plus employée. ■

La micro-informatique : Nous maîtrisons son avenir.

Pourquoi la micro informatique?

La constante évolution des micro-processeurs a permis de mettre au point du matériel et une méthode à la portée de tous. Le micro-ordinateur sans avoir la complexité des gros ordinateurs a la faculté de résoudre une multitude de problèmes de comptabilité et de gestion courante. Diminuer les coûts, accroître la productivité des entreprises tel est le pari de la micro-informatique.

Les champs d'application.

Artisans, commerçants, petites entreprises, centres de recherches et universités n'ont ni l'utilité ni les moyens de financer et de rentabiliser une grosse unité informatique. Ils ont cependant besoin d'avoir un outil capable d'optimiser leurs résultats et de les aider à choisir judicieusement leurs investissements par secteurs de l'entreprise.

Le groupe «Sud-Est Informatique»

La micro-informatique est aujourd'hui fiable, efficace, économique. Elle devient l'indispensable solution d'avenir de la petite entreprise. Seul un groupe parfaitement homogène représentant une force d'achat, des programmes éprouvés, un service après-vente efficace et dynamique, peut répondre à un besoin spécifique de l'entreprise régionale. C'est la raison d'être du groupe **Sud-Est Informatique** auquel les plus grands fabricants américains font confiance pour distribuer leur matériel et en assurer la maintenance...



Apple II

Le leader des micro-ordinateurs. Dans le monde de la micro-informatique, la force d'Apple, c'est sa capacité à s'adapter et à innover.



Altos - TKL 8000

Système de gestion évolutif mono ou multi-postes. Large gamme de disques et de logiciels. L'équipement hautement professionnel.



Commodore

Pour la gestion des P.M.E., la bureautique, l'instrumentation, les applications industrielles, l'enseignement, les bureaux d'études.

10 spécialistes à votre service.

Les 10 spécialistes du groupe **Sud-Est Informatique** sont à votre disposition pour vous présenter un outil de travail modulable, adapté à votre problème spécifique. N'hésitez pas à les consulter et à leur demander une démonstration sur place ou à domicile sans engagement de votre part.

Votre spécialiste conseil

L'Onde Maritime Informatique

conseil - boutique - S.A.V
28, Bd du Midi - Tél. (93) 47.44.30
06150 Cannes la Bocca

Société Duclot

av. de la Libération - Lupino
20200 Bastia - Tél. (95) 31.68.12

A.C.T Informatique

analyse - conseil - télétraitement
33, rue de Paris - 06000 Nice
Tél. (93) 80.06.62

Biosignal

études - recherches biomédicales
114, av. de la Californie - 06200 Nice
Tél. (93) 86.50.67

Microtek

conseil - boutique informatique
2, Bd Rainier III - Monaco
Tél. (93) 50.60.98

Mécanographie Dracenoise

27, Bd de la Liberté - 83300 Draguignan
Tél. (94) 67.15.56

S.C.M.S

formation - boutique informatique
115, rue Rémy Belleau - 83700 St Raphaël
Tél. (94) 95.47.15

Somevar

77, rue Léon Isnard - 83700 St Raphaël
Tél. (94) 95.73.03

Technic-Bureau

29, av. Gambetta - 83000 La Seyne / mer
Tél. (94) 94.64.73

Nous maîtrisons son avenir. Maîtrisez le vôtre.

Pour plus de précision cerchez la référence 141 du « Service Lecteurs »

L'analyse budgétaire annuelle

Prévision et contrôle des dépenses sont des facteurs fondamentaux de la gestion d'une entreprise ou d'un portefeuille familial.

Il n'est pas de société ou de foyer qui ne puisse se passer d'une prévision et d'une analyse de ses dépenses, effectuant ainsi un contrôle précis des budgets.

Nous vous présentons aujourd'hui un programme réalisant les fonctions fondamentales de l'analyse budgétaire. Tout comme le programme Basic sur le « calcul de l'emprunt » publié dans notre précédent numéro, ce programme est modulaire et vous fournira ainsi un canevas pour développer des outils de gestion adaptés à vos besoins. La longueur de ce programme est d'environ 300 octets.

LISTING N. 1

```
10 T$=" "
20 S$="APPUYEZ SUR 'SHIFT' POUR CONTINUER"
30 PRINT CHR$(147)
40 DIM M$(11):DIM B$(11):DIM A$(11)
50 DIM D$(11):DIM P$(11)

60 PRINT "ANALYSE BUDGETAIRE ANNUELLE"
70 PRINT:PRINT T$:PRINT
80 PRINTTAB(5)"CE PROGRAMME VOUS PERMET"
90 PRINT
100 PRINTTAB(5)"DE PROCEDER A UNE ANALYSE"
110 PRINT
120 PRINTTAB(5)"BUDGETAIRE ANNUELLE"
130 PRINT
140 PRINTTAB(5)"A PARTIR DES DONNEES ENTREES"
150 PRINT
160 PRINTTAB(5)"EN DEBUT D'ANNEE"
170 PRINT" POUR LE BUDGET":PRINT
180 PRINTTAB(5)"ET MENSUELLEMENT"
190 PRINT" OU EN FIN D'ANNEE":PRINT
200 PRINTTAB(5)"POUR LES DEPENSES REELLES"
210 PRINT
220 PRINT
230 PRINT T$:PRINT T$:PRINT:PRINT S$
240 WAIT 516.1

250 PRINT CHR$(147)
260 PRINT:PRINT"FRAPPEZ LES LETTRES INDIQUEES"
270 PRINT
280 PRINT" ENTRE PARENTHESES":PRINT
290 PRINT" POUR OBTENIR VOTRE INFORMATION"
300 PRINT
310 PRINT T$:PRINT
320 PRINT"BUDGET DE L'ANNEE (BA)"
330 PRINT
340 PRINT"PREVISIONS DE DEPENSES EN COURS(PD)"
350 PRINT
360 PRINT"DEPENSES REELLES A CE JOUR (DR)"
370 PRINT
380 PRINT"DEPASSEMENTS A SIGNALER (DS)"
390 PRINT
400 PRINT"TOTAUX CUMULES A CE JOUR (TC)"
410 PRINT
420 PRINT"DONNEES DEJA ENTREES (DA)"
430 PRINT
440 PRINT:PRINT T$:PRINT
450 INPUT"BA * PD * DR * DS * TC OU DA":J$
460 GOTO 560
470 PRINT T$
480 PRINT"BUDGET DE L'ANNEE (BA)"
490 PRINT"PREVISIONS DE DEPENSES EN COURS(PD)"
500 PRINT"DEPENSES REELLES A CE JOUR (DR)"
510 PRINT"TOTAUX CUMULES A CE JOUR (TC)"
520 PRINT"DEPASSEMENTS A SIGNALER (DS)"
530 PRINT"DONNEES DEJA ENTREES (DA)"
540 INPUT"BA * PD * DR * TC * DS OU DA":J$
550 IF J$="PD" THEN 630
570 IF J$="BA" THEN 770
580 IF J$="DS" THEN 830
590 IF J$="DR" THEN 930
600 IF J$="DA" THEN 1000
610 IF J$="TC" THEN 1050
620 GOTO 250
```

L'analyse budgétaire consiste principalement à établir des prévisions de recettes ou de dépenses, puis à comparer les résultats obtenus avec ces prévisions afin d'en dégager des écarts caractéristiques.

Les différences constatées seront autant d'indicateurs de la bonne ou de la mauvaise marche financière de l'entreprise ou du foyer. Ainsi les écarts entre le **prévu** (le budgeté) et le **réalisé** peuvent entraîner des décisions importantes concernant votre gestion.

Le programme présenté ici permet de procéder à une analyse budgétaire annuelle à partir des données du budget entrées en début d'année et des dépenses réelles effectuées mois après mois. Celui-ci est volontairement simple, il montre la facilité avec laquelle un programme de gestion doit être utilisé. En effet il ne faut jamais perdre de vue que l'informatique de gestion est avant tout un outil et qu'elle est destinée à faciliter et non à compliquer le travail d'un gestionnaire, qu'il s'agisse d'un dirigeant d'entreprise ou d'un particulier cherchant à mieux gérer ses rentrées.

La technique de programmation employée ici est la **programmation modulaire** contrôlée par menu dont le schéma synoptique est donné **figure 1**. Dans la programmation modulaire, chaque module est indépendant de son voisin, et l'utilisateur choisit le module de calcul qu'il désire, par l'entremise d'un « menu » précisant les différentes options disponibles.

Le programme

Le programme complet est divisé en deux grandes parties, cha-

cune d'elle contenant plusieurs sous-ensembles (**listing 1 et 2**).

Ainsi, la première partie (**lignes 10-620**) est organisée de la façon suivante :

Lignes 10-50 : Routine d'initialisation des variables.

Lignes 60-240 : Présentation du programme : générique.

Lignes 250-460 : Affichage du menu complet.

Lignes 470-620 : Affichage du menu succinct pour les rappels.

La deuxième partie (**lignes 630-2120**) comprend 6 options de calcul réalisant chacune une tâche particulière (dans l'ordre d'apparition dans le programme) :

Lignes 630-760 : Option 2 – Prévision des dépenses en cours (PD).

Lignes 770-820 : Option 1 – Budget annuel (BA).

Lignes 830-920 : Option 5 – Dépassement du budget (DS).

Lignes 930-990 : Option 3 – Dépenses réelles (DR).

Lignes 1000-1040 : Option 6 – Données entrées (DA).

Lignes 1050-1100 : Option 4 – Totaux cumulés (TC).

Lignes 2000-2120 : Données du programme (DATA).

Examinons en détail ce que représente chacun de ces modules :

● Première partie

Initialisation du programme

Les lignes 10-50 effectuent l'initialisation du programme en effaçant l'écran par l'instruction CHR\$(147) (ligne 30), en dimensionnant les variables pour les 12 mois de l'année, (lignes 40-50) et en donnant des valeurs initiales à 2 variables alpha-numériques T\$ et S\$ (lignes 10-20).

Générique et menu

Les lignes 60-240 réalisent l'affichage du « générique » du programme. Celui-ci est constitué d'un simple texte exprimant l'objet du programme.

Listing 1. – Cette première partie du programme contient les routines d'initialisation et les « menus » permettant de choisir les différentes options de calcul.

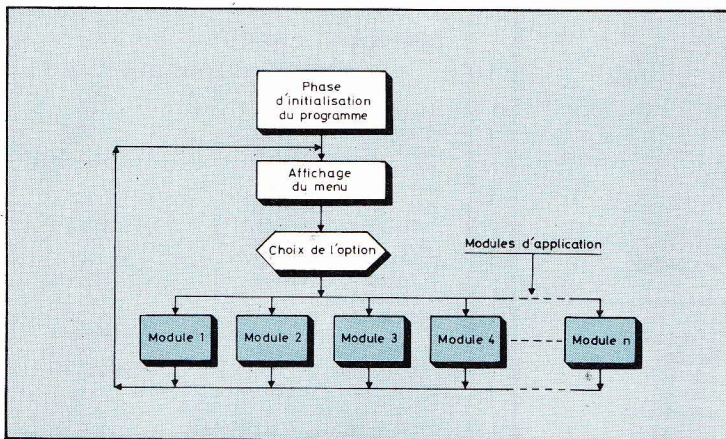


Fig. 1. - Schéma synoptique d'un programme modulaire contrôlé par menu. L'utilisateur d'un tel programme peut choisir à chaque moment le module de traitement qu'il désire. Chaque module est totalement indépendant des autres, ce qui permet d'ajouter ou de retirer un module facilement sans avoir à modifier l'ensemble du programme. De plus, dans ce type de programmation, un module peut être lui-même un programme modulaire contrôlé par un autre menu.

L'instruction WAIT 516,1 de la ligne 240 sert à marquer un temps d'arrêt dans l'exécution du programme.

Pour le relancer, il suffira d'appuyer sur la touche SHIFT. Les lignes 250-620 constituent deux modules de « menus ». Un menu est un programme ou une routine présentant à l'utilisateur les options mises à sa disposition.

Lorsque celui-ci a fait son choix, une instruction de branchement permet d'exécuter le module de programme désiré. D'une part, le menu complet (lignes 250-460) visualise en toutes lettres les différentes options et, d'autre part, les lignes 470-620 constituent un menu succinct utilisé, à titre de rappel, pour visualiser les diverses options du programme après le traitement de chaque module. L'affichage du menu succinct est systématique après l'édition de chaque tableau.

Le menu général tel qu'il apparaît sur l'écran, après exécution de ce module, est présenté figure 2.

● Deuxième partie

Option 1

Budget Annuel (BA)

Ce module (lignes 770-820) affiche sur l'écran les prévisions budgétaires à partir des données annuelles introduites directement dans le programme pour l'ensemble de l'année. A la fin de cette option, comme à la fin des 6

autres options que nous allons examiner, le programme effectue un branchement inconditionnel à la ligne 470 pour le rappel du menu succinct.

Option 2

Prévision des dépenses en cours (PD)

Cette option (lignes 630-760) prend en compte les prévisions de dépenses en cours. En effet, il arrive souvent qu'au cours de l'année les prévisions des dépenses doivent être réajustées. Pour ce faire il suffit d'indiquer le taux de majoration ou de minoration (+ ou - un pourcentage) auquel les prévisions budgétaires sont affectées (lignes 640-690). L'affichage des prévisions budgétaires majorées ou minorées suivant la valeur désirée est effectuée aux lignes 700-750.

Un exemple d'application de cette option est donné figure 3.

Option 3

Dépenses réelles (DR)

Ce module (lignes 930-990) affiche les dépenses réelles réalisées

Listing 2. - Ce programme comprend l'ensemble des options de traitement disponibles à l'utilisateur : budget de l'année, prévisions de dépenses, dépenses réelles, totaux cumulés, dépassements à signaler et l'affichage des données déjà introduites dans le programme.

LISTING N. 2

```

630 PRINTCHR$(147)
640 PRINT:PRINT:PRINT"ENTREZ LE POURCENTAGE"
650 PRINT
660 PRINT"DE MAJORATION":PRINT
670 PRINT"OU DE MINORATION":PRINT
680 PRINT"PRECEDE DU SIGNE + OU -":PRINT
690 PRINT:INPUT X:PRINT
700 PRINT CHR$(147)
710 PRINT"PREVISIONS AVEC":X;"% CONTRE"
720 PRINT:PRINT T$:PRINT
730 RESTORE
740 FOR M=0 TO 11:READ M$(M),B$(M),A$(M)
750 PRINT M$(M),B$(M)+B$(M)*X/100:NEXT M
760 GOTO 470
770 PRINT CHR$(147)
780 PRINT"BUDGET DE L'ANNEE":PRINT
790 PRINT T$:PRINT
800 RESTORE:FORM=0TO11:READM$(M),B$(M),A$(M)
810 PRINT M$(M),B$(M):NEXT M
820 GOTO 470
830 PRINT CHR$(147)
840 PRINT"DEPASSEMENT DU BUDGET"
850 PRINT T$:PRINT
860 PRINT"MOIS"," BUDGET"," DEPENSE"," DEPASSEM."
870 RESTORE
880 FOR M=0 TO 11:READ M$(M),B$(M),A$(M)
890 D$(M)=A$(M)-B$(M):P$(M)=A$(M)/B$(M)*100
900 IFSGN(D$(M))=1 THEN PRINTM$(M),B$(M),A$(M),D$(M)
910 NEXT M
920 GOTO 470
930 PRINT CHR$(147)
940 PRINT"DEPENSES A CE JOUR":PRINT
950 PRINT T$:PRINT
960 RESTORE
970 FOR M=0 TO 11:READ M$(M),B$(M),A$(M)
980 PRINT M$(M),A$(M):NEXT M
990 GOTO 470
1000 PRINT CHR$(147):TAB(10)"ENTREZ DANS L'ORDRE"
1010 PRINT T$:PRINT
1020 PRINT:PRINT TAB(10)" MOIS"," BUDGET"," DEPENSES"
1030 PRINT
1040 LIST 2000-
1050 PRINT CHR$(147)
1060 PRINT"TOTALS CUMULES A CE JOUR"
1070 PRINT T$:PRINT
1080 PRINT"MOIS"," BUDGET"," DEPENSE"," DIFFERENCE"
1090 T=0:T1=0:DP=0
1100 RESTORE
1110 FOR M=0 TO 11
1120 READ M$(M),B$(M),A$(M)
1130 T=T+B$(M)
1140 T1=T1+A$(M)
1150 IF T1=T1+A$(M) THEN T1=0
1160 DP=T-T1
1170 IF T1=0 THEN DP=0
1180 PRINT M$(M),T,T1,DP
1190 NEXT M
1200 GOTO 470
2000 DATA JANVIER , 750000 , 450000
2010 DATA FEVRIER , 800000 , 400000
2020 DATA MARS , 250000 , 500000
2030 DATA AVRIL , 500000 , 400000
2040 DATA MAI , 600000 , 750000
2050 DATA JUIN , 700000 , 300000
2060 DATA JUILLET , 950000 , 550000
2070 DATA AOUT , 650000 , 750000
2080 DATA SEPTEMBRE , 850000 , 600000
2090 DATA OCTOBRE , 250000 , 200000
2100 DATA NOVEMBRE , 500000 , -
2110 DATA DECEMBRE , 900000 , -
READY.
  
```



```

FRAPPEZ LES LETTRES INDIQUEES
ENTRE PARENTHESES
POUR OBTENIR VOTRE INFORMATION

-----
BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
DEPASSEMENTS A SIGNALER (DS)
TOTALS CUMULES A CE JOUR (TC)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * DS * TC OU DA ? BA
    
```

Fig. 2. - Affichage du menu permettant de choisir l'une des options de traitement. L'option BA sélectionnée ici est donnée à titre d'exemple.

Fig. 3. - L'option 2 visualise le tableau donnant les prévisions des dépenses en cours. Le taux de majoration ou de minoration auquel les prévisions budgétaires sont affectées est introduit au préalable à la demande du programme.

MOIS	BUDGET	DEPENSE	DIFFERENCE
JANVIER	750000	450000	300000
FEBVRIER	1550000	850000	700000
MARS	1800000	1350000	450000
AVRIL	2300000	1750000	550000
MAI	2900000	2500000	400000
JUIN	3600000	2800000	800000
JUILLET	4550000	3350000	1200000
AOUT	5200000	4100000	1100000
SEPTEMBRE	6050000	4700000	1350000
OCTOBRE	6300000	4900000	1400000
NOVEMBRE	6800000	0	0
DECEMBRE	7700000	0	0

```

BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
TOTALS CUMULES A CE JOUR (TC)
DEPASSEMENTS A SIGNALER (DS)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * TC * DS OU DA? DA
    
```

Fig. 5. - Présentation des valeurs budgétées et réalisées ainsi que leurs différences sous forme cumulée.

jusqu'à la dernière entrée effectuée dans le programme par l'utilisation des instructions DATA (lignes 2000-2120).

Un exemple d'application de ce module est présenté figure 4. Aucune dépense n'ayant été enregistrée pour les mois de novembre et décembre, le programme affiche « zéro » pour ces 2 mois.

Option 4

Totaux cumulés (TC)

L'option 4 (lignes 1050-1100) présente les valeurs budgétées, les

valeurs réalisées ainsi que les écarts sous leur forme cumulée. Le tableau correspondant à ces résultats est donné figure 5.

Option 5

Dépassement du budget (DP)

Ce module (lignes 830-920) visualise les mois et les valeurs pour lesquels le budget a été dépassé.

Pour le gestionnaire ces mois constituent des indicateurs d'anomalies particulières qui n'avaient pas été prévues lors de l'élaboration du budget. Un exemple de dépassement est donné figure 6.

```

ENTREZ LE POURCENTAGE
DE MAJORATION
OU DE MINORATION
PRECEDE DU SIGNE + OU -
? +10
PREVISIONS AVEC 10 % CONTRE
-----
JANVIER 825000 750000
FEVRIER 880000 800000
MARS 275000 250000
AVRIL 550000 500000
MAI 660000 600000
JUIN 770000 700000
JUILLET 1045000 950000
AOUT 715000 650000
SEPTEMBRE 935000 850000
OCTOBRE 275000 250000
NOVEMBRE 550000 500000
DECEMBRE 990000 900000
-----
BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
TOTALS CUMULES A CE JOUR (TC)
DEPASSEMENTS A SIGNALER (DS)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * TC * DS OU DA? DR
    
```

MOIS	BUDGET	DEPENSE	DEPASSEM.
MARS	250000	500000	250000
MAI	500000	750000	250000
AOUT	650000	750000	100000

```

BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
TOTALS CUMULES A CE JOUR (TC)
DEPASSEMENTS A SIGNALER (DS)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * TC * DS OU DA? TC
    
```

Fig. 6. - Option 5. Affichage des mois et des valeurs pour lesquels le budget a été dépassé.

```

DEPENSES A CE JOUR
-----
JANVIER 450000
FEVRIER 400000
MARS 500000
AVRIL 400000
MAI 750000
JUIN 300000
JUILLET 550000
AOUT 750000
SEPTEMBRE 600000
OCTOBRE 200000
NOVEMBRE 0
DECEMBRE 0
-----
BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
TOTALS CUMULES A CE JOUR (TC)
DEPASSEMENTS A SIGNALER (DS)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * TC * DS OU DA? DS
    
```

Fig. 4. - Option 3. Les dépenses réelles effectuées jusqu'à ce jour (jusqu'à la dernière entrée effectuée dans les DATA). Aucune dépense n'ayant été enregistrée pour novembre et décembre, le programme affiche des zéros pour ces deux mois.

MOIS	BUDGET	DEPENSES
2000 DATA JANVIER	750000	450000
2010 DATA FEBVRIER	800000	400000
2020 DATA MARS	250000	500000
2040 DATA AVRIL	500000	400000
2050 DATA MAI	600000	750000
2060 DATA JUIN	700000	300000
2070 DATA JUILLET	950000	550000
2080 DATA AOUT	650000	750000
2090 DATA SEPTEMBRE	850000	600000
2100 DATA OCTOBRE	250000	200000
2110 DATA NOVEMBRE	500000	-
2120 DATA DECEMBRE	900000	-

```

BUDGET DE L'ANNEE (BA)
PREVISIONS DE DEPENSES EN COURS(PD)
DEPENSES REELLES A CE JOUR (DR)
TOTALS CUMULES A CE JOUR (TC)
DEPASSEMENTS A SIGNALER (DS)
DONNEES DEJA ENTREES (DA)
-----
BA * PD * DR * TC * DS OU DA? DS
    
```

Fig. 7. - Option 6. Visualisation des données du programme déjà introduites. On peut supposer qu'en début d'année, les mois de l'année, ainsi que les montants des dépenses prévues dans le budget, sont entrés. Dans un deuxième temps, en demandant l'accès aux DATA (option DA), on indique les dépenses effectuées mois après mois.

Option 6

Données entrées (DA)

L'option 6 (lignes 1000-1040) affiche les données du programme (fig. 7). Celles-ci sont introduites sous la forme de DATA. En début d'année les montants prévus dans le budget ainsi que les mois auxquels se rapportent ces prévisions sont entrés dans le programme.

Dans un deuxième temps au cours de l'année, l'accès aux DATA peut être réalisé à tout moment, pour y placer les dépenses effectuées au cours des mois.

Les lignes 2000-2120 représen-

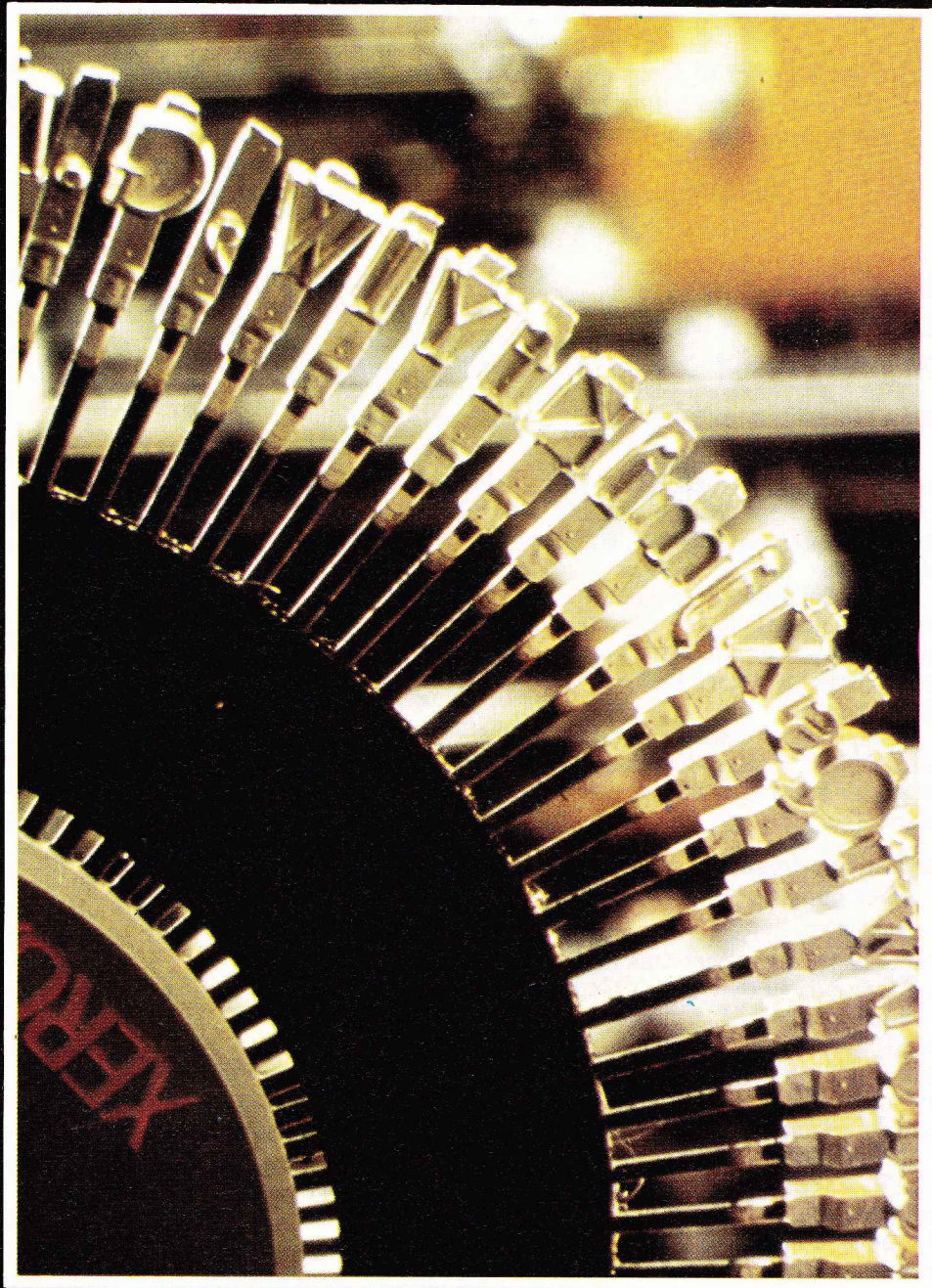
tent les DATA tels qu'elles ont été entrées dans le programme.

Comme dans le « calcul de l'emprunt » publié dans notre précédent numéro, les deux instructions spécifiques : WAIT 516,1 - ligne 240 - et CHR\$(146) - lignes 30, 250, 630, 700, 770, 830, 930, 1000 et 1050 ne sont pas prises en compte par tous les interpréteurs Basic. Vous devrez, le cas échéant, les remplacer par des instructions (ou ensemble d'instruction) équivalentes. ■

E. ADAMIS

Les imprimantes

Gros plan sur les caractères d'une tête d'impression type « marguerite ». (Doc. Xerox).



Bien qu'elles ne soient pas généralement l'élément qui suscite le plus vif intérêt pour l'utilisateur d'un système informatique, il est bon de ne pas les négliger car d'une part, elles génèrent des documents sanctionnant un travail qui va être jugé sur son contenu et sur sa forme, et d'autre part, en cas de non fonctionnement, elles immobilisent un système au moment psychologiquement crucial que constitue la sortie des résultats.

Le sujet est particulièrement vaste et il existe de nombreux types d'imprimantes qui, selon leur technologie et leurs fonctions possèdent un éventail étendu de possibilités.

Néanmoins, et si l'on considère la technologie comme

critère de sélection (critère que vous retrouverez tout au long de cet article), les méthodes d'impressions peuvent être classées en trois catégories :

- Les imprimantes à impact ou sans impact.
- L'impression caractère par caractère ou ligne par ligne.
- Les caractères préformés ou générés par une matrice de points.

Nous nous sommes attachés à décrire ici, ces principes d'impression en insistant plus particulièrement, toutefois, sur les imprimantes matricielles à aiguilles qui, aujourd'hui, sont celles correspondant le mieux au marché de la micro-informatique.

La vitesse d'impression s'exprime en nombre de caractères/seconde CPS pour les imprimantes caractères/caractère et en nombre de lignes/seconde pour les imprimantes lignes.

Les 3 choix technologiques...

Avant d'aborder la façon dont sont résolus, du point de vue de la technologie, les problèmes liés à l'impression d'un message sur un support (papier ou autre) et de déterminer, par voie de conséquence ce qui caractérise chaque type d'imprimantes, nous allons examiner ces 3 choix technologiques.

Toute imprimante utilise une combinaison de ces 3 méthodes d'impression.

1^{er} choix : imprimantes à impact ou sans impact

■ Les imprimantes à impact

Les imprimantes à impact sont basées sur une méthode bien connue de tous puisque c'est celle utilisée dans les machines à écrire classiques.

Comme dans les anciennes machines à écrire, chaque caractère est supporté par un marteau venant frapper le papier à travers un ruban encreur.

L'impression consiste donc en un transfert d'encre du ruban sur le papier grâce à un impact mécanique.

■ Les imprimantes sans impact

Dès 1952, plusieurs types d'imprimantes sans impact étaient développés aux Etats-Unis.

Ici, le procédé d'impression n'est plus mécanique mais basé sur un phénomène physique : électrostatique, électrolytique thermique ou lumineux (xérogaphie).

Nous analyserons plus loin, en détail, les caractéristiques des imprimantes utilisant ces différents systèmes d'impression, mais nous pouvons déjà les définir brièvement :

● **L'impression électrostatique :** Ce procédé consiste en l'impression directe par charge électrique d'un papier spécial et révélateur. Dans cette classe, nous trouvons aussi l'impression par jet d'encre. Ici, de minuscules gouttelettes sont chargées électriquement et dirigées à l'aide d'un champ électrique sur un papier ordinaire. L'impression électrostatique présente l'avantage d'être silencieuse et rapide.

● **L'impression électrolytique :** Le papier est imbibé d'une solution électrolytique. Des électrodes font passer un courant électrique à travers le papier, et les ions en provenance des électrodes métalliques forment l'image des caractères.

● **L'impression thermique :** L'impression se fait sur un papier spécial sensible à la chaleur, par échauffement local par stylets ou éléments chauffants.

● **Phénomènes lumineux :** Ce sont les imprimantes dites « XEROGRAPHIQUES ».

L'image est formée d'abord sur un support intermédiaire photorécepteur, préalablement chargé, ou par balayage d'un rayon laser sur un support photoconducteur.

Le transfert de l'image se fait à l'aide d'un colorant qui se dépose aux endroits où le support est chargé, par contact avec du papier ordinaire puis fixation.

En général, les imprimantes sans impact sont plus performantes, plus rapides, plus fiables et plus silencieuses que les imprimantes avec impact mécanique.

2^e choix : les modes d'impression caractère/caractère ou ligne/ligne

■ Le mode caractère par caractère

Ce sont les imprimantes dites « sérielles ». Les caractères sont imprimés les uns à la suite des autres, pour former une ligne, à une vitesse relativement faible.

Chaque caractère est positionné par déplacement du système d'impression ou du papier.

■ Le mode ligne par ligne

Ce sont les imprimantes dites « lignes » dans lesquelles l'impression de tous les caractères d'une ligne se fait en parallèle (simultanément). Le système d'impression couvre toutes les positions de caractères de la ligne, il y a donc autant de marteaux de frappe que de caractères dans une ligne.

3^e choix : Caractères préformés ou générés par points

■ Les caractères préformés

Le jeu de caractères est gravé une fois pour toute sur un support mobile. L'impression est réalisée lors du pressage des caractères sur le papier à travers un ruban encreur.

Les supports de caractères sont de formes très variées : cylindre, boule, marguerite, tulipe pour le mode d'impression caractère/caractère ou tambour, chaîne, bande... lorsque l'impression est du type parallèle, ligne par ligne.

■ Caractères générés par points

Chaque caractère est ici représenté par des points dessinés à l'intérieur d'une matrice définie généralement par 5 × 7 points ou 7 × 9 points. Cette méthode offre une grande souplesse d'emploi puisqu'elle présente l'avantage de pouvoir mettre à la disposition de l'utilisateur une gamme très étendue de symboles et de caractères.

C'est l'électronique de commande qui sélectionne les points correspondant à la configuration du signe choisi.

Dans les imprimantes à impact, celui-ci a lieu soit à « l'arrêt » lorsque le support papier est arrêté, soit au « vol ».

dans un même intervalle de temps (avec le décalage dû au temps de propagation), le même nombre d'intervalles élémentaires soit décompté aux deux extrémités de la liaison. Pour cela, chacune d'elle est équipée d'une horloge et l'extrémité émettrice envoie périodiquement un signal de synchronisation.

CCITT :

Comité Consultatif International pour le Télégraphe et le Téléphone : organisme officiel ayant décidé des normes internationales de transmissions téléphoniques.

OCR :

Optical character recogni-

tion : lecture optique de caractères.

DOT MATRIX :

Terme employé pour désigner la matrice à aiguilles d'une imprimante.

MTBF :

Mean Time Between Failure : définit le temps moyen entre deux pannes d'un matériel (il n'est significatif que statistiquement parlant et ne s'applique qu'aux matériels fonctionnant en deçà des spécifications du constructeur et du DUTYCYCLE).

DUTYCYCLE :

Cycle de travail. En ce qui

concerne une imprimante, son « dutycycle » exprimera (en pourcentage) le temps moyen d'utilisation répétée. Exemple : 50 % du Dutycycle pour 50 % de densité d'impression signifiera que l'imprimante pourra imprimer pendant quatre heures consécutives (50 % de 8 heures, c'est-à-dire une journée de travail) des lignes dont le nombre de caractères ne dépassera pas la moitié de la capacité totale (66 pour 132 col. et 40 pour 80 col.).

MTTR :

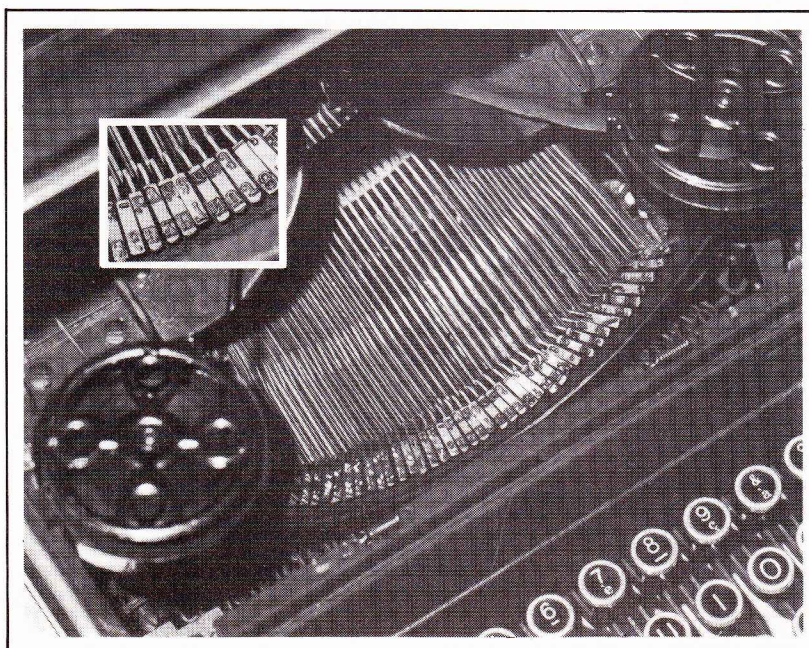
Mean Time To Repair : signifie temps moyen d'une réparation. ■

Les imprimantes à impact

L'impression à impact concerne un grand nombre d'imprimantes telles que les imprimantes à disque, tambour, chaîne, train, peigne, bande, matricielle à aiguilles...

Les imprimantes à impact permettent l'utilisation d'une très grande variété de papier et l'impression simultanée de copies par carbone.

L'impact a lieu soit à « l'arrêt » lorsque le support papier est arrêté, soit au « vol ».



Le « demi-cirque » des marteaux supportant les caractères dans les premières machines à écrire.

L'impression caractère/caractère

Les premières imprimantes :

Le marteau...

Les premières imprimantes utilisaient le principe des machines à écrire dans lequel l'ensemble des caractères (**le jeu**) était supporté par des **marteaux**, disposés en « **demi-cirque** », afin de frapper le papier au même endroit. Bien entendu, cette méthode d'impression était très lente (quelques caractères par seconde-CPS).

Le cylindre...

Mais, le début de l'ère informatique, lorsque l'on parle d'impression, commence avec les télétypes, tel que le « télétype 33 » qui utili-

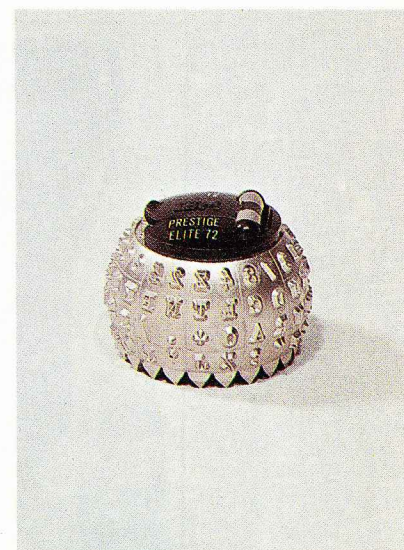
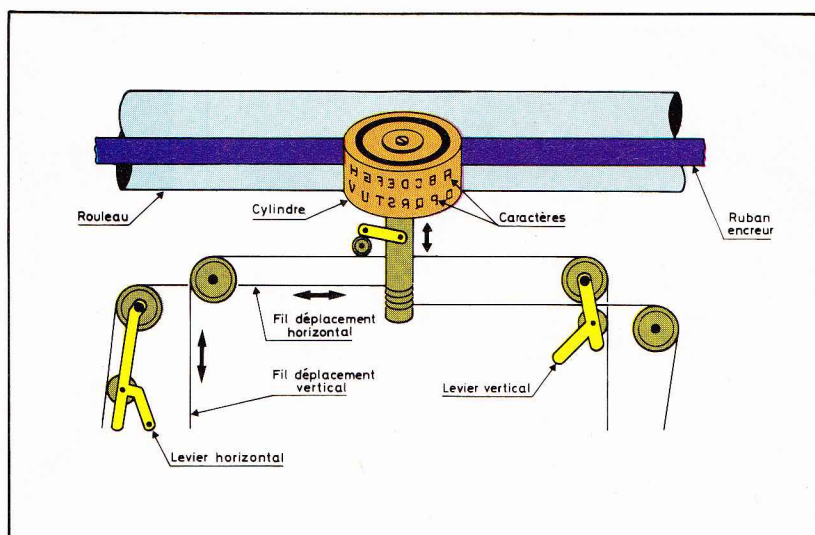


Fig. 1. – Le principe du cylindre – Le jeu de caractères est entièrement supporté par un cylindre proche du papier (1 mm). Le positionnement des caractères s'effectue par rotation autour de l'axe vertical et translation. L'impression a lieu en projetant le cylindre (ou le rouleau) sur le papier à travers le ruban encreur.

Une sphère d'impression IBM.

sait le principe du **cylindre** (fig. 1). Ici, le jeu de caractères est entièrement supporté par un cylindre, le positionnement de chaque caractère se faisant à partir d'une rotation autour de son axe vertical et d'une translation.

L'impact mécanique a lieu en projetant le cylindre sur le papier (à travers le ruban encreur) ou le rouleau porte-papier sur le cylindre. Placé plus près du papier que dans le cas précédent, le cylindre permet des vitesses d'impression plus grande : de 10 à 15 CPS

... et la boule

Vers le milieu des années 60, I.B.M. présente un système d'impression à boule (sélectric) très proche du cylindre dans son principe mais permettant le changement du jeu de caractères avec souplesse.

Les caractères sont disposés sous forme d'anneaux sur la surface de la sphère. Une double rotation horizontale/verticale de celle-ci permet de sélectionner le caractère voulu (fig. 2).

L'impression est réalisée par projection de la tête sur le papier après son positionnement latéral le long du chariot porte-tête.

Les vitesses ne dépassent guère 15 à 20 caractères par seconde.

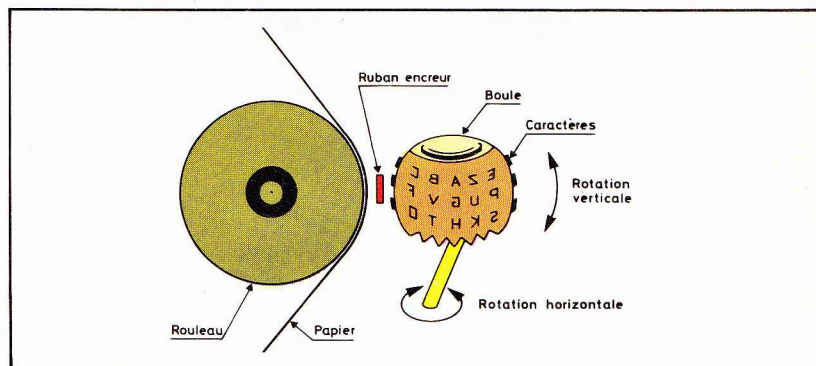
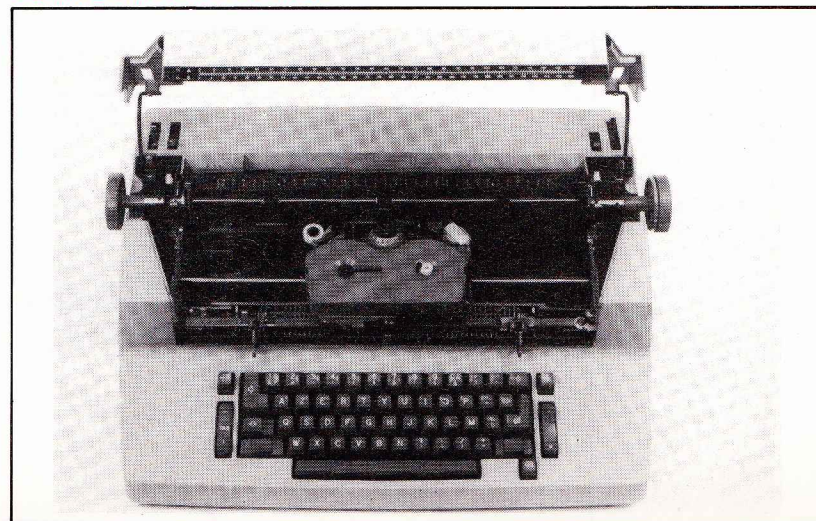
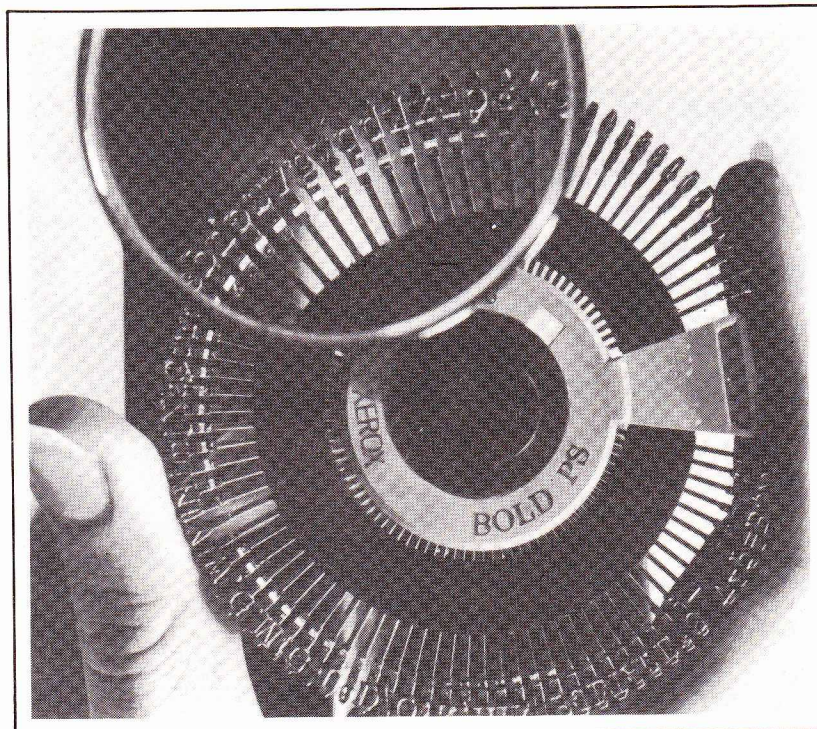


Fig. 2. – Le principe de la boule – Les caractères sont disposés sous forme d'anneaux sur la surface de la boule. Une double rotation horizontale/verticale de la boule permet de sélectionner le caractère voulu. L'impression est réalisée par projection de la sphère sur le papier.

Machine à écrire IBM 82C à boule.



Les imprimantes à caractère plein, de type « marguerite » ou « tulipe », sont très utilisées dans les applications de traitement de textes.



Une tête d'impression « marguerite » et, ci-dessus, gros plan sur le marteau de frappe. (Doc. Xerox).

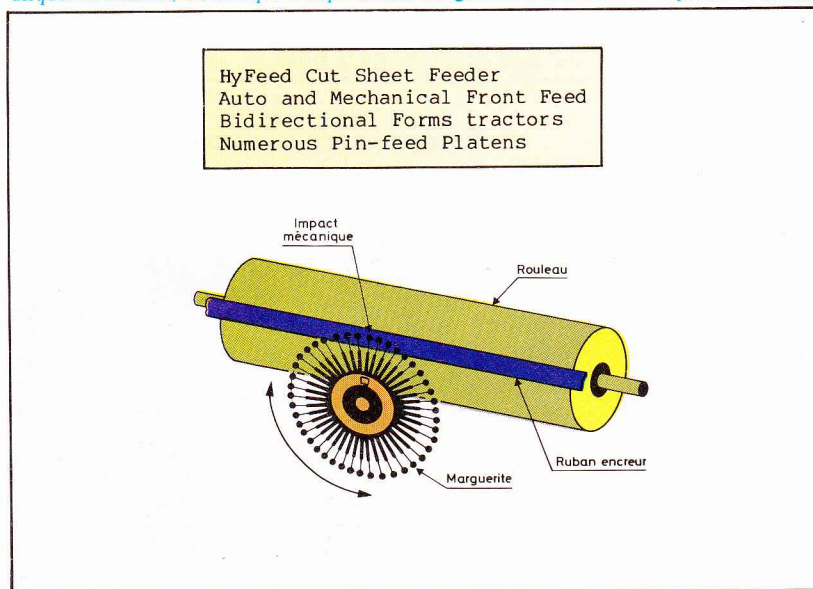
Les imprimantes à disque : de la Marguerite à la Tulipe

Au milieu des années 70, naît l'imprimante à caractère plein de type « marguerite » ou « tulipe ».

Elles sont très utilisées dans des applications de traitement de textes, du fait de leur grande qualité d'impression. Ces imprimantes sont relativement lentes (15 à 55 caractères par seconde) et onéreuses

pour le marché micro-informatique. L'arrivée d'imprimantes japonaises (Tec, Olympia) utilisant cette technologie laisse cependant présager d'une baisse importante des prix.

Fig. 3. – L'impression à marguerite – Une « marguerite » est une roue dont les rayons sont des bras souples qui supportent chacun un caractère à leur extrémité. Le positionnement des caractères s'effectue par rotation dans le plan vertical de la roue et déplacement latéral le long du rouleau. L'impression est obtenue par la frappe d'un marteau sur les caractères du disque. Ci-dessous, un exemple d'impression à marguerite réalisé sur une imprimante Diablo.



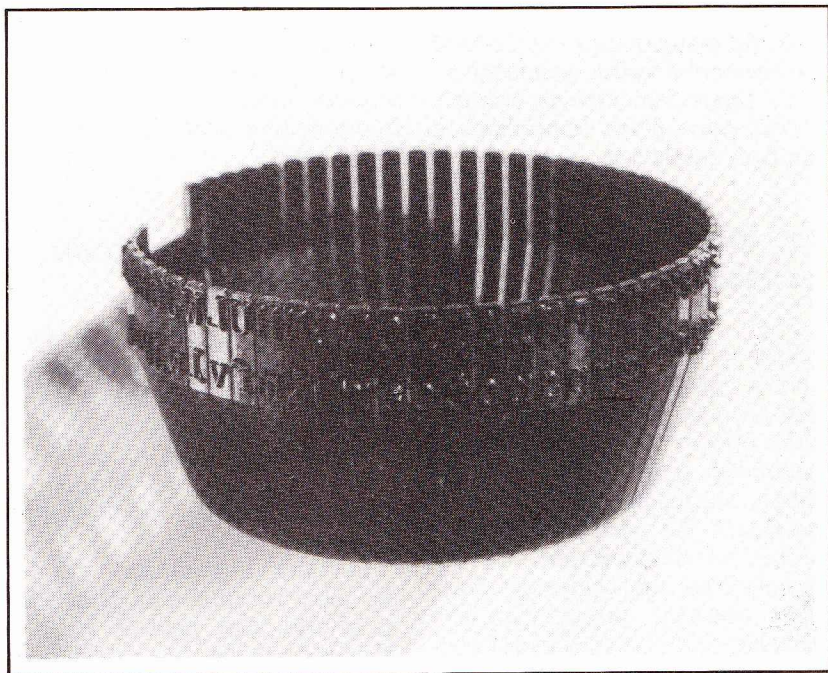
Le principe d'impression de la « Marguerite » (Daisy wheel), développé initialement par **Diablo**, repose sur la rotation dans un plan vertical d'une roue dont les rayons sont des bras souples qui supportent chacun un caractère à leur extrémité extérieure.

Le chariot sur lequel est montée la roue se déplace horizontalement le long d'un axe et l'impression est effectuée par la frappe d'un marteau sur les caractères du disque (fig. 3).

L'interchangeabilité du disque (plastique ou fer) assure une grande souplesse au niveau des jeux de caractères.

Des efforts sont faits pour augmenter la vitesse d'impression, en utilisant deux têtes par exemple (Qume twin track : 75 CPS).

Le principe de la « Tulipe » utilisé par NEC repose aussi sur la rotation d'un disque mais cette



Ci-dessus une « tulipe » French Pica 10... (Doc Yrel).



Ci-contre une imprimante à tulipe commercialisée par NEC : la Spin Writer avec son dispositif d'alimentation des documents. (Doc. Yrel).

fois-ci dans le plan horizontal, les caractères étant supportés par deux bras verticaux, l'impression est provoquée par la frappe d'un marteau comme pour la marguerite.

ce qui augmentait la vitesse d'impression.

A la ligne suivante, les caractères se déplaçaient en sens inverse et ainsi de suite pour les autres lignes.

L'inertie de la barre qu'il fallait déplacer dans un sens puis dans l'autre limitait la vitesse d'impression de ces imprimantes à 200 lignes par minute (LPM).

Les imprimantes lignes

Quelle que soit la méthode employée, le mode d'impression série, caractère par caractère, est nécessairement lent et ne peut guère dépasser 200 CPS ; 600 CPS pour les imprimantes matricielles à aiguilles.

Pour obtenir des vitesses d'impression encore plus élevées il faut imprimer une ligne entière d'un seul coup, tous les caractères composant la ligne étant sélectionnés pratiquement simultanément (en parallèle).

Les imprimantes à tambour

L'imprimante à barre oscillante

L'imprimante à barre oscillante fut l'une des premières imprimantes lignes.

Chaque caractère était monté sur un support solidaire d'une barre horizontale se déplaçant longitudinalement devant 132

marteaux. La barre faisait défiler devant chacun des marteaux tous les caractères du jeu. Il suffisait de commander électroniquement la frappe des marteaux au bon moment, lorsque le caractère sélectionné était présent au bon endroit. Une barre porteuse pouvait contenir 2 ou 3 jeux de caractères

Un système annonçant le tambour vit ensuite le jour. Il consistait à utiliser des roues portant chacune un jeu de caractères complet et dont la rotation autour d'un axe associée à la frappe des marteaux permettait l'impression des caractères choisis immobilisés sur une même ligne.

Cependant, si la technologie des imprimantes à tambour n'est plus de pointe de nos jours, il est bon, tout de même, de s'y attarder car

A 1 500 lignes/mn, le tambour tourne à 1 500 tr/mn. Pour un jeu de 80 caractères, le moment de l'impact doit être calculé à 1/1 000^e de seconde près.

le parc de machines installées est important (CONTROL DATA, DATA PRODUCTS...) et ce type d'imprimantes figure encore au catalogue de plusieurs constructeurs de systèmes.

Un cylindre métallique sur lequel est gravé un jeu complet de caractères pour chaque colonne d'impression est en rotation continue à vitesse constante.

Le papier et le ruban encreur se trouvent entre le tambour et le banc de marteaux. Chaque marteau, activé par un solénoïde, frappe à la volée (fig. 4) papier, ruban et caractère adéquat. L'impression d'une ligne correspond à une rotation complète du tambour. Compte-tenu des vitesses atteintes (de 300 à 1 500 caractères par minute) on peut imaginer combien l'électronique doit être sophistiquée pour pouvoir synchroniser frappe des marteaux et passage des caractères sélectionnés.

A 1 500 LPM, le tambour tourne à 1 500 tours/minute. Pour un jeu de 80 caractères le moment de l'impact doit être calculé à moins de 1 millième de seconde près.

Des imperfections apparues au niveau de l'alignement des caractères, le peu de souplesse dans le changement des jeux de caractères et la complexité des réglages

électromécaniques pour les techniciens de maintenance ont favorisé l'avènement d'autres technologies.

La figure 5 montre un exemple d'application d'une imprimante à tambour spécialisée.

Les imprimantes « chaîne »

Vers la fin des années 50, I.B.M. présenta son imprimante 1403 à chaîne (600 lignes par minute) connectée au système 1401.

Le principe est simple, puisqu'il ressemble pour toute la partie ruban, marteaux et solénoïdes au précédent. Le tambour qui tournait autour d'un axe horizontal est remplacé par une chaîne de caractères tournant autour d'un axe vertical.

Les caractères constituent les maillons de la chaîne et le jeu de caractères est répété plusieurs fois. La sélection des caractères par l'électronique repose sur le même principe que pour le tambour.

Là encore, le changement de jeu de caractères n'est pas aisé, mais le problème de l'ondulation verticale des caractères ne se pose plus, bien qu'il puisse parfois laisser place à un décalage horizontal des caractères par rapport aux marteaux.

Tous les caractères défilent devant les marteaux et l'impression se fait à la volée. Les plus grandes vitesses atteintes avec ce type d'imprimante sont de l'ordre de 2 000 LPM.

Les imprimantes train chaîne-train

I.B.M. continua de son côté avec ses imprimantes à chaîne de la famille 1403. Une légère modification de la « chaîne » amena l'imprimante « Train ». Cette stratégie fut judicieuse puisque le marché se déplaça vers la chaîne et le train.

Sur une imprimante « train », des blocs porteurs d'un groupe de caractères sont poussés le long d'un rail au lieu que chaque caractère soit tiré comme c'était le cas de la chaîne. La chaîne-train était un mélange de ces deux techniques. On peut citer à titre d'exemple l'imprimante train I.B.M. 3216 et la DATA PRINTER.

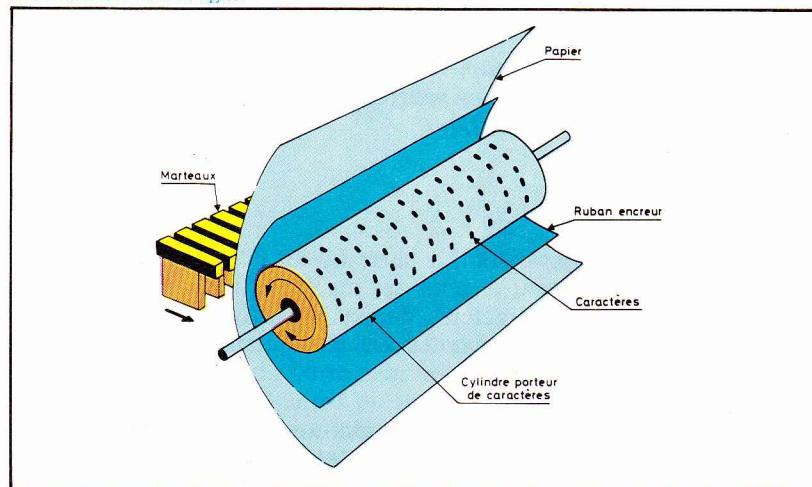
Les imprimantes à peigne

TALLY est à l'origine de ce type d'imprimante issu d'un heureux « mariage » entre l'impression par barre horizontale et l'impression matricielle. En effet, chaque dent du peigne (qui sert en même temps de marteau) porte une bille au lieu de porter un caractère plein. La première commande permet l'impression de tous les points constituant le haut des caractères. La deuxième commande, après déplacement vertical du papier imprime la deuxième rangée de points...

Ainsi, l'impression est provoquée par le **déplacement vertical du papier** et le **déplacement horizontal du peigne**.

Pour des caractères constitués d'une matrice de points 7 x 7 il faut donc réaliser 7 commandes conjointement à 7 sauts de papier pour imprimer complètement tous les caractères d'une ligne.

Fig. 4. - Le principe du tambour - Le tambour est un cylindre métallique sur lequel est gravé un jeu complet de caractères pour chaque colonne d'impression. A chaque tour du tambour, une ligne est imprimée complètement. Le papier et le ruban encreur sont placés entre le tambour et un banc de 132 marteaux pour l'impression simultanée des 132 caractères d'une ligne.



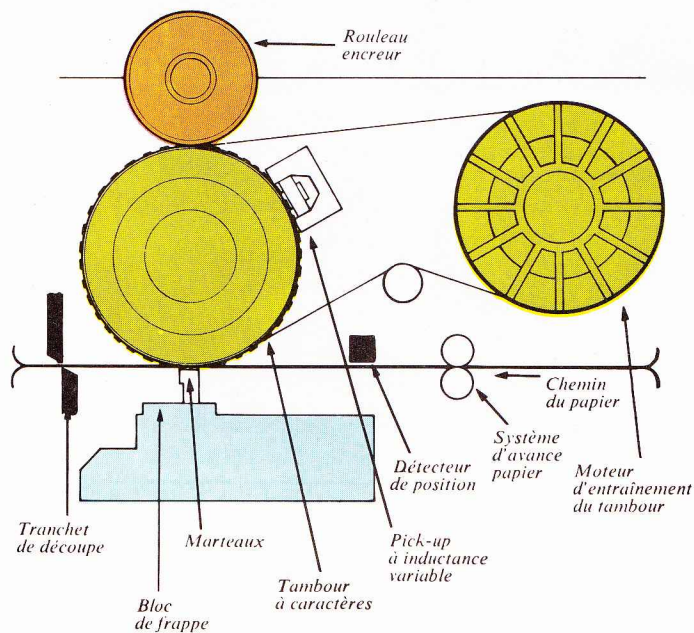
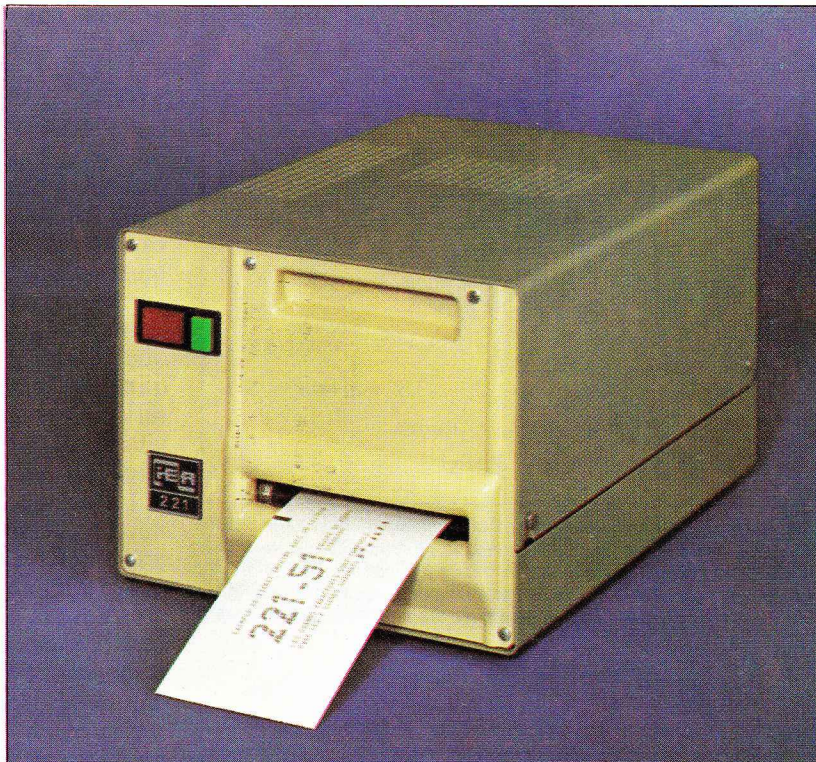
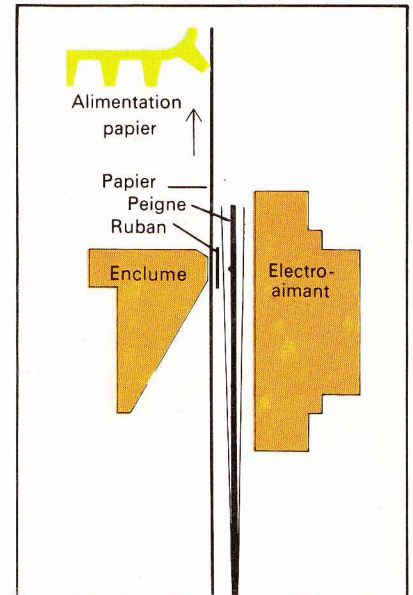
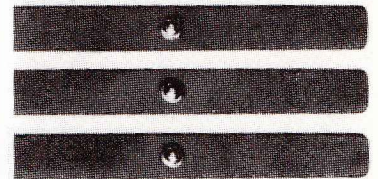


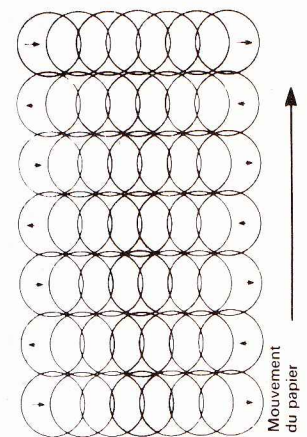
Fig. 5. - Une imprimante à tambour spécialisée - L'imprimante IER 221 spécialisée dans l'impression de tickets et de billets utilise le principe du tambour d'une manière originale. Le tambour est composé de 12 roues qui supportent chacune un jeu de caractères. A chaque roue est affecté un marteau qui frappe verticalement le caractère choisi. Un rouleau à grande réserve d'encre remplace le ruban encreur de l'imprimante à tambour traditionnelle.



Mécanisme d'impression simplifié.



Les marteaux...



Mouvement du marteau
Séquence d'impression, matrice 7 x 7.

Fig. 6. - Principe d'une imprimante à peigne - Un peigne est constitué de toute une ligne de marteaux d'impression comme ceux représentés ci-dessus au centre. Une première commande permet l'impression de tous les points constituant le haut des caractères. Le papier se déplace verticalement et une deuxième commande imprime la 2^e rangée de points des mêmes caractères. Il y a un marteau par caractère, le balayage complet de la matrice s'effectue par une oscillation horizontale du peigne.

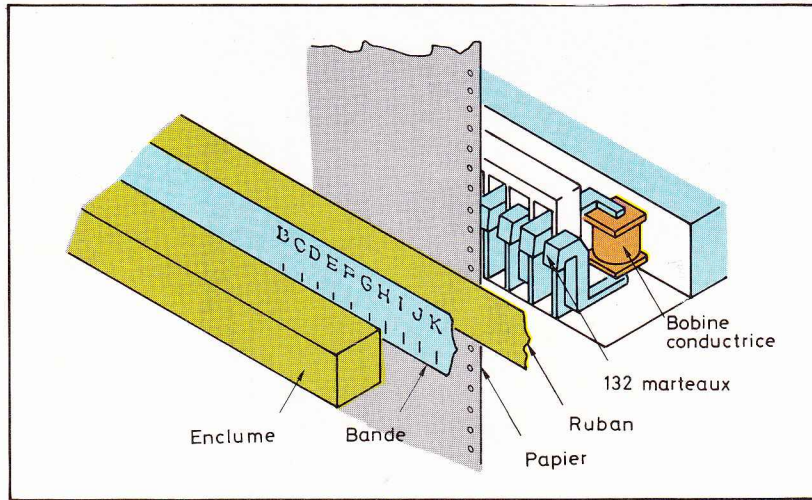
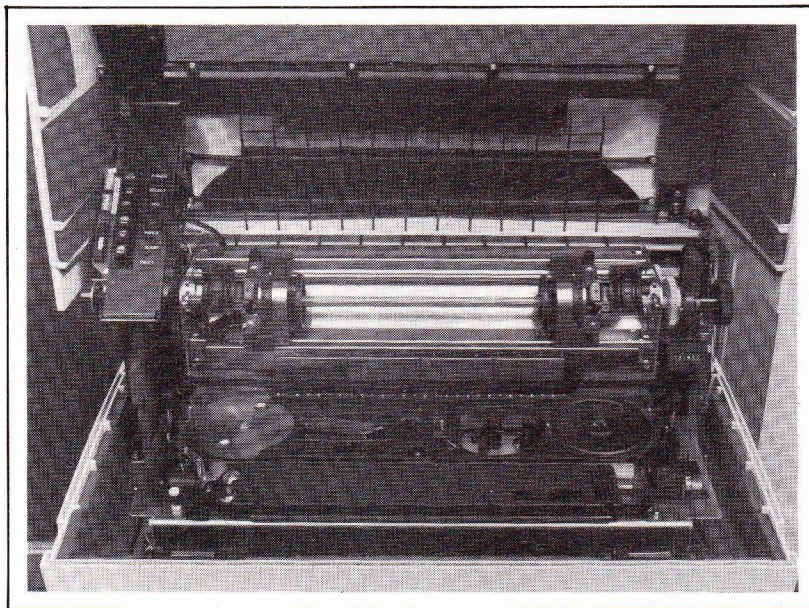


Fig. 7. - Principe de l'imprimante à bande - Une bande métallique sur laquelle apparaissent les caractères en relief tourne à vitesse constante. L'électronique de commande (à microprocesseur) détermine la position du caractère choisi sur la bande et l'endroit où il doit être imprimé sur le papier. Lorsque le caractère sélectionné passe devant la colonne (le marteau) bonne à imprimer, la coïncidence déclenchera la commande du marteau et le caractère sera imprimé à la volée.

Vue interne d'une imprimante à bande Centronics série 6000. (Doc. Centronics).



Le balayage complet de la matrice s'effectue par une oscillation horizontale du peigne, provoquée par un lien de flexion en acier scellé au moteur pas à pas qui oscille lui-même selon un axe de 15°.

Le **figure 6** montre le principe du peigne utilisé sur les imprimantes TALLY 2000 et 3000.

Les imprimantes à bandes

Depuis quelques années une variante de la chaîne : la bande, utilisée par la plupart des constructeurs, a gagné une place de choix sur ce marché.

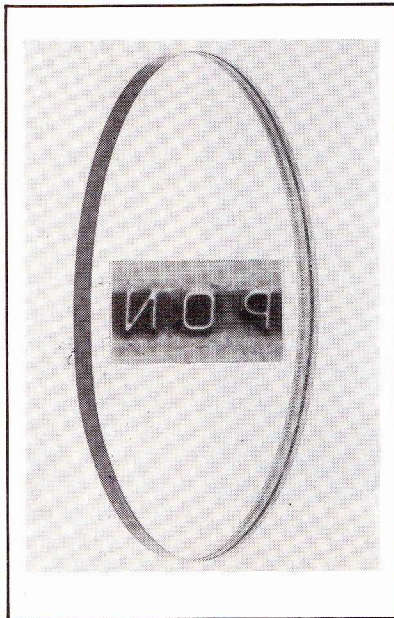
Cette technologie est celle actuellement employée par plusieurs

grands constructeurs tels que I.B.M., DATA PRODUCTS, CONTROL DATA, DOCUMENTATION, DATA PRINTER et CENTRONICS. Le principe en est le suivant : le moteur synchrone entraîne une bande métallique sur laquelle apparaissent les **caractères en relief** ; la bande tourne à une vitesse constante. Le jeu de caractères est répété plusieurs fois sur la bande. Sous chaque caractère apparaît une marque, également en relief, qui servira à la fois au contrôle de la vitesse de la bande et d'horloge (**fig. 7**).

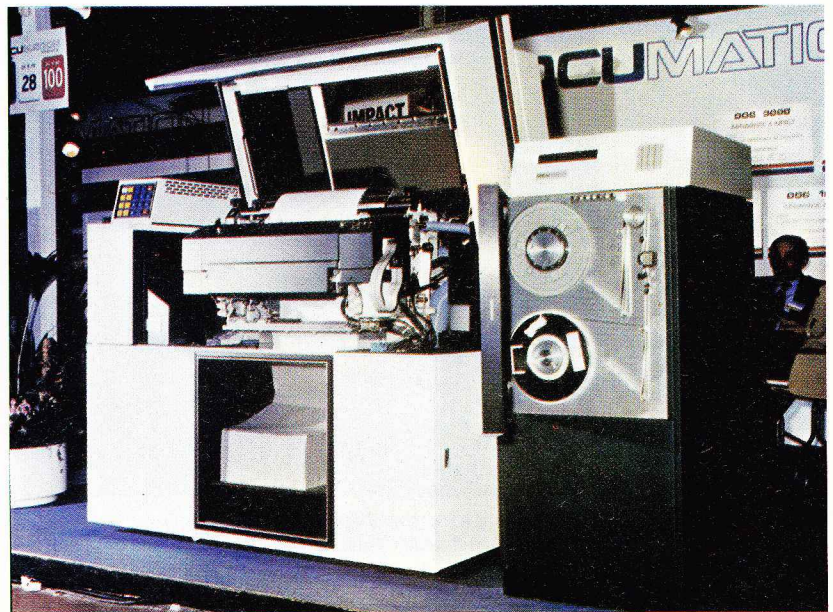
La logique, contrôlée par un microprocesseur 8 bits compare les caractères en place dans la mémoire d'impression (reçus par l'ordinateur) la position où les caractères doivent être imprimés (n° de colonnes) et l'endroit le plus proche où se trouvent les caractères sur la bande. Ainsi lorsque le caractère sélectionné passera devant la colonne reconnue valide (bonne à imprimer), la coïncidence déclenchera la commande d'un marteau et le caractère sera imprimé à la volée. La scrutation permanente des caractères sur la bande permet d'autoriser plusieurs fois l'impression du même caractère tout en gardant ce même caractère disponible en mémoire pour la ligne suivante. Ceci a pour effet de permettre des vitesses élevées 600, 1 200 ou 2 000 LPM.

Par rapport aux imprimantes à tambour ou même à chaînes (le principe des imprimantes à chaînes est à peu près le même que celui des imprimantes à bandes à deux différences près. La synchronisation de la vitesse de la chaîne est du type asservie, donc indirecte. La structure de la chaîne comporte un support et une partie rapportée sur laquelle se trouvent les caractères), l'imprimante à bande permet une grande souplesse dans le choix de la bande.

L'utilisateur peut changer lui-même la bande sans faire appel à un spécialiste. Aucun réglage spécial n'est nécessaire. La bande se remet normalement en bonne place toute seule.



Une bande et, au centre, le détail des caractères...



Une imprimante à bande Documation TPS 3000. Des vitesses d'impression de 3 000 LPM...

Les imprimantes à caractères générés par une matrice de points

A la fin des années 60, apparaissent les imprimantes matricielles à aiguilles développées notamment par CENTRONICS aux U.S.A. et LOGABAX en France. Cette méthode consiste, non plus à utiliser des caractères préformés, mais à constituer les caractères au fur et à mesure de ses besoins.

Les imprimantes matricielles

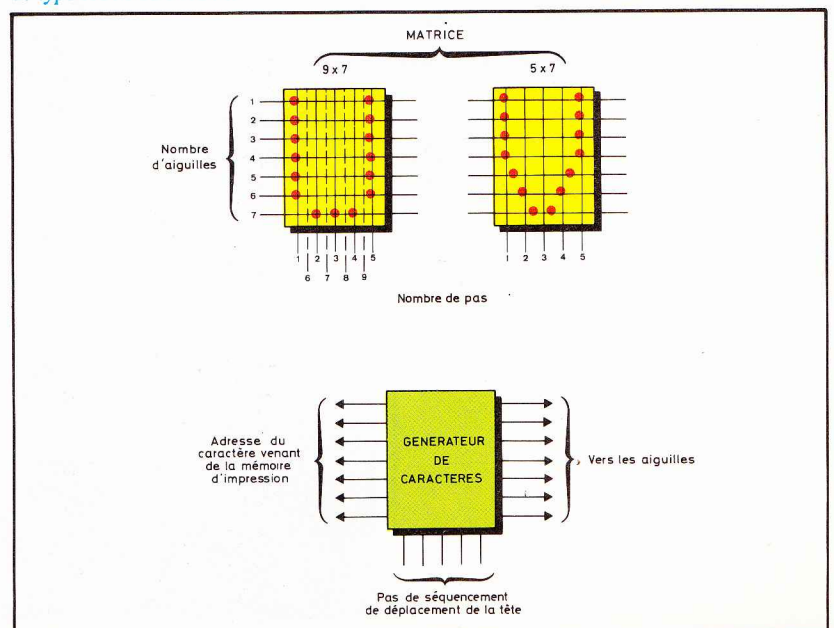
Elles sont appelées ainsi parce que le type d'impression utilise une matrice d'aiguilles.

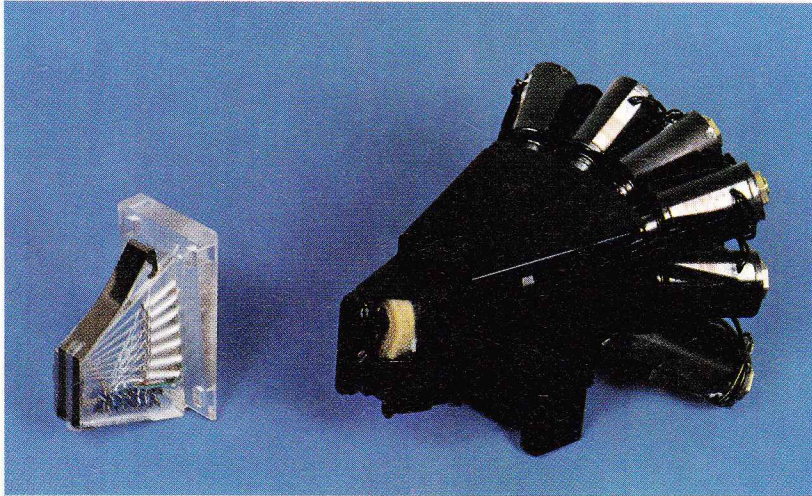
Qu'est-ce qu'une matrice ? C'est en quelque sorte un rectangle (ou un carré) dont L serait le nombre d'aiguilles et n le nombre de déplacements nécessaires à la tête pour « dessiner » un caractère. Cette définition s'exprime sous la forme d'un produit de deux chiffres ($n \times L$) dont le premier, n, est toujours le nombre de déplacements en « PAS » et le second, L, le nombre d'aiguilles de la tête d'impression.

Par exemple : 5×7 veut dire 5 pas nécessaires pour imprimer un caractère d'une hauteur de 7 aiguilles (fig. 8).

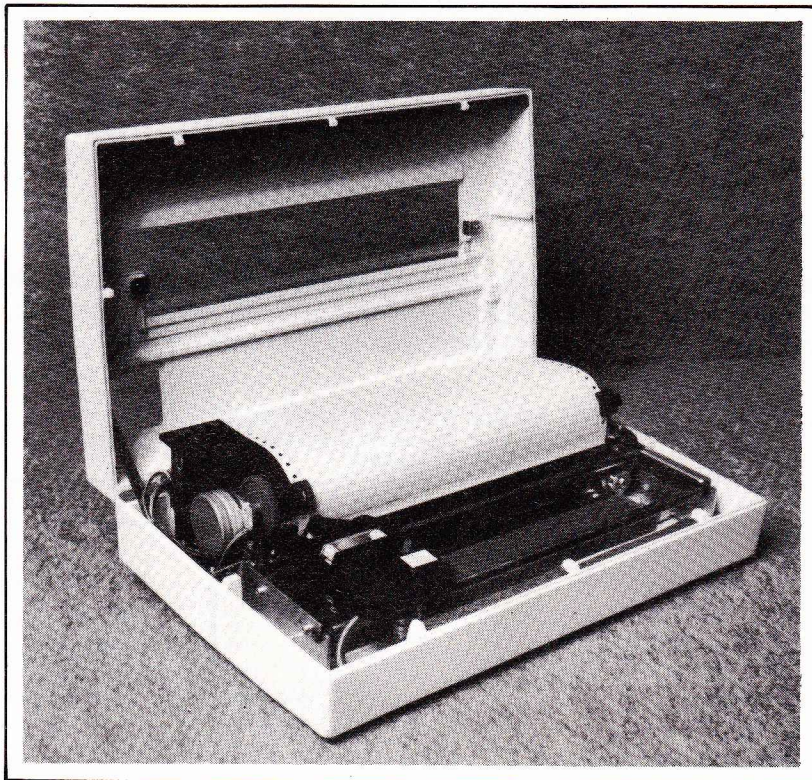
Le « parcours » logique de l'impression consiste à restituer sur le

Fig. 8. - Deux exemples de caractères générés par point à l'aide d'une matrice 9×7 et d'une matrice 5×7 . Le « dessin » de chaque caractère est préalablement stocké dans une mémoire de type ROM ou PROM.

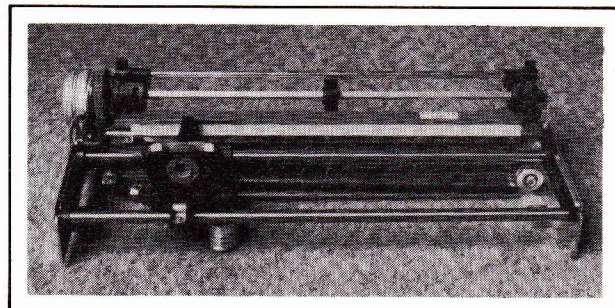




Deux exemples de tête d'écriture à aiguilles. Vous pouvez distinguer les aiguilles sur les extrémités gauches de chaque tête.



Une imprimante à aiguille et son mécanisme (Honeywell S 30).



papier le dessin du caractère, préalablement établi dans une mémoire (généralement ROM ou PROM). Le dessin sera inscrit dans le rectangle de référence constituant la matrice.

L'adresse d'un caractère à imprimer (en provenance de l'ordinateur) est écrite dans une autre mémoire (RAM). Lorsque l'on veut imprimer ce caractère on va lire la mémoire RAM et donner l'adresse de ce caractère au GÉNÉRATEUR DE CARACTÈRE. Quand la tête d'impression se déplace, le système qui pilote la tête génère un signal d'horloge permettant de séquencer le déplacement et de fabriquer les PAS. Chaque pas correspond à un top d'horloge qui vient également adresser le générateur de caractères. Chaque fois qu'un pas d'horloge de déplacement de la tête est émis, il lit une partie de l'adresse, donc de l'image contenue dans le générateur de caractères. La logique donne aussitôt l'ordre de commande des aiguilles correspondantes, ceci jusqu'à ce que le nombre de pas requis soit atteint. A ce moment une nouvelle adresse d'un caractère est envoyée au générateur de caractères.

Les imprimantes matricielles étant particulièrement bien adaptées au marché des micro-ordinateurs, examinons qu'elles sont leurs caractéristiques essentielles :

Le système d'impression

Il est généralement composé des parties suivantes :

- chariot se déplaçant le long de deux barres de guidage,
- tête d'écriture montée sur ce chariot,
- cartouche ou cassette contenant le ruban encreur,
- moteur d'entraînement assurant le déplacement de l'ensemble.

Le chariot se déplace sur deux barres d'acier dont une reçoit la bague principale du chariot et

Le point sur les imprimantes des micro-ordinateurs

Depuis deux ou trois ans l'essor de la micro-informatique a amené bon nombre de sociétés tournées peu à peu vers l'électronique et l'informatique, à développer leurs efforts dans ce nouveau créneau des « minis » imprimantes.

Si la révolution technologique est moins spectaculaire que dans d'autres domaines de l'informatique, il n'en reste pas moins vrai que la miniaturisation des composants électroniques et la compacité des mécanismes ont permis aux imprimantes de suivre harmonieusement l'évolution des micro-ordinateurs, notamment en ce qui concerne le rapport prix/performance.

Lors de la première boutique informatique, au SICOB 78, il était très difficile de trouver une imprimante dont le prix soit en rapport avec celui d'un micro-ordinateur.

A la lumière de ce que nous avons pu apprendre par les constructeurs, importateurs, boutiques micro et utilisateurs, nous avons essayé de dégager un certain nombre de tendances qui nous semblent caractériser ce segment du marché des imprimantes.

Ainsi, nous distinguons plusieurs niveaux :

Le marché lui-même

Les performances accrues des micro-ordinateurs, l'augmentation des capacités de mémorisation, l'orientation des applications vers la gestion des PMI, le rapprochement avec la mini-informatique font, en partie, évoluer les besoins des utilisateurs vers des imprimantes à chariot plus large (132 colonnes à 10 caractères au pouce) et plus rapide (120 à 180 caractères par seconde).

Ceci laisse présager que, dans les deux années qui vont s'écouler, le marché des imprimantes « moyenne vitesse » (100 à 250 CPS) devrait connaître une évolution identique à celle des imprimantes « basse vitesse », soit une multiplication du nombre des fabricants et une baisse de prix importante.

La percée de sociétés japonaises (OKI, TEC, NEC...) par le biais d'importateurs jouera pour cela un grand rôle.

Les caractéristiques des imprimantes

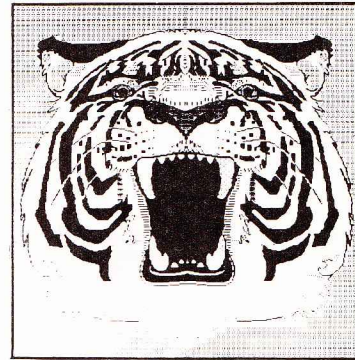
L'impression par matrice d'aiguilles sur papier normal reste, ici, la plus utilisée. De plus, les imprimantes tendent à s'adapter aux possibilités graphiques et de traitement de texte des micro-ordinateurs. Ainsi, l'IMP d'AXIOM est une véritable imprimante graphique.

La 737 de CENTRONICS qui possède une matrice N x 9 et qui **justifie à droite** répond aux petites applications de traitement de texte.

L'augmentation des possibilités de mémorisation permet de ne pas bloquer le calculateur pendant le temps d'édition.

En outre, le niveau sonore des imprimantes diminue (la tendance se porte vers 60 dB et moins) de même que leurs poids et leur volume.

Un exemple d'impression graphique réalisé sur imprimante à aiguille IDS 440.



L'après-vente

La notion de garantie n'est pas toujours facile à cerner et la durée est variable en fonction du vendeur et du type d'achat. En effet, deux cas peuvent se présenter :

■ L'imprimante est achetée en même temps que le micro-ordinateur, la garantie s'étend généralement à l'ensemble de la configuration (ceci d'autant plus si l'imprimante est celle figurant au catalogue du constructeur du micro-ordinateur).

■ L'imprimante est achetée seule : la durée de la garantie dépend souvent du constructeur d'imprimantes. Le vendeur final s'en tenant généralement à la répercuter à l'acheteur.

Compte-tenu du coût de plus en plus faible du matériel, il devient difficile à un constructeur ou à un importateur de proposer un contrat de maintenance incluant le déplacement du technicien chez l'utilisateur.

Plusieurs autres formules sont offertes à l'utilisateur parmi lesquelles on peut citer :

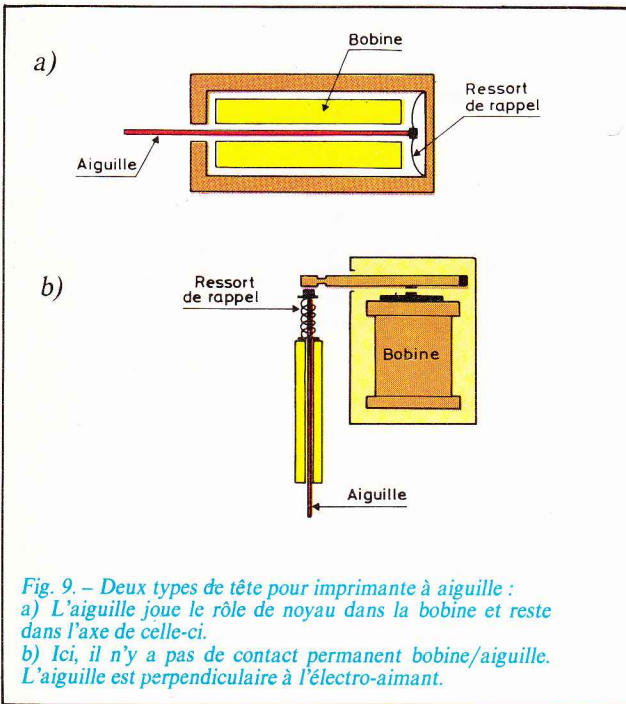
● L'utilisateur technicien assure la maintenance lui-même, suit une formation, utilise la garantie et se fournit en pièces détachées chez le constructeur lorsque la garantie est terminée.

● L'utilisateur non technicien souscrit un contrat d'extension de garantie (pièces et main d'œuvre) moyennant un retour du matériel chez le vendeur ou le constructeur.

● L'utilisateur ne souscrit pas de contrat et renvoie le matériel au coup par coup chez le vendeur ou le constructeur.

Les interfaces

C'est parfois pour l'utilisateur non spécialiste une limite dans son éventail de choix, malgré l'extension d'une sortie RS232C sur la plupart des micro-ordinateurs et des imprimantes. Mais un certain nombre de sociétés OEM et de boutiques commercialisent des interfaces spécifiques permettant d'adapter les configurations de micro-ordinateurs de façon beaucoup plus souple aux besoins des utilisateurs. ■



l'autre sert uniquement au guidage et à l'absorption du jeu éventuel bague-barre principale.

Selon les constructeurs ces deux barres sont placées sur un plan vertical ou horizontal. Du point de vue des avantages ou des inconvénients de ces deux possibilités rien de spécial n'est à signaler. Notons qu'en général il n'est pas nécessaire de graisser les barres ; les bagues étant « autolubrifiantes ».

■ Le ruban encreur

Deux principes sont utilisés :
 - Le ruban en deux rouleaux situés soit de part et d'autre du châssis, soit côte à côte à gauche ou à droite. Mais ce système présente deux inconvénients, celui d'avoir obligatoirement un système d'inversion, mécanique ou électrique et celui de manipuler le ruban lors de son remplacement.

Aujourd'hui la formule la plus

fréquente est la cassette non réutilisable qui possède les avantages de se manipuler très simplement et de n'avoir qu'une très faible partie du ruban à l'air libre.

■ La tête d'impression

Depuis 1970, date de naissance de la première imprimante utilisant la technologie matricielle, beaucoup de variantes ont vu le jour. Les têtes possédant 7 aiguilles puis 9 aiguilles sont disposées verticalement ou en quinconce. La figure 9 représente deux types de tête : solénoïde et aiguilles solidaires, solénoïde et aiguilles séparées.

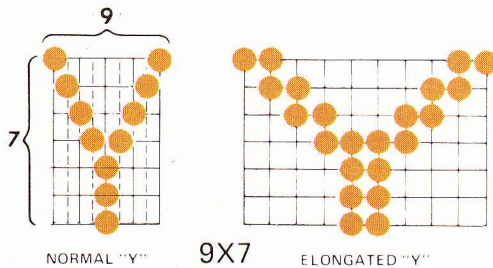
Une grande partie de l'évolution des imprimantes matricielles s'est orientée vers une plus grande vitesse. C'est ce qui a motivé toutes les recherches en matière de tête à aiguilles. Plus la vitesse augmente plus les électro-aimants de commandes sont sollicités. La

■ Caractères normaux.

YOU ARE LOOKING TO REDUCE — 10 caractères/pouce

YOU ARE LOOKING TO REDUCE — 12 caractères/pouce

YOU ARE LOOKING TO REDUCE — 16,5 caractères/pouce



■ Caractères élongués.

YOU ARE LOOKING TO REDUCE — 10 caractères/pouce

YOU ARE LOOKING TO REDUCE — 12 caractères/pouce

YOU ARE LOOKING TO REDUCE — 16,5 caractères/pouce

■ Ruban couleur et justification à droite sur la Facit 4540.

TEXT APPLICATIONS

The "sister ship" of the Facit 4542, the Facit 4540, has become the standard printer for proof reading and fast printout in typesetting applications because of its unique printout quality.

Fig. 10. - Quelques exemples d'impression obtenus sur imprimante à aiguilles.

■ Impression réalisée sur imprimante DRI 8180 (espacement proportionnel).

PASCAL MT+

L'ensemble livre comprend :

- . compilateur BCD.
- . compilateur en point flottant
- . éditeur de lien
- . aide à la mise au point symbolique et interactive
- . interfaces d'entrées/ sorties (run-time) en source et en objet
- . librairie complète de sous-programmes utilitaires
- . manuel en français
- . programmes d'exemples

CP/M est une marque déposée Digital Research
 Pascal Mt est une marque déposée MT
 microSYSTEMS

D.M.L. est représentant exclusif du produit
 Z.A. de Courtaboëuf tel: 928 01 31
 bâtiment AUVIDULIS
 avenue de l'océanie BP90-91402 ORSAY

Toutes les informations contenues dans ce document sont données à titre indicatif et sans aucun engagement.

vitesse moyenne aujourd'hui est de l'ordre de 180 CPS. Pour une matrice de 5 x 7, la fréquence de commande de chaque aiguille pourra être de 180 x 5 = 940 cycles par seconde, donc 940 Hz (proche de 1 kHz) d'où la nécessité d'avoir des aiguilles plus courtes. Des vitesses plus grandes sont ainsi atteintes (200 et même 300 CPS).

En outre, les constructeurs se sont orientés vers un marché nouveau où la vitesse n'est pas déter-

minante mais bien plutôt la qualité du caractère imprimé. Certes, une imprimante à aiguilles ne remplacera pas, pour la qualité du caractère, une imprimante du type marguerite ou tulipe où le caractère est « plein » mais peut s'en approcher au maximum en augmentant la noirceur du caractère.

Plusieurs solutions existent :

D'abord redéfinir la matrice. De 5 x 7 on est passé à 9 x 7 puis à 9 x 9. Cinq pas nuls sont toujours imprimés mais en augmen-

tant le nombre des points de référence dans le générateur de caractères on peut dessiner un caractère mieux formé, plus « joli ». Pour parvenir à cela on crée électroniquement des demi-pas entre chaque pas réel, augmentant ainsi le nombre de lignes d'adressage du générateur de caractère. Avec 9 aiguilles on va avoir la possibilité d'imprimer de vraies minuscules dont les jambages descendront sous les lignes des majuscules.

La figure 10 résume les diffé-

■ Résumé de ce que peut faire une imprimante matricielle (SEIKO ST 5150).

昭和55年 7月 4日

ST5150 PRINTER

☆ SEIKO SYSTEM ☆

!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu vxyz{|}~.「」, .ヲアイウエオカキクケコサシセソタツテトナニノハヒフヘホマミメヤヨリルレロワ°
 \~x÷≤≥+&§☆○◇□△▽※〒→←順和年曜日火水木金土


!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu vxyz{|}~.「」, .ヲアイウエオカキクケコサシセソタツテトナニノハヒフヘホマミメヤヨリルレロワ°
 \~x÷≤≥+&§☆○◇□△▽※〒→←↑↓±≠∞「」'""

!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu vxyz{|}~.!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu vxyz{|}~

0123456789@ABCDEFGHIJKLMN OPQRSTUVWXYZ
 0123456789@ABCDEFGHIJKLMN OPQRSTUVWXYZ
 0123456789@ABCDEFGHIJKLMN OPQRSTUVWXYZ

RAM DATA EDIT

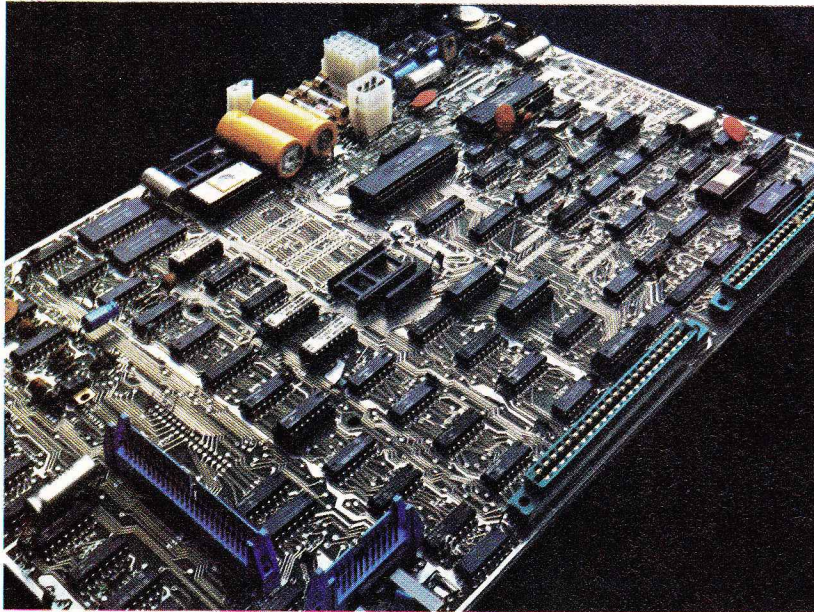
RAM DATA AND PROGRA



川口コーロ、ハナノエ、フ、メグルサ、カズキ、カケサシ、フ

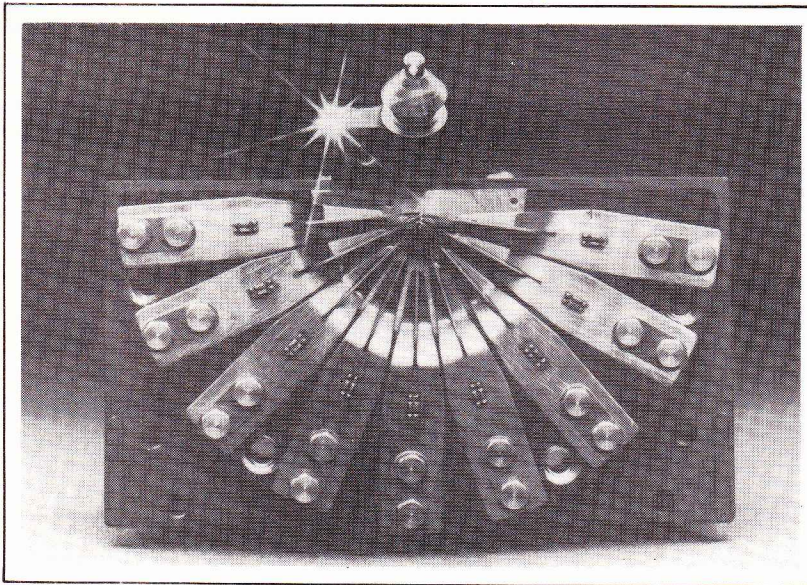
PAPER	PARADE	PARASOL	PARCH	PARDON	PARK	PARLOR	OPEN
-------	--------	---------	-------	--------	------	--------	------

Il semble qu'aujourd'hui on se dirige vers « l'imprimante » multifonctions.



La carte logique de l'imprimante Centronics 703. (Doc. Centronics).

Gros plan sur la tête d'impression de l'imprimante Facit 4540/42.



rentes possibilités d'une imprimante matricielle.

Pour s'approcher davantage du marché du traitement de texte on a vu apparaître de nouvelles techniques. Par exemple, l'espacement proportionnel qui consiste à faire varier par programme les espaces inter-caractères et les densités d'impressions (par exemple de 8 à

24 CPI). La définition du caractère devient très bonne, on ne voit presque plus les points d'impacts caractéristiques de l'impression matricielle.

L'impression ELONGUEE est constituée par un caractère dont chaque rangée d'aiguille significative est doublée, donnant ainsi un caractère de largeur double.

■ L'électronique

La partie électronique d'une imprimante est de plus en plus miniaturisée ce qui ne veut pas dire de moins en moins complexe.

Aujourd'hui, le microprocesseur est très employé dans la conception de ces imprimantes que l'on pourrait nommer de la 3^e génération. Elles ne possèdent plus en général qu'une seule carte électronique regroupant toutes les fonctions logiques et analogiques telles que :

- partie interface (parallèle ou série)
- partie CPU (processeur et ses mémoires)
- partie amplification (commandes des moteurs et des aiguilles)
- partie alimentation (fourniture des tensions nécessaires)
- partie logique de test (éventuellement).

Dans ce type de fonctionnement l'imprimante devient un véritable micro-ordinateur.

Tout est organisé autour du microprocesseur (8080, 8085, 6800, 6502, etc.) et de sa mémoire. Mémoire de fonctionnement contenant les micro-instructions nécessaires au pilotage de l'impression (ROM ou EPROM). Mémoire de stockage pour les caractères à imprimer venant de l'ordinateur ou pour les instructions transitant à l'intérieur de la logique d'impression et de commande (RAM).

Les nouvelles tendances

L'exemple de FACIT est un essai intéressant.

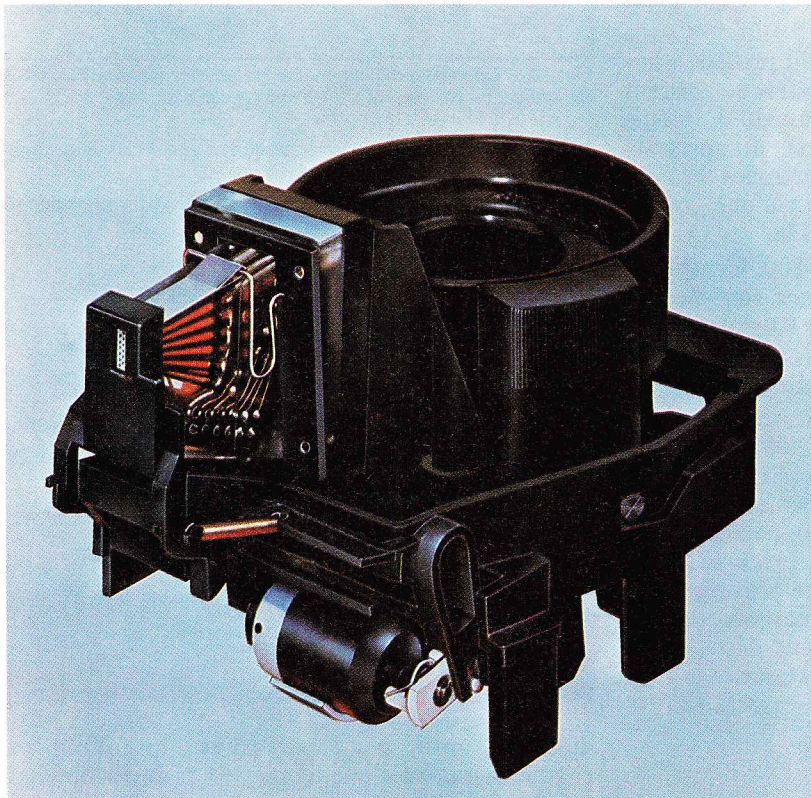
La tête à aiguilles désormais traditionnelle a été remplacée par des petits électro-aimants sur la palette desquels on a rapporté des stylets. L'extrémité de ces stylets est munie de pointes minuscules constituant la partie d'impact. L'intérêt de cette formule est l'absence de déplacement d'une aiguille puisque la course des palettes est très courte. Ceci devrait garantir un taux de fiabilité intéressant sans pour cela altérer la qualité de l'impression.

Les imprimantes sans impact

Imprimante à jet d'encre, électrostatique, électrolytique, thermique ou xérogaphiques, autant de types d'imprimantes sans impact.

L'impression sans impact est plus performante, plus fiable, et plus silencieuse. Elle est particulièrement bien adaptée pour les imprimantes très rapides lorsque le débit est supérieur à 10 000 lignes/mn.

Les systèmes d'impression les plus intéressants sont ceux dont le support est le papier ordinaire, après transfert (xérogaphie) et les imprimantes à jet d'encre.



Tête d'écriture à jet d'encre développée par Siemens (PT 80).

Les imprimantes à jet d'encre

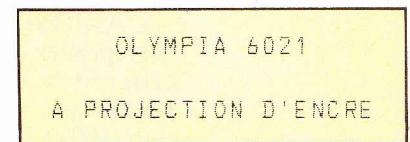
L'impression à jet d'encre présente comme singularité de pouvoir être utilisée à des vitesses faibles (inférieure à 100 caractères par seconde) comme à des vitesses extrêmement rapides, 45 000 lignes par minute (MEAD DIJIT)! Beaucoup d'études ont été consacrées à cette technologie depuis une quinzaine d'années, mais la seule machine qui paraît vraiment avoir rencontré un réel succès est l'I.B.M. 6640 avec 7 000 unités installées aux Etats-Unis dans des applications traitement de textes.

Le principe de ce type d'imprimantes repose sur la projection de gouttelettes d'encre sous pression à partir d'un ou plusieurs conduits. Des gouttelettes polarisées éjectées à un rythme constant traversent un champ électrique et sont ensuite déviées par des déflecteurs pour former les caractères sur le papier. A la sortie des conduits une gouttière permet de récupérer l'encre en cas de non impression. Les caractères sont dessinés point par point.

Ces imprimantes doivent être bien conçues pour contrôler parfaitement la fluidité de l'encre et empêcher les bavures.

La figure 11 montre un exemple d'impression à jet d'encre d'une nouvelle machine de traitement de texte développée par Olympia.

Fig. 11. - Un exemple d'impression à jet d'encre pour une machine de traitement de texte développée par Olympia.



Imprimantes électrostatiques

L'impression électrostatique présente l'avantage d'être silencieuse et rapide mais a l'inconvénient d'utiliser du papier d'un coût élevé sans fournir de copie. Ce type d'imprimante a surtout fait une percée sur le marché des imprimantes traceuses de courbes (GOULDVERSATEC).

Le procédé d'impression électrostatique permet d'atteindre une vitesse de 18 000 lignes par minute (PPS d'HONEYWELL).

Les traceurs de courbes (dont les vitesses peuvent dépasser les 3 000 lignes par minute) utilisent généralement une rangée d'électrodes fixées, alignées horizontalement, qui associées au mouvement du papier provoquent l'impression des caractères sous forme matricielle. Le papier traité spécialement est d'abord chargé positivement à l'emplacement des points sélectionnés par les électrodes.

Le passage dans un bain provoque l'attraction de particules d'encre chargées négativement et forme les caractères. L'encre est ensuite fixée par pression ou chauffage.

Les imprimantes à jet d'encre peuvent atteindre des vitesses extrêmement rapides : 45 000 lignes/mn.

Imprimantes électrolytiques et électrographiques

Ces deux procédés relativement voisins répondent généralement au même type d'applications et ont en commun d'utiliser un papier qui change de couleur en fonction de la tension de l'organe d'impression.

Le premier est dit procédé « humide » car les électrodes agissent sur du papier humidifié, tandis que le second est dit procédé « sec » car les électrodes brûlent un papier recouvert d'une couche d'aluminium (fig. 12).

L'AXIOM EX-800, Centronics microprinter, wenger, sont des imprimantes électrographiques 80 colonnes dont la vitesse varie entre 100 et 250 LPM.

Compte tenu de son coût peu élevé (inférieur à 3 000 F), de son faible encombrement et de son niveau sonore très bas et malgré la qualité et le prix du papier utilisé, ce type de matériel est aujourd'hui bien placé pour répondre aux besoins des marchés « grand public » qui peuvent découler de l'avènement télématique.

Imprimantes thermiques

L'impression thermique mise au point vers 1972 présente à peu près les mêmes avantages et inconvénients. Elle connaît un certain succès notamment par l'intermédiaire de Texas Instruments au niveau des téléimprimeurs et semble reprendre un second souffle sur le marché des HARD COPY* (I.B.M. Philips).

Il s'agit là encore d'une impression matricielle fondée sur une élévation de température des points de la matrice (en fonction des caractères à imprimer) provoquant le changement de couleur du papier **thermosensible**.

L'impression peut se faire aussi sur une colonne de 7 (ou 9) électrodes se déplaçant latéralement

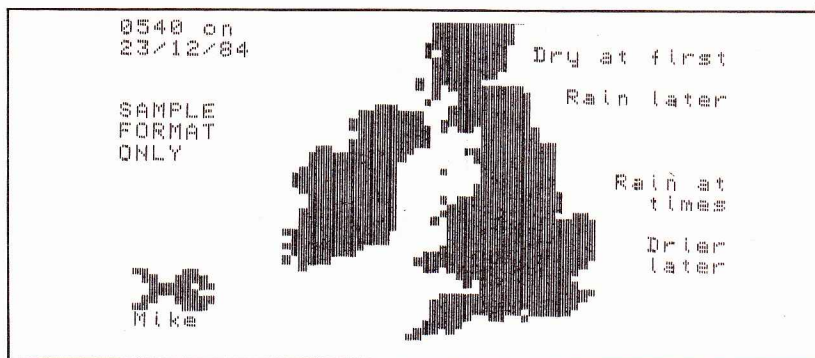


Fig. 12. — Un exemple d'impression réalisée sur imprimantes électrographiques. Les électrodes brûlent un papier recouvert d'une couche d'aluminium.

Un rayon laser modulé, à faible puissance, génère les caractères à imprimer (sous-système d'impression IBM 3800).



dessinant ainsi, point par point, chaque caractère d'une ligne.

La simplicité de ces imprimantes et leur faible coût rend leur emploi intéressant dans les calculatrices.

La vitesse moyenne des imprimantes utilisant ce type de procédé est de 30 caractères par seconde.

Les imprimantes xérographiques/les imprimantes à laser

En 1976 est apparue sur le marché l'I.B.M. 3800, aujourd'hui figure de proue des imprimantes xérographiques.

La xérographie consiste à dessiner au moyen d'un faisceau lumineux (laser) sur une couche pho-

tosensible puis à transférer le dessin obtenu sur le papier.

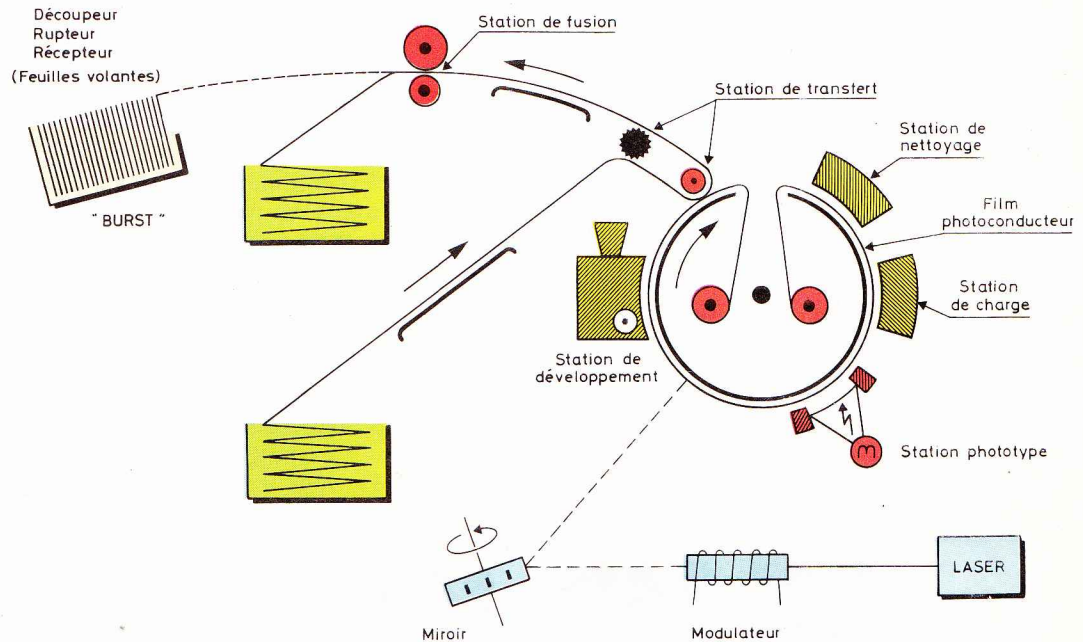
Le principe de l'impression du type LASER est donc sans impact. Il associe l'emploi d'un laser pour la génération des caractères à celui de l'électrophotographie pour l'impression. Ceci permet d'obtenir de très grandes vitesses d'impression effectuée **page par page** (vitesse moyenne 200 pages par minute soit 12 000 LPM) avec un maximum à ce jour d'environ 20 000 LPM.

La **figure 13** présente une vue schématique d'une imprimante xérographique et de ses différents organes.

Bien entendu, le coût extrêmement élevé de ce type d'imprimante (1 500 000 F) ne la destine qu'à des applications particulières, inhérentes à l'informatique « lourde ».

Fig. 13

Principe des imprimantes xérogaphiques (à laser)



Déroulement des opérations :

Le tambour central est recouvert d'un film photoconducteur qui est chargé électrostatiquement par une station spécialement conçue à cet effet. Lorsque l'impression est commandée, chaque caractère à imprimer est traduit à l'aide d'une table de conversion, en une série de points correspondant ou non à une insolation de la surface du film. Cette insolation est réalisée par un modulateur qui allume ou éteint le faisceau du laser, suivant que le point doit être noir ou blanc. Ensuite, la surface insolée du film est amenée par la rotation du tambour dans une station où il y a dépôt par effet électrostatique, d'une poudre noire dont les particules se fixent sur les endroits insolés. Puis le film passe avec le papier servant de support d'impression, dans la station de transfert. Ici, l'envers du papier est chargé positivement, ce qui a pour effet d'attirer sur le papier les parti-

cules de poudre qui résidaient sur le film. Enfin, le papier passe dans la station de fusion entre deux rouleaux, dont l'un est chauffé et dont l'autre va servir de compresseur, afin de faire incruster définitivement la poudre dans le papier. Un système optionnel permet d'éclater le papier et de découper la bande Caroll, au lieu d'entasser le papier en accordéon.

Dimensions de l'impression (document CISI)

Dimensions hors tout du papier : 303 x 215 mm.
 Largeur imprimable : 279 mm (11 pouces).
 Nombre de caractères par pouce en standard : 15 CPI.
 Nombre maximum de caractères par ligne : 150.

Les stations...

- Stations de nettoyage et de chargement électrostatique :

Système déchargeant et chargeant électrostatiquement le film photoconducteur.

- Laser - Modulateur - Miroir :

Système permettant d'impressionner le film photoconducteur.

- Station de développement :

Système d'encrage par effet électrostatique.

- Station de transfert :

Système permettant le transfert sur le papier de l'information préalablement déposée sur le film photoconducteur.

- Station de fusion :

Système permettant de plaquer définitivement les caractères sur le papier.

- Station phototype :

Système permettant de photocopier sur le film l'image d'un phototype.

- Station découpeur-rupteur :

Système éclatant les pages d'un listing en découpant la bande CAROLL et en mettant les pages les unes derrière les autres.

L'avènement de la télématique devrait faire naître un marché pour de petites imprimantes « copie d'écran » à prix très bas.

Périphériques

Conclusion

Avec l'évolution de la micro-informatique, les imprimantes matricielles ont encore quelques belles années devant elles.

Certaines études prévoient un doublement du parc de ce type de machines entre 1979 et 1982 pour atteindre plus de 170 000 unités. Graphisme, matrice de points à haute densité et couleur devraient permettre à ces imprimantes de plus en plus sophistiquées de devenir multi-fonctions et de pouvoir répondre à des applications scientifiques, de gestion (BARCODE, OCR) ou de traitement de texte.

Ce dernier marché dont on parle beaucoup actuellement semble évoluer dans deux directions opposées en matière d'impression. D'une part apparaissent sur le marché des imprimantes marguerites à faible vitesse (autour de 20 CPS) proposées par TEC, OLYMPIA, ... et faible coût et

d'autre part des développements permettant d'augmenter la vitesse des imprimantes marguerites traditionnelles sont réalisées notamment par QUME (Twin Track).

De nouvelles technologies peuvent apparaître dans ce domaine, à commencer par celle de la Quietwriter annoncée par CENTRONICS qui repose sur le principe d'un styilet se déplaçant dans un espace à trois dimensions. Par pression plus ou moins forte sur le ruban encreur, cette imprimante permet de reproduire l'écriture humaine (avec les pleins et les déliés) à la vitesse d'une imprimante de type marguerite. La technologie du jet d'encre parfaitement maîtrisée est également une réponse possible pour ce type d'application.

Des imprimantes sans impact de type xérographique (600 à 6 000 LPM) vont probablement

remplacer à terme les imprimantes à bande dont la part de marché devrait commencer à décroître vers le milieu des années 80.

Dans le domaine du laser les perspectives sont très largement ouvertes puisque l'idée d'un matériel multi-fonctions, pouvant être utilisé indifféremment comme photocopieur ou comme imprimante, semble faire son chemin.

Les entreprises de Presse et les sociétés d'Édition sont concernées par ces évolutions qui peuvent permettre d'améliorer la technologie de la photocomposition en simplifiant à l'aide du laser la production des plaques Offset. L'avènement de la télématique devrait de son côté faire naître un marché pour de petites imprimantes « copie d'écran » à prix très bas. ■

G. BICOS
S. ANDRE

LE HP 85 A UN N° DE TÉLÉPHONE : 627.23.57

En vous adressant à LTA (Logiciels Thèmes Applications), vous saurez tout sur les prodigieuses capacités du HP 85, l'ordinateur Hewlett-Packard.

Indépendamment du HP 85 en lui-même, LTA vous informera sur toutes les applications du HP 85 : calculs mathématiques et scientifiques, gestion des stocks, gestion des fichiers et des payes, gestion de portefeuilles, calculs micro et macro économiques.

N'hésitez pas. Téléphonnez à LTA et parlez leur du HP 85 : c'est leur passion.

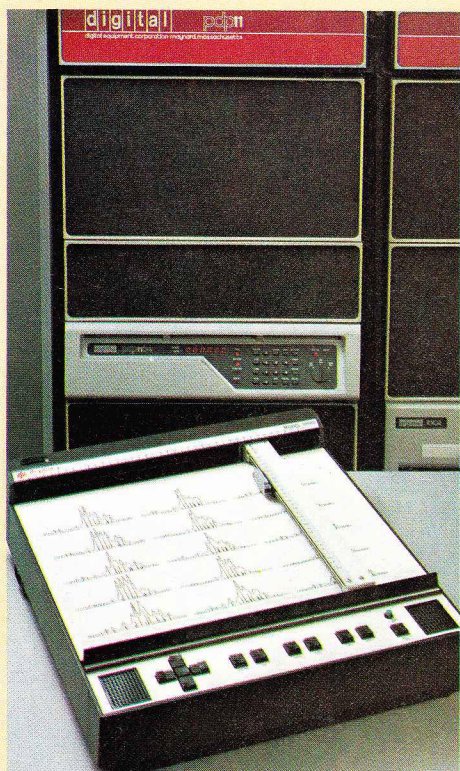
LTA. 154, rue Cardinet 75017 Paris, tél. : 627.23.57



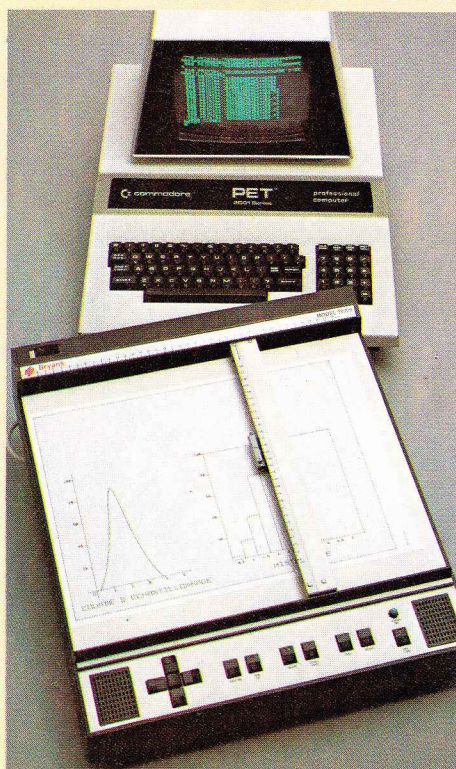
Micro Expo. Stands 89.90.91

Computagraph

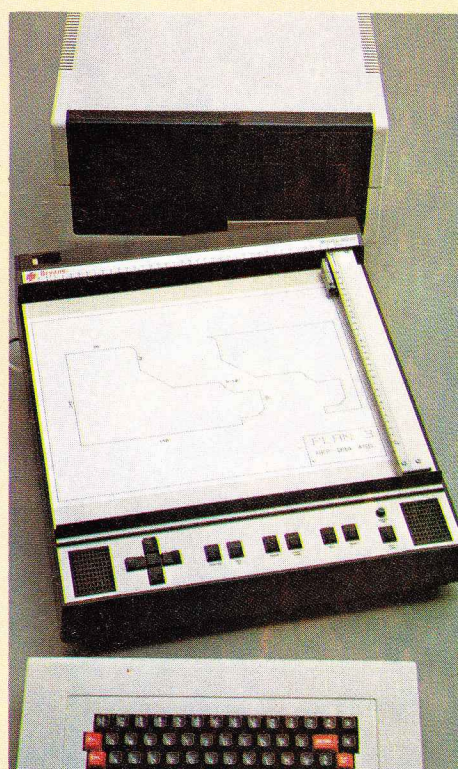
de Bryans



En périphérie des gros ordinateurs tel le Digital PDP 11, le Computagraph satisfait en toutes circonstances aux applications scientifiques et industrielles les plus sophistiquées.



Relié directement à tout micro ou mini-ordinateur, ici le PET Commodore, le Computagraph constitue l'outil pédagogique idéal pour la simulation et la représentation graphique.



Etude de pièces, visualisation et modification, tracés de plans, etc. Avec le Computagraph, le dessin automatisé pénètre en force dans tous les bureaux d'études.

trace tous les concepts

en une ou six couleurs

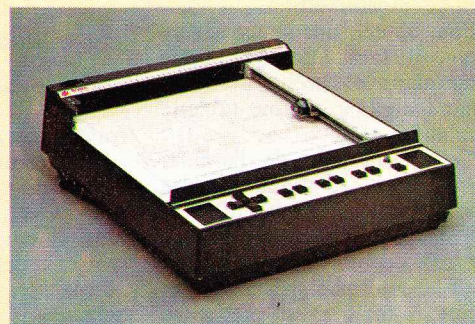
Traceur XY numérique format A3, le Computagraph permet d'accéder à moindre coût au dessin automatique et à tous travaux impliquant la représentation de courbes $y = f(x)$.

Il est connectable directement à tout micro ou mini-ordinateur, et par MODEM, en temps partagé, à toute Unité Centrale.

Caractéristiques principales : • Connexion directe via BUS IEEE 488 ou RS 232 C (V24) • Intelligence et génération de caractères incorporées • Mémoire tampon de 2048 caractères (2K octets).

Options :

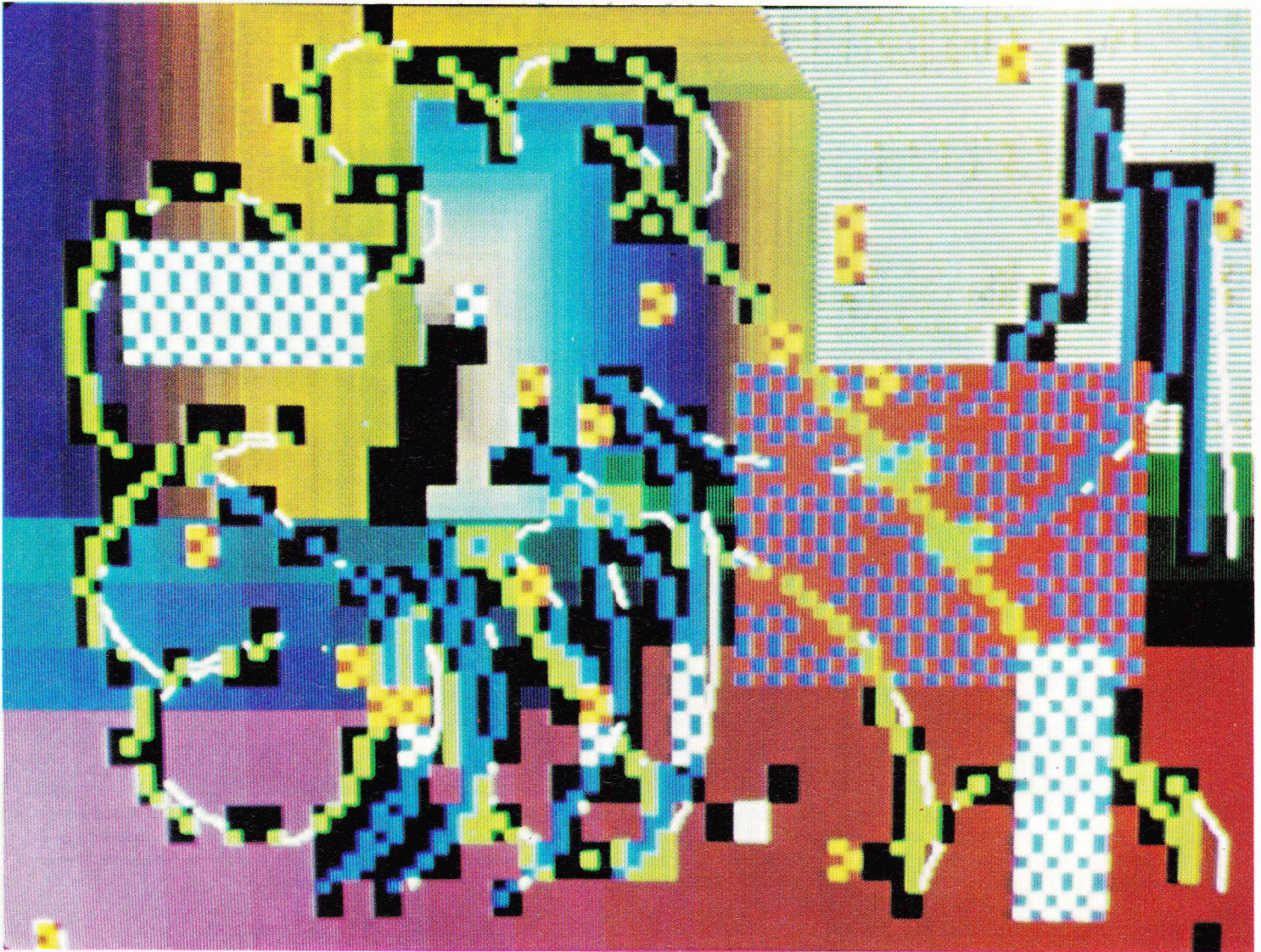
• Mémoire extensible à 4096 caractères • Traceur XY numérique transformable en XY analogique par simple changement de module • Avance automatique de papier (montage ultérieur possible).



Computagraph A3 version multicolor 6 couleurs.

Notre couverture :

Le principe « d'émergence »



Aujourd'hui, en recherche graphique, l'ordinateur est presque toujours considéré comme un outil de transformation.

Pour obtenir une image sur un écran, il faut introduire cette image dans le système avec une caméra digitale ou une tablette graphique.

Cette utilisation de l'ordinateur est symptomatique de notre époque qui privilégie la prolifération des « boîtes noires ». On ne sait pas « comment ça marche » (sauf les multiples spécialistes, et encore...) mais on se préoccupe surtout de ce qui « entre » et de ce qui « sort ».

Dans ce système de « boîte noire », le procédé de transformation de l'image est toujours indifférent à la nature de l'image à

Il y a « émergence » quand de nouvelles conséquences apparaissent, quand de nouvelles formes « émergent » sur l'écran.

transformer. C'est l'époque des « hommes-digitaliseurs » et des « hommes-manipulateurs d'images ». C'est à qui la coupera le plus finement en petits morceaux, la fera la mieux tourner, la coloriera de manière la plus agressive...

Dans ce cas, le programme informatique est complètement indifférent à la nature de l'image, il n'est pas du tout « intime » avec elle.

L'ordinateur est alors consi-

déré comme un simple outil de transformation et utilisé de manière « triviale ». Il n'apporte que des effets techniques supplémentaires à l'artiste. L'ordinateur est utilisé de manière fermée.

Néanmoins, il est possible de l'utiliser de manière ouverte, tout simplement, « parce que l'ordinateur, c'est avant tout un formidable système de manipulation de symboles » *. C'est bien la première machine qui peut émettre des choix par elle-même, à partir

des règles du jeu qui sont fixées par le « créateur de programme ». Et, l'utiliser de cette façon est bien sûr beaucoup plus excitant, l'ordinateur devient un outil qui permet d'aller plus loin dans la découverte de son propre processus créatif.

Prenons l'exemple du peintre. Quand il met une tache de couleur dans son tableau, il « pèse » tout le poids de cette tache par rapport à l'ensemble et, son prochain geste tiendra compte de cette tache-là. La forme et la couleur de sa tache sont liées intimement au reste du tableau et à sa manière de peindre. Cette

* Et pas seulement de symboles arithmétiques.

tache est « intime » à sa création de peintre.

Si ce peintre, à force de « métier » arrive à se pencher sur le « comment ça marche » de la boîte noire de son cerveau, il pourra alors énoncer quelques règles qui sont valables pour lui (peut-être pour d'autres...). Par exemple, une tache rouge précède généralement un trait jaune à l'autre bout du tableau ou des hachures dans la forme vide la plus proche...

A ce moment-là, avec ces quelques règles en poche, il peut en faire un programme simple, puis, de fil en aiguille, le compliquer.

L'ordinateur est ici employé de manière optimale, c'est-à-dire comme **manipulateur de symboles**.

Le peintre fait un chemin avec l'ordinateur, il l'utilise au mieux de ses capacités au fur et à mesure que son analyse et son exigence s'aiguisent.

Comme avec ses pinceaux, l'homme définit sa palette et ses règles de composition, que l'on peut regrouper en règles de structures et de succession.

Prenons, par exemple, un choix de couleur pour la palette : rouge, vert, bleu, et des règles de structure :

- ligne/courbe
- vide/hachure
- épais/fin
- ouvert/fermé
- bien placé/mal placé
- transmissible/non transmissible.

Nous pouvons donner alors quelques règles transitoires de succession :

- trait blanc devient point épais vert ou bleu,
- mode épais suit mode fin,
- après un trait rouge, un carré noir,
- hachures en haut à gauche amènent un renforcement de la diagonale tragique.

Ce sont ces règles qui constituent l'ossature du programme. C'est l'ordinateur qui détermine, **choisit** sur l'écran les éléments à tracer en fonction de ces règles de structure et de succession qui sont le reflet des choix du peintre face à la toile.

Au fur et à mesure de l'évolution des résultats, l'artiste modifie les règles ou la palette à son gré, les simplifie ou les complique. La complexité du programme augmente avec la longueur du dialogue artiste-ordinateur.

Les premiers essais du programme consistaient à situer des formes les unes par rapport aux

autres dans l'espace de l'écran ; à présent, des commandes de structure comme celles de transmission sur réseau permettent d'obtenir des images complexes dont une partie (partie épaisse) est

Il y a « **émergence** » quand de nouvelles conséquences apparaissent, quand de nouvelles formes « émergent » sur l'écran. A l'artiste d'en tirer profit. Seul l'emploi de l'ordinateur de cette ma-

connaissons actuellement entre les hommes et les machines. En effet, alors que des progrès gigantesques ont été faits du point de vue technique (réduction de la taille des systèmes, augmentation de vitesses de calcul), les progrès quant à cette relation vitale sont presque inexistantes.

Ces machines à « computer » ont d'abord été créées par une « **culture minoritaire** » celle des techniciens et des informaticiens et ensuite utilisées pour des applications comme toute particulières, des applications de transformation technique.

Il est temps que les poètes et les créateurs utilisent les ordinateurs, qu'ils définissent leur manière de dialoguer avec ces machines, d'en modifier la conception et d'ouvrir de nouvelles voies.

Pour aller plus loin, il est temps aussi de faire basculer des tabous tenaces comme celui de l'affectivité des machines. Bien malin celui qui aujourd'hui peut affirmer radicalement que les machines n'ont pas d'affectivité.

Un fait est certain, plus la recherche avance dans le domaine de l'intelligence artificielle, (travaux du Docteur Feigenbaum à Stanford et du Docteur Papert au MIT) moins il apparaît que l'on puisse définir radicalement de frontières entre l'intelligence et l'affectivité.

Si les ingénieurs qui sont « intelligents par définition » jurent que les ordinateurs ne sont que des machines intelligentes, pourquoi les artistes qui sont « affectifs par définition » ne découvrirait-ils pas que les machines sont aussi affectives ? ■

Bernard DEMIAUX



Une partie des images peut être codée en caractères vidéotex donc transmissible par téléphone sur les réseaux Teletel ou Eurodial...

codée en caractères vidéotex donc transmissible par téléphone sur les réseaux TELETEL ou EURODIAL par exemple ; l'autre partie (partie fine) est codée en haute définition, c'est sans doute la future génération des « **transmissibles** ».

Ce qui est intéressant ici, c'est l'idée que sur la même image (voir page de couverture et illustrations), une partie peut être transmise/diffusée sur les réseaux habituels et que l'autre partie puisse être retenue ou pas encore transmise.

Est-ce que cette image à plusieurs niveaux ne symbolise-t-elle pas globalement le phénomène de communication dans sa complexité même !!

Ce qui est troublant, c'est que dans ce cas, c'est l'ordinateur qui définit ce qui peut être « transmis » et ce qui peut être « retenu » dans l'image. **L'artiste a défini/programmé les règles, l'ordinateur fait les choix.**

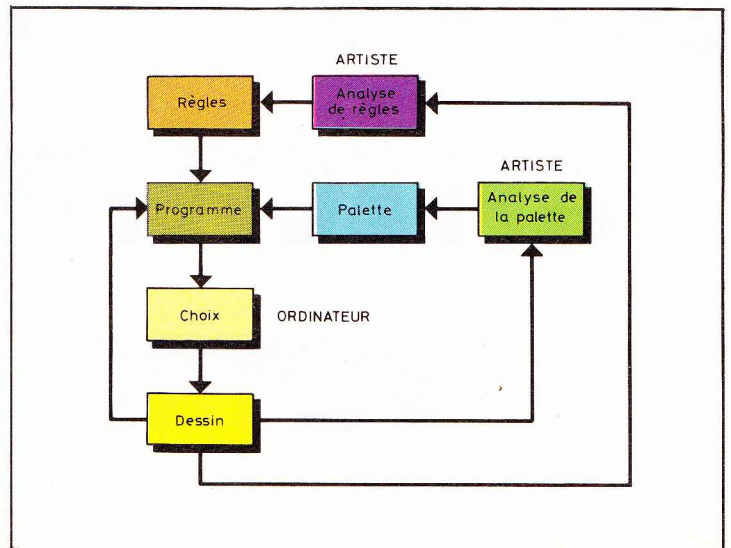
Ce cheminement met en relief deux points fondamentaux. Tout d'abord la notion « **d'émergence** ». Quand l'artiste formule une/des règles de composition, il n'a pas toujours une idée précise des conséquences des règles. On peut très bien maîtriser les règles mais pas les conséquences.

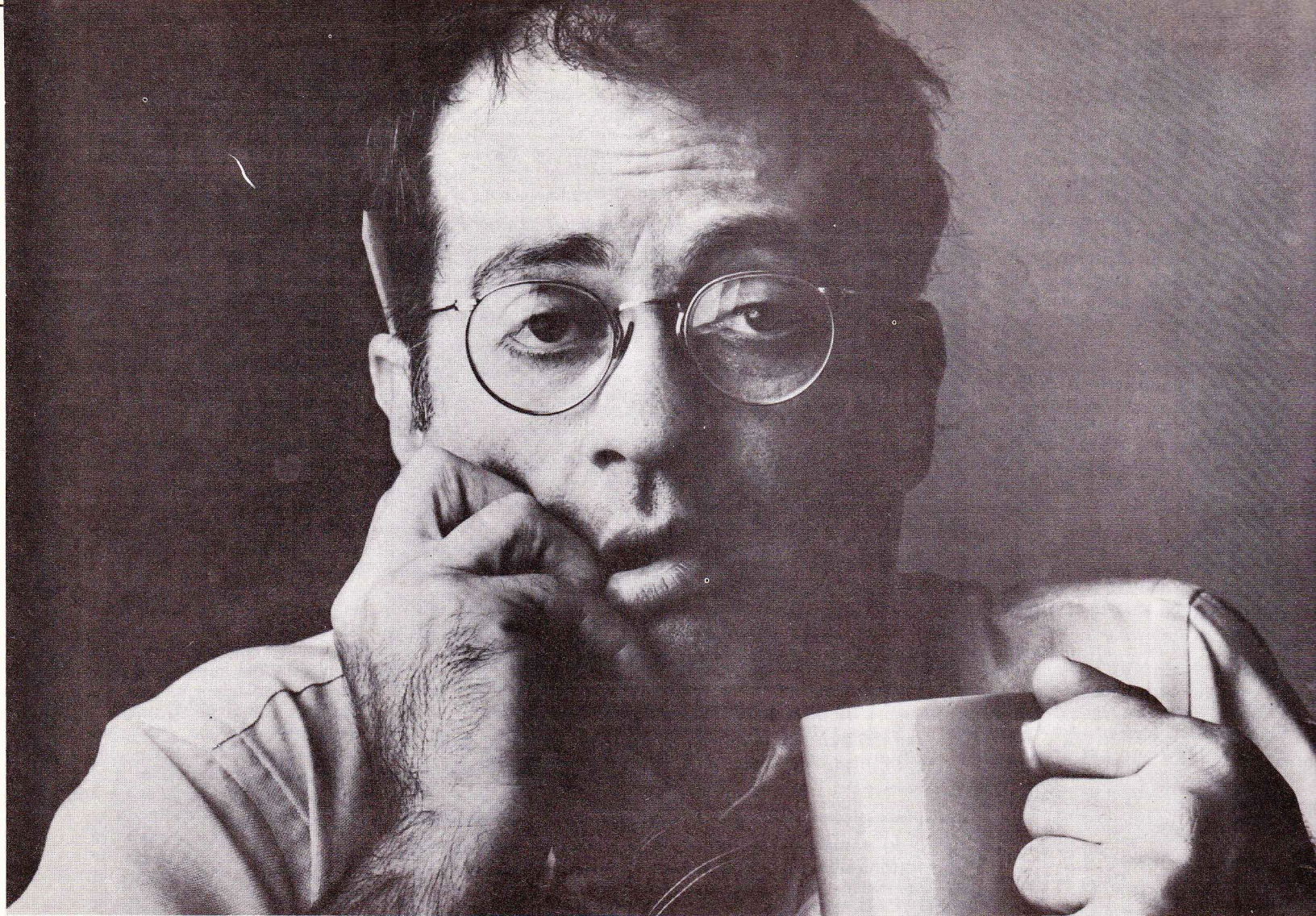
nière permet de mettre en relief cette notion et de l'utiliser de façon créative.

L'algorithme qui sert de base au programme du peintre sera d'autant meilleur qu'il reproduira les comportements de l'artiste face à ses couleurs et ses formes.

L'autre point fondamental est cette notion de **dialogue artiste-machine**. Il est encore à inventer et en tout cas, il sera sûrement différent des dialogues que nous

Fig. 1. - Symbolisation de la relation artiste/ordinateur.





**LES ERREURS ALEATOIRES
AFFECTENT SUREMENT
VOTRE SYSTEME**

Deuxième partie
de
notre dossier :

les microprocesseurs

16 bits

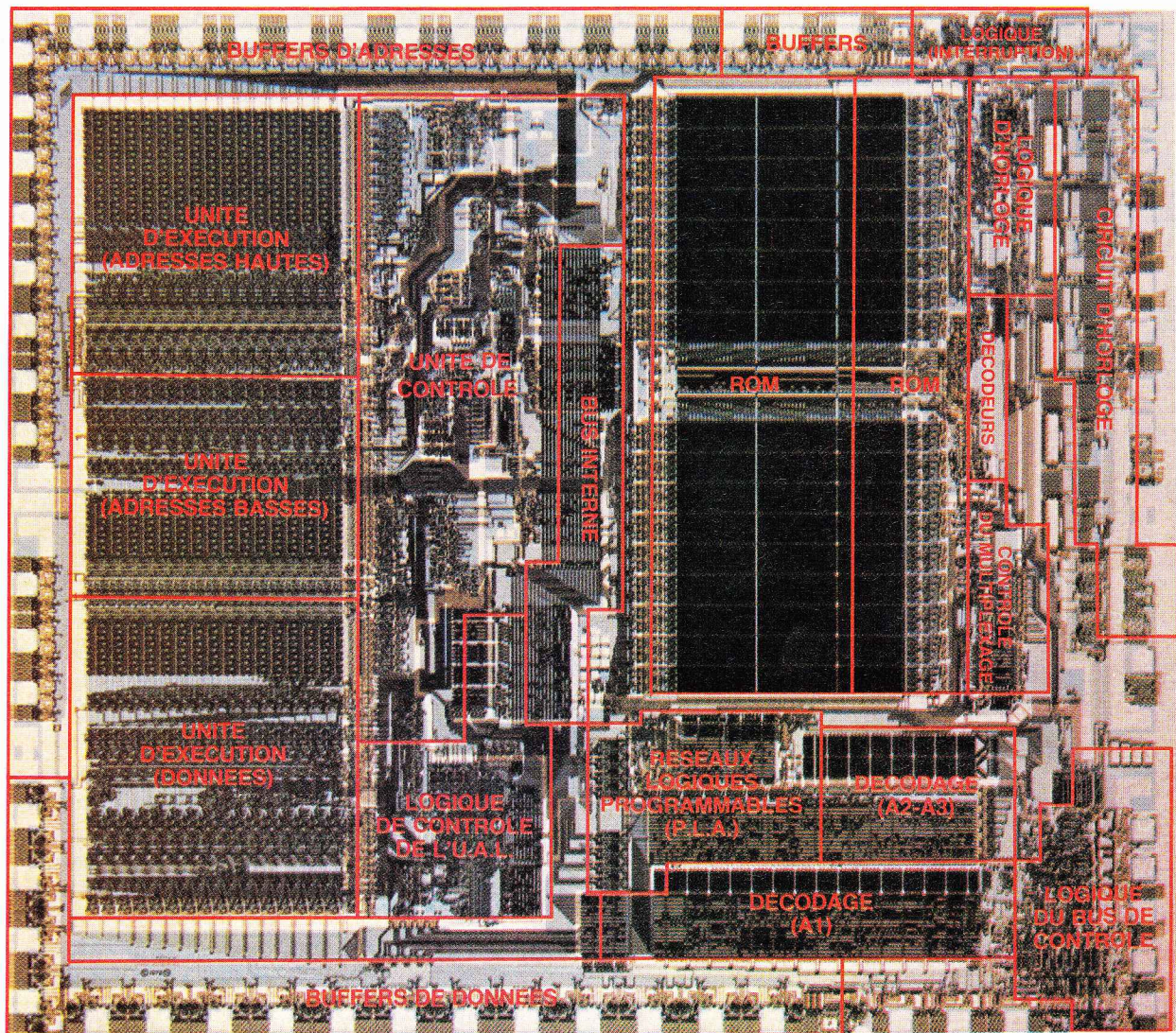


Photo 1. - Vue interne du 68000 : 70 000 composants actifs sur cette « puce » de 6 à 7 mm de côté !

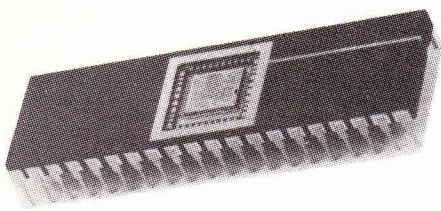
Les microprocesseurs 16 bits constituent certainement aujourd'hui les composants les plus évolués de la technologie micro-électronique.

Dans notre précédent numéro, nous avons analysé le marché actuel des microprocesseurs 16 bits et l'évolution de leurs caractéristiques depuis la naissance des « anciens » microprocesseurs 16 bits, il y a maintenant 5 ans. Cette approche nous a permis de définir ce que devrait être, à notre avis, le microprocesseur 16 bits « idéal » pour construire le plus efficacement possible un micro-ordinateur (mini ?) :

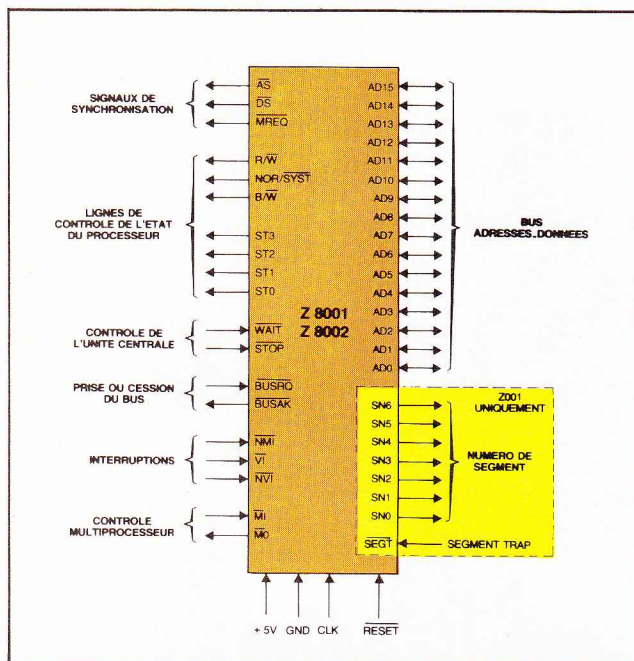
- boîtier de 40 broches seulement,

- opérations sur 16 bits (y compris la multiplication et la division),
- espace adressable de 512 K-octets sans segmentation,
- compatibilité au niveau du langage binaire, avec un « 8 bits » connu.

A titre de comparaison, nous avons étudié les caractéristiques d'un microprocesseur 16 bits largement diffusé : le 8086 d'INTEL. Nous vous proposons maintenant le Z8000 de Zilog et le 68000 de Motorola.



Le Z 8000



Le Z 8000 de Zilog est un microprocesseur 16 bits réalisé en technologie N-MOS. Une de ses caractéristiques les plus intéressantes concerne ses possibilités d'adressage et de gestion de l'espace mémoire.

Deux versions du Z 8000 sont actuellement disponibles (fig. 1) :

- Le Z 8002 présenté dans un boîtier de 40 broches possède un espace mémoire adressable limité à 64 K-octets (16 fils d'adresse).
- Le Z 8001 livré dans un boîtier de 48 broches est une version dite « segmentée », du Z 8000.

Ainsi, à l'aide d'un boîtier supplémentaire appelé MMU (Memory Management Unit), il peut gérer un espace de 16 M-octets. Le circuit de gestion MMU contrôle à cet effet 24 bits d'adresse physique.

Bien entendu le Z 8001 peut aussi être programmé en mode « non segmenté » (à l'aide du bit SEG du registre d'état).

Deux versions du Z 8000. Le Z 8002 ne comporte que 40 broches et ne peut adresser que 64 K-octets, tandis que le Z 8001 (48 broches) est une version segmentée permettant d'étendre l'espace adressable à 16 M-octets.

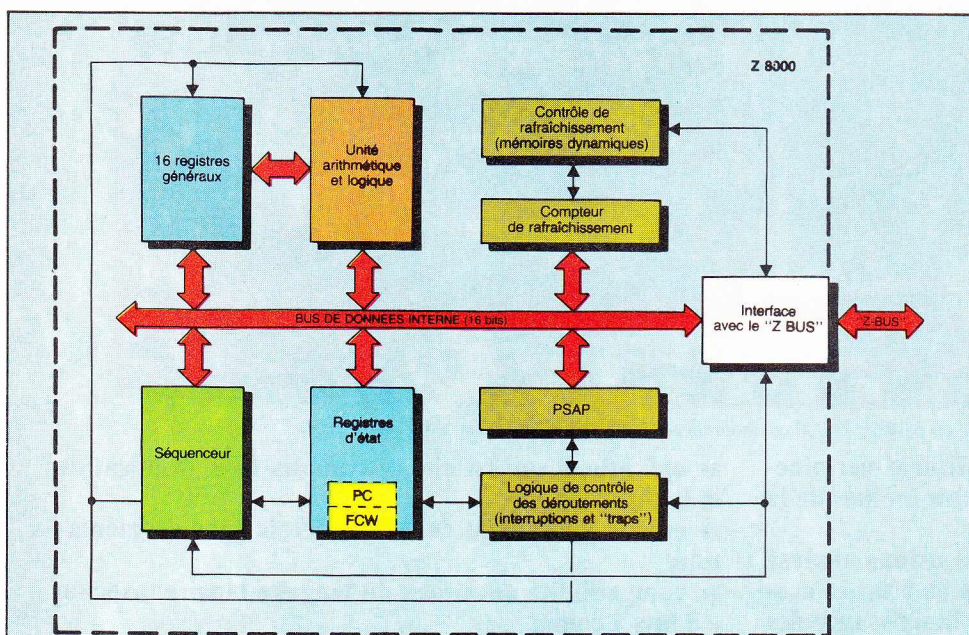


Fig. 2. - Schéma bloc représentant la structure interne du Z 8000. On y distingue :

- Un bus de données interne de 16 bits.
- Une logique d'interface avec le « Z-bus ».
- Un ensemble de 16 registres généraux.
- L'unité arithmétique et logique.
- Une unité de séquençage permettant la recherche et l'exécution des instructions.
- La logique de contrôle des dérivements (interruptions et traps).
- Les circuits permettant le rafraîchissement des boîtiers de mémoire RAM dynamique, externes.

Organisation interne :

Le Z 8000 dispose de 16 registres universels de 16 bits chacun (fig. 2 et 3).

Ces registres généraux peuvent, selon la programmation, devenir accumulateur, compteur, ou pointeur de pile...

Cette configuration autorise 6 modes d'adressage utilisables grâce aux instructions principales ainsi que la manipulation de types de données variés (mots de 16, 32 bits, digits BCD, ...).

La désignation d'un des 16 registres généraux se fait par les 4 bits du champ « adresse » de l'instruction.

Les huit premiers registres ($R_0 - R_7$) sont accessibles par moitié ($R_H - R_L$), ce qui permet à l'utilisateur de travailler au niveau de l'octet.

Les instructions peuvent aussi manipuler des « doubles mots », dans ce cas les registres sont groupés par paires. Ceci permet de considérer le Z 8000 comme un microprocesseur « 32 bits » disposant de huit registres généraux.

En outre, et sous certaines conditions, le Z 8000 a la possibilité de manipuler

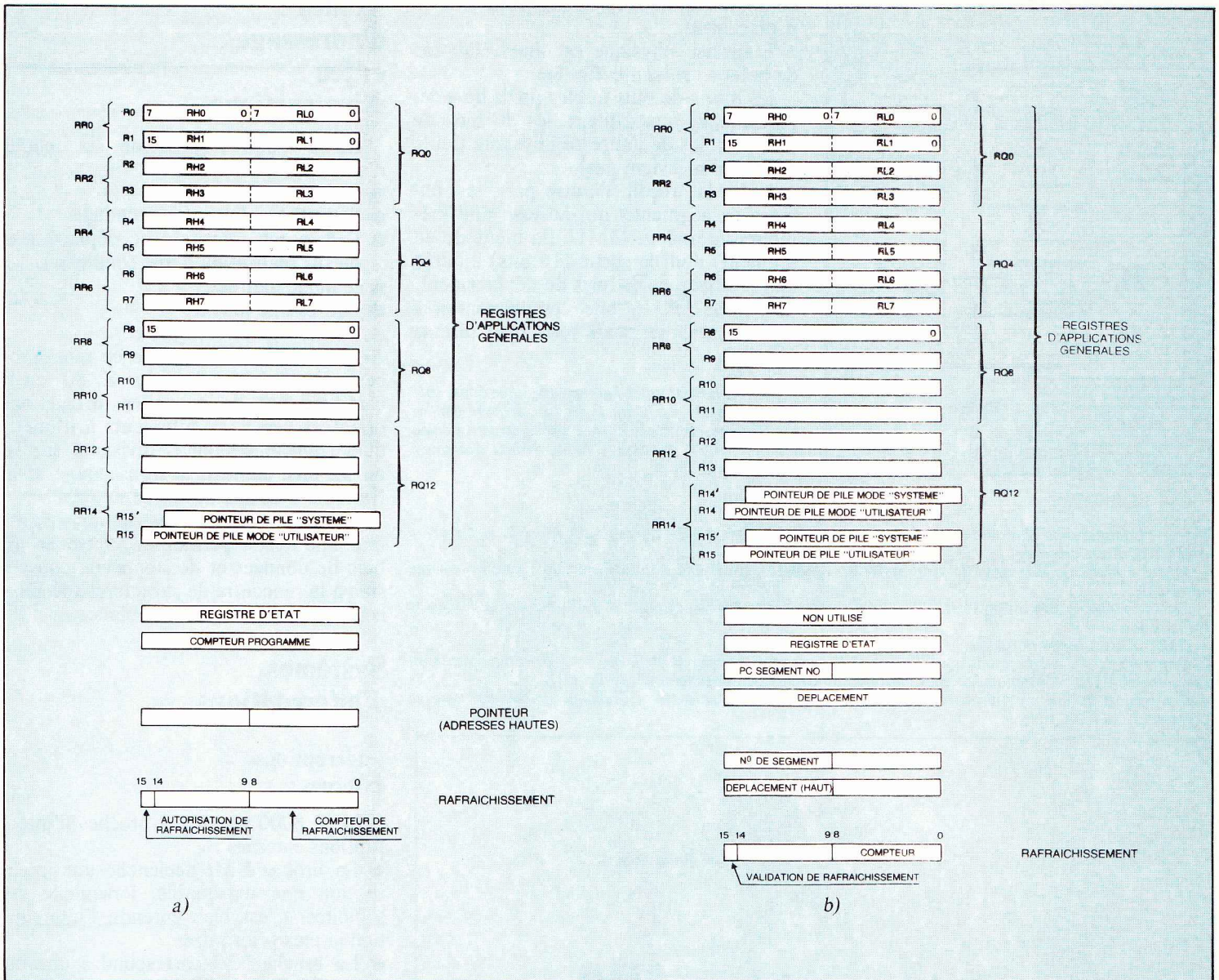


Fig. 3. - Registres internes du Z 8000 : a) version non segmentée ; b) version segmentée.

des mots de 64 bits (il dispose alors de 4 registres généraux).

En plus de ces 16 registres internes, la structure du Z 8000 comporte :

- un registre d'état dont la configuration est représentée **figure 4**,
- un compteur programme,
- un registre pointeur de table d'indirection (TRAP) et d'interruptions,
- un compteur permettant le rafraîchissement de mémoires dynamiques à cadence programmable.

Comme pour le 68000, le Z 8000 peut être utilisé sous deux modes fondamentaux : le mode « NORMAL » (utilisateur) et le mode SUPERVISEUR.

En mode « SUPERVISEUR » toutes les instructions du Z 8000 sont disponibles sans distinction, tandis qu'en mode

« NORMAL » le jeu d'instructions est restreint à celles nécessaires à l'établissement d'un programme utilisateur.

Cette philosophie issue de la mini-informatique assure une haute sécurité de fonctionnement.

En mode superviseur les registres R14 et R15' (Z 8001), ou R15' pour le Z 8002, deviennent des pointeurs de pile gérés par hardware et sont utilisés lors des interruptions.

Organisation de la mémoire

À l'instar du 8086, les bus d'adresses et de données du Z 8000 sont multiplexés dans le temps.

Comme nous l'avons vu, le Z 8001 peut fonctionner en mode « segmenté » et

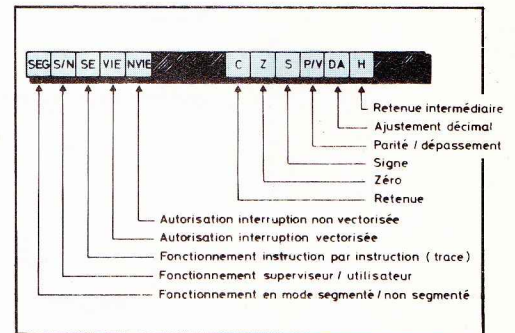
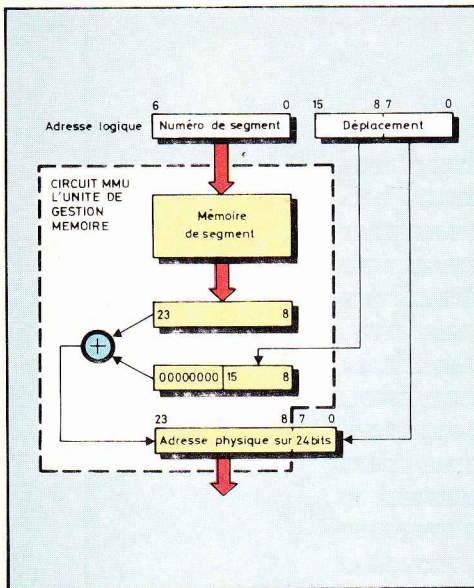


Fig. 4. - Le registre d'état du Z 8000 : 5 bits ne sont pas utilisés.

ainsi adresser 16 M-octets à l'aide de 24 bits d'adresse.

Un circuit spécial (MMU) élabore ces 24 bits d'adresse physique (celle qui sera envoyée sur le bus) à partir de l'adresse logique émise par le microprocesseur qui



comporte un **numéro de segment**, et d'un **déplacement**.

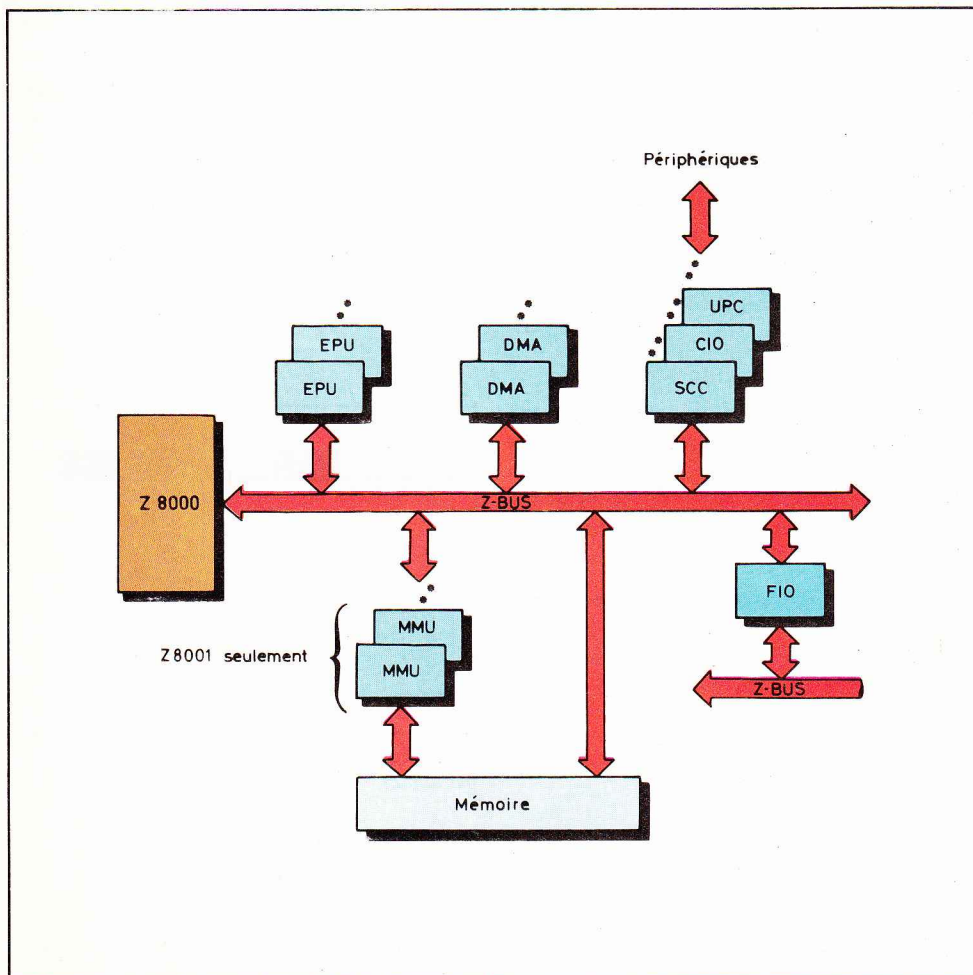
L'adresse physique est alors élaborée de la façon suivante (**fig. 5**) :

- les 8 bits de plus faibles poids du « déplacement » constituent les 8 bits de poids faibles de l'adresse physique (ici le MMU n'intervient pas),
- le MMU reçoit, d'autre part, le « numéro de segment » qui adresse une mémoire, interne au MMU, (la mémoire de segment) dont la sortie (16 bits) ajoutée aux 8 bits de poids forts du déplacement, détermine les 16 bits complémentaires représentant les poids forts de l'adresse physique.

Fig. 5. - Elaboration de l'adresse physique émise sur le bus, à partir de l'adresse logique délivrée par le microprocesseur. Cette adresse physique est calculée par un boîtier spécifique baptisé MMU (Memory Management Unit).

Fig. 6. - Configuration d'un système bâti autour d'un Z 8000. Le microprocesseur génère le « Z-bus » auquel se connectent divers boîtiers :

- MMU : Memory Management Unit (pour le Z 8001) permettant une extension de l'espace mémoire adressable.
- EPU : Extending Processing Unit qui sont des boîtiers permettant d'accroître les performances de l'unité de traitement.
- DMA : Direct Memory Access.
- Divers boîtiers d'interface tels que l'Universal Peripheral Controller (UPC), ou, pour les communications-série, des SCC (Serial Communication Controller), le CIO (Counter-timer and parallel I/O)...
- Des boîtiers de pile d'entrée-sortie (FIFO I/O), baptisés FIO, permettant l'échange de données avec un processeur externe.



Modes d'adressage et jeu d'instructions

Le Z 8000 dispose de six modes d'adressage principaux :

- l'adressage immédiat,
- l'adressage direct (ou étendu),
- l'adressage relatif (avec déplacement ou sur réalisation d'une condition),
- le mode indirect,
- l'adressage indexé,
- l'adressage d'un registre.

Le jeu d'instructions de ce microprocesseur inclut celui du Z 80 qu'il complète par de nouvelles instructions orientées vers les traitements arithmétiques (multiplications et divisions, sur 16 ou 32 bits, signés), le transcodage et le traitement de chaînes de caractères.

A titre d'exemple, l'instruction « translate and test » permet de convertir un bloc de données et de stopper la conversion à la rencontre de caractères spéciaux comme, par exemple, les « blancs ».

Systèmes d'interruptions

Interruptions externes :

Le Z 8000 possède 3 broches d'interruptions externes :

- La broche NMI déclenche une interruption non masquable, lorsqu'elle est sollicitée. C'est bien entendu l'interruption la plus prioritaire.
- La broche NVI correspond à une interruption masquable non vectorisée.
- Enfin, la broche VI permet d'engendrer une interruption masquable vectorisée.

Interruptions internes :

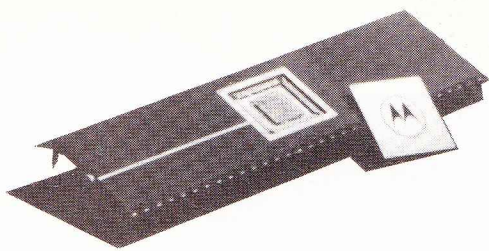
Les traps

Une interruption peut être déclenchée lorsqu'un événement interne ou externe « anormal » a lieu (par exemple l'utilisation en mode « Normal » d'une instruction réservée au mode « Superviseur »).

A chaque événement de ce type, il se produit un déroutement spécialisé baptisé « trap ».

Le traitement de ce déroutement est identique à celui d'une interruption « classique » après sauvegarde du contexte.

Rappelons que les « traps » ont pour rôle de faciliter la tâche du programmeur en lui indiquant ses erreurs. ■



Le 68000

par W. DEBACHE *

Développé par Motorola, le microprocesseur MC 68000 est réalisé en technologie H MOS *.

L'utilisation de cette technologie, dérivée de celle du N MOS jusqu'ici employée, permet de réduire dans un rapport 2 à 3, les dimensions des cellules élémentaires (transistors).

Conséquence directe de cette miniaturisation : le 68000 contient près de 70 000 composants actifs sur une puce de 6 à 7 mm de côté (le 6800 en intégrait environ 5 000).

En outre, grâce à cette réduction de taille, une cellule H MOS fonctionne quatre fois plus vite que son équivalente N MOS et ceci, pour une même consommation.

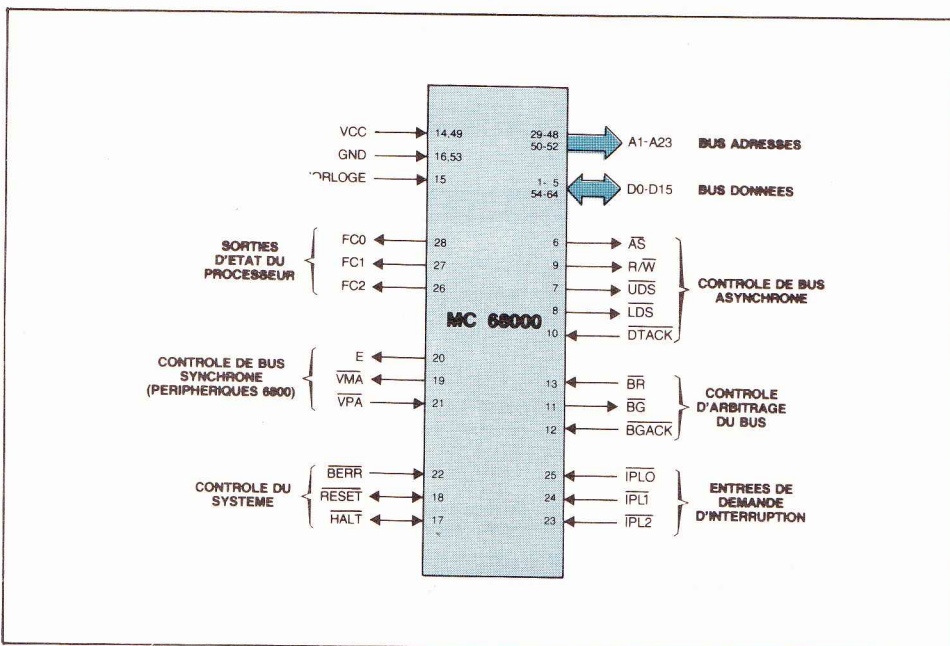


Fig. 1. - Brochage du 68000 : Un boîtier de 64 broches...

Un boîtier de 64 broches...

Le 68000 se présente sous la forme d'un boîtier DIL de 64 broches.

Ce nombre important de connexions extérieures offre l'avantage :

- d'adresser directement une capacité mémoire importante (23 lignes d'adresse soit 8.388.608 mots (8 mégamots) de 16 bits),
- de disposer d'un bus de données de 16 bits physiques,
- de posséder un bus de contrôle très complet, ce qui facilite l'emploi du circuit et offre des ressources multiples lors de la conception de systèmes.

La figure 1 représente le brochage du 68000.

Une architecture interne 32 bits

Si le 68000 dispose physiquement d'un bus de données de 16 bits, il présente, du

Les broches du 68000.

LIGNES ADRESSES

A1-A23 : (Sorties). En conjonction avec les lignes UDS et LDS la capacité d'adressage est de 16 Mégaoctets.

LIGNES DONNEES

DO-D15 : (Bidirectionnelles).

CONTROLE DE BUS ASYNCHRONE

Ces lignes de contrôle permettent au 68000 de communiquer avec des dispositifs asynchrones. En particulier l'utilisation de mémoires lentes, donc peu coûteuses, est rendue possible grâce à ces lignes. On accède aux octets par les sorties UDS et LDS.

AS : (sortie) Address Strobe. Indique la validation de l'adresse.

R/W : (Sortie). Read/Write. Ecriture ou lecture.

UDS-LDS : (Sorties) Upper Data Strobe - Lower Data Strobe. Identifient l'octet haut ou l'octet bas des données de 16 bits.

DTACK : (Entrée) Data Transfer Acknowledge. Signal émis par un périphérique ou une mémoire pour valider le transfert des données.

CONTROLE D'ARBITRAGE DU BUS

Ces trois lignes simplifient la conception matérielle d'un système multiprocesseurs ou les procédures DMA.

BR : (Entrée) Bus Request. Indique au processeur qu'un dispositif externe réclame le bus.

BG : (Sortie) Bus Grant. Fournit l'accord du 68000 pour qu'un dispositif externe prenne possession du bus.

BGACK : (Entrées) Bus Grant Acknowledge. Confirme au 68000 que le dispositif externe est possesseur du bus.

ENTREES DE DEMANDE D'INTERRUPTION

IPL0, IPL1, IPL2 : (Entrées) Interrupt Priority Level. Procurent au processeur le niveau de priorité d'une demande d'interruption.

CONTROLE DU SYSTEME

BERR : (Entrée) Bus Error. Déclenche le Trap « Bus Error » dans le cas d'une anomalie hardware.

RESET : (Bidirectionnel).

En entrée, permet l'initialisation du 68000 et de son environnement.

En sortie, résulte de l'instruction RESET pour initialiser exclusivement l'environnement.

HALT : (Bidirectionnel).

En entrée, arrête le processeur à la fin du cycle en cours.

En sortie, indique que le processeur est arrêté (dans le cas d'une double faute par exemple).

CONTROLE DE BUS SYNCHRONE

Ces lignes permettent l'emploi de dispositifs synchrones et en particulier celui des périphériques 6800.

E : (Sortie) Enable. Synchronisation de l'organe extérieur.

VPA : (Entrée) Valid Peripheral Address. Indique au 68000 qu'il communique avec un périphérique synchrone (type 6800 par exemple).

VMA : (Sortie) Valid Memory Address. Indique la validité de l'adresse sur le bus, en fonctionnement synchrone.

SORTIES D'ETAT DU PROCESSEUR

FC0, FC1, FC2 : (Sorties) Function Code. Communiquent à l'extérieur l'état dans lequel se trouve le 68000. En particulier, elles indiquent si le processeur fonctionne en mode superviseur ou utilisateur et si l'information en traitement est du type donnée ou programme. Ces sorties offrent une sécurité supplémentaire pour le système et par leur décodage permettent de quadrupler la capacité d'adressage.

ALIMENTATION ET HORLOGE

VCC : (2 broches) : 5 V.

GND (2 broches) : 0 V.

Horloge : 8 MHz (TTL)

point de vue interne, les caractéristiques d'un microprocesseur 32 bits.

Comme le montre la **figure 2**, il possède en effet 15 registres de travail, et deux registres « pointeurs de pile ». Chacun de ces 17 registres a une taille de 32 bits, ce qui confère au 68000 une « capacité interne » très importante.

Le registre d'état se compose de 2 octets : le premier réservé au mode **superviseur** et l'autre, à la disposition de **l'utilisateur**.

6 bits sont encore inexploités dans le registre d'état, ce qui permettra certainement une extension de ce microprocesseur dans le futur.

D'autre part, le compteur ordinal n'utilise que 24 bits (16 méga-octets adressables) alors qu'il s'agit d'un registre de 32 bits.

Mode « superviseur » et mode « utilisateur »

Le 68000 peut fonctionner sous deux modes appelés « **superviseur** » et « **utilisateur** ».

En mode superviseur, toutes les fonctions (et en particulier toutes les instructions) sont accessibles, alors qu'en mode utilisateur toutes les possibilités du processeur ne peuvent pas être mises en œuvre.

Par exemple, l'arrêt du processeur issu de l'instruction STOP, n'est pas accessible en mode utilisateur, c'est-à-dire au programme d'application.

Cette notion de deux niveaux de service hiérarchisés de la machine, qui existaient déjà sur certains « minis », confère au système une sécurité et un haut niveau d'exploitation encore jamais atteints sur les microprocesseurs de l'ancienne génération.

Les interruptions

Aux trois entrées de demandes d'interruption **IPL0**, **IPL1** et **IPL2** correspondent 8 niveaux d'interruption avec priorité.

L'utilisateur peut masquer une demande d'interruption jusqu'à un certain niveau grâce aux 3 bits **I₀**, **I₁** et **I₂** du registre d'état. (**fig. 3**).

La priorité se fait dans le sens ascendant : le niveau 7 correspond à une interruption non masquable et le niveau 0 représente le fonctionnement « normal » du système (pas de demande d'interruption).

Pour comprendre le mécanisme de prise en compte de ces interruptions, supposons que les bits **I₂**, **I₁** et **I₀** soient respectivement aux états logiques 1, 0 et

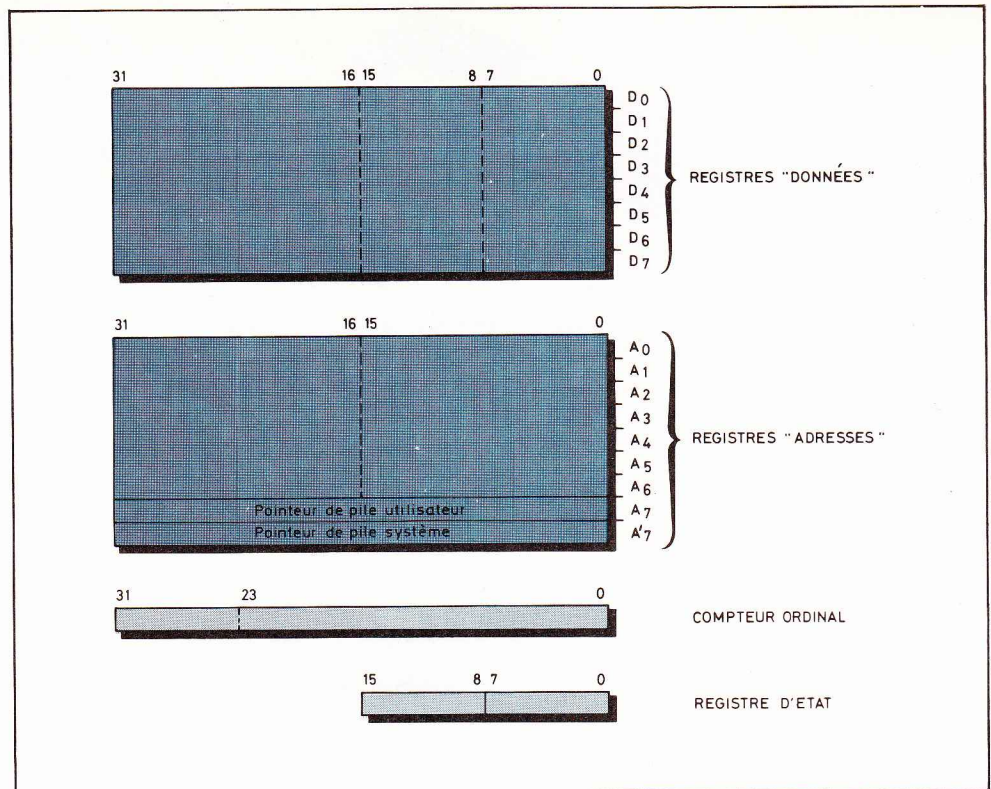
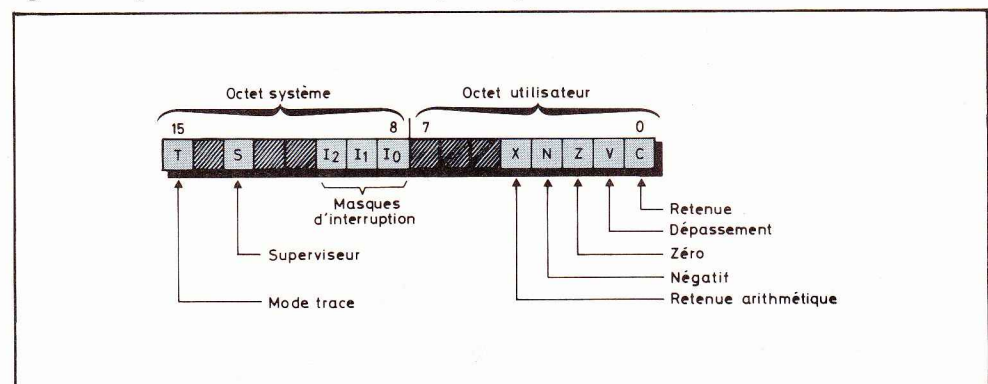


Fig. 2. - Les registres internes du 68000.

Fig. 3. - Le registre d'état : 6 bits sont encore disponibles pour les extensions futures.



0, ce qui correspond à la valeur décimale 4. Ainsi :

- une demande d'interruption de niveau inférieur ou égal sera ignorée,
- par contre, une demande d'interruption de niveau 5, 6 ou 7 sera exécutée.

199 « vecteurs » contenant l'adresse de début des programmes d'interruption peuvent être associés à ces 7 niveaux d'interruption disponibles (le niveau 0 correspond à un fonctionnement normal).

192 de ces vecteurs sont utilisables pour des périphériques capables de fournir leurs propres numéros de vecteurs.

Pour les autres, tels que périphériques 6800 (PIA, ACIA, etc.) qui ne disposent pas de la logique nécessaire, le 68000 fera référence au niveau de l'interruption

demandée pour sélectionner le vecteur parmi les 7 restants.

On parle alors d'**autovectorisation**.

Cette puissante structure d'interruption reflète les extraordinaires progrès réalisés en quelques années : 199 interruptions possibles pour le 68000 contre les 2 (NMI et IRQ) du 6800.

Les « traps »

Ce concept nouveau confirme la tendance logicielle du 68000 et l'importance que l'on accorde à la **sécurité** du système.

* Ingénieur ESE, Willy DEBACHE appartient au département « Marketing MPU » de la firme Motorola (France).

* High Density Metal Oxyde Semiconductor (voir Micro-Systèmes n° 3, p. 51 : « La Technologie H MOS »).

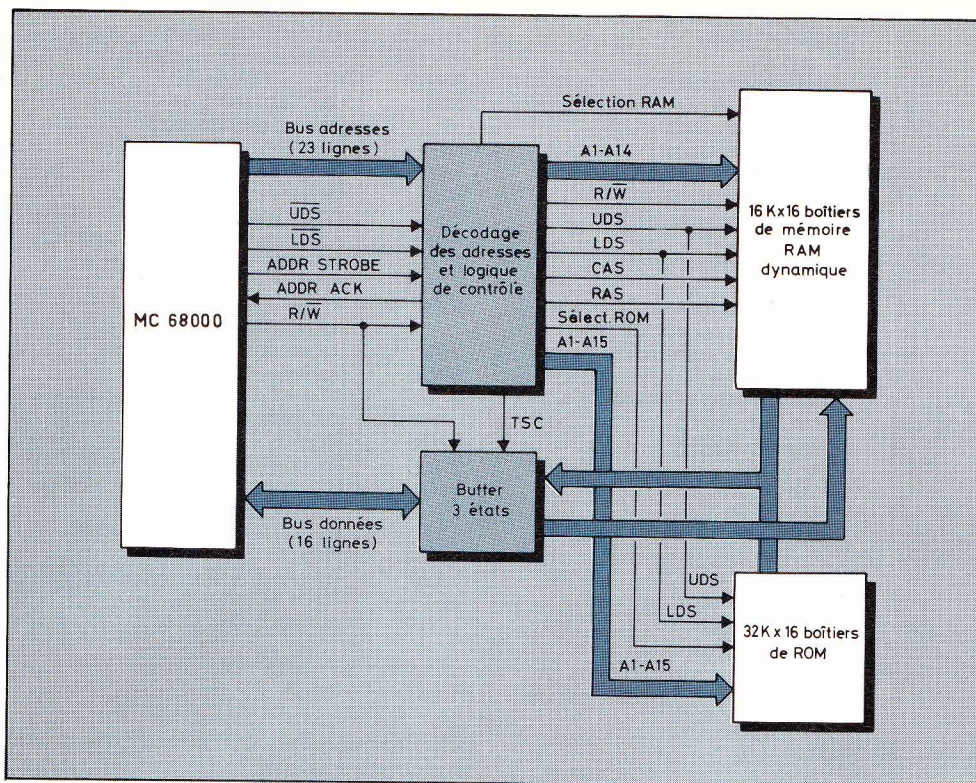


Fig. 4. - Système « minimum » bâti autour du 68000 comprenant une RAM de 16 K-mots et une ROM de 32 K-mots. On distingue ici les signaux nécessaires aux boîtiers de mémoires RAM dynamique :

- UDS (Upper Data Strobe) : Validation de l'octet de poids fort des données.
- LDS (Lower Data Strobe) : Validation de l'octet de poids faible des données.
- CAS : Validation de l'adresse « colonne ».
- RAS : Validation de l'adresse « ligne ».
- R/W : Lecture/Ecriture.

A chaque Trap ou « piège », est associé un vecteur qui contient une adresse de début d'un programme.

Un « Trap » est un déroutement du programme principal résultant le plus souvent d'une anomalie (hard ou soft).

La division par zéro est un exemple de Trap. En effet, lorsqu'une telle anomalie se produit, le 68000 va chercher l'adresse de début du programme « division par zéro », contenue dans le vecteur associé.

C'est le concepteur qui écrit le programme de déroutement à travers lequel il pourra, dans notre exemple, soit recalculer le « zéro » avec plus de précision, soit envoyer un message d'erreur et continuer, soit encore arrêter le processeur.

Le 68000 dispose également d'autres Traps. Citons, la détection d'une instruction illégale, une violation de privilège (emploi de ressources superviseur dans le mode utilisateur), une erreur de bus (activation de la ligne BERR, Bus Erreur)...

D'autres « pièges » sont déclenchés directement à partir d'instructions telles que « CHECK », qui permet de vérifier si une variable est comprise entre deux valeurs. Un dépassement d'encadrement provoque le Trap « CHECK ».

Il existe également 16 instructions

Trap (notées Trap # 0, Trap # 1, etc.) qui permettent de dérouter volontairement, à partir du logiciel, le programme en cours.

Le mode « Trace »

Le 68000 est doté du mode « Trace » ou « pas à pas », ce qui jusqu'à présent, n'était pas possible avec les microprocesseurs 8 bits.

Après chaque instruction, le microprocesseur teste, de manière interne le bit « T » du registre d'état.

Si ce bit est à zéro, il passe à l'instruction suivante.

Par contre, s'il est à l'état « 1 », il exécute un programme démarré à l'endroit indiqué par le vecteur TRACE. De plus, en conjonction avec la ligne HALT l'exécution « cycle par cycle » est rendue possible.

Le mode Trace offre par exemple de larges facilités pour la conception d'un moniteur de mise au point.

Les modes d'adressage

Le 68000 dispose de nombreuses possibilités d'adressage regroupées en cinq modes principaux :

- adressage direct aux registres,
- adressage absolu,
- adressage relatif au compteur ordinal,
- adressage indirect,
- adressage immédiat.

Les registres adresse ou donnée peuvent éventuellement servir d'index avec un déplacement variable.

Les post incrémentation et prédécrémentation permettent d'utiliser les registres adresse comme de véritables pointeurs de pile.

Grâce à l'adressage « relatif au compteur ordinal » le code objet * 68000 peut être exécuté indépendamment de sa position physique en mémoire.

Cette « auto-relogeabilité » du code permet d'écrire des logiciels sur ROM exécutables sur n'importe quel système à base du 68000, quelle qu'en soit la configuration matérielle.

Jeu d'instructions

Motorola a volontairement limité à une soixantaine les instructions du 68000 afin de minimiser l'effort de l'utilisateur (tableau I).

L'instruction « Test and Set » (TAS) facilite l'implémentation de sémaphores nécessaires pour établir des exclusions mutuelles, utiles par exemple dans le cas d'une configuration multiprocesseurs, lorsque deux processeurs demandent **simultanément** l'accès à un bus.

LINK et UNLINK sont des instructions de haut niveau qui permettent d'allouer de manière automatique une zone mémoire. Elles simplifient la gestion dynamique de mémoire (par exemple pour les structures de blocs en PASCAL) ou encore la passation de paramètres entre deux programmes.

Il est à noter que les résultats d'opérations peuvent directement être rangés dans des cases mémoires.

De même les opérandes sont accessibles sans intermédiaires à partir de la mémoire.

A titre d'exemple, l'instruction MOVE transfère des données de registre à registre, de registre à mémoire (ou vice-versa) et surtout de mémoire à mémoire.

L'orthogonalité

Le 68000 présente une structure orthogonale, c'est-à-dire que pour la quasi totalité des cas on peut associer à n'importe laquelle de ses instructions le mode

* Code objet : C'est le code directement exécutable par le microprocesseur.

d'adressage et le format des données (8, 16 ou 32 bits) que l'on désire.

Cette structure régulière accroît l'efficacité du système et simplifie le travail du programmeur.

Diminution du coût des logiciels

L'ensemble des caractéristiques du 68000 (jeu d'instructions, architecture, modes d'adressage, etc.) lui permet de supporter des techniques de programmation modernes. En plus de la possibilité de générer un code exécutable indépendamment de sa position en mémoire, il permet l'écriture de programmes modulaires, récurifs* ou réentrants*.

A titre d'exemple, l'EXORMACS, outil de développement du 68000 dispose d'un compilateur PASCAL.

La famille 68000

En plus des périphériques 6800 qui lui sont directement compatibles, le 68000 disposera prochainement de ses propres périphériques.

Parmi les plus importants citons :

- le contrôleur universel de périphériques (IPC),
- l'unité de gestion mémoire (MMU),
- le contrôleur et correcteur d'erreurs mémoire (E.C.C.),
- le contrôleur DMA (DMAC).

Mnémonique	Description	Mnémonique	Description
ABCD	Addition décimale	MOVE	Transferts
ADD	Addition binaire *	MULS	Multiplication signée
AND	Et logique	MULU	Multiplication non signée
ASL	Décalage arithmétique vers la gauche	NBCD	Négation décimale
ASR	Décalage arithmétique vers la droite	NEG	Complément à deux (avec ou sans retenue)
BCC	Branchement conditionnel (parmi 16 conditions)	NOP	Pas d'opération
BCHG	Test et changement d'un bit	NOT	Complément logique
BCLR	Test et mise à zéro d'un bit	OR	Ou logique
BRA	Branchement inconditionnel	PEA	Déchargement de l'adresse effective
BSET	Test et mise à 1 d'un bit	RESET	Réinitialisation des circuits externes
BSR	Branchement sous-programme	ROTL	Rotation vers la gauche *
BTST	Test d'un bit	ROTR	Rotation vers la droite *
CHK	Vérifie l'encadrement d'un registre	RTE	Retour d'une exception (interruption ou trap)
CLR	RAZ d'un opérande	RTS	Retour de sous-programme
CMP	Comparaison arithmétique	SBCD	Soustraction décimale
DBCC	Test d'une condition (parmi 16), décrémentation et branchement	SCC	Mise à 1 d'un opérande suivant une condition parmi 16
DIVS	Division signée	STOP	Stop
DIVU	Division non signée	SUB	Soustraction binaire *
EOR	Ou exclusif	SWAP	Echange partie haute et partie basse d'un registre
EXG	Echange de registres	TAS	Test et positionne à 1 l'opérande
EXT	Extension de signe	TRAP	Trap (16 disponibles)
JMP	Saut	TRAPV	Trap sur overflow (dépassement)
JSR	Saut à un sous-programme	TST	Test d'un opérande
LEA	Chargement de l'adresse effective	UNLK	Dual de LINK
LINK	Allocation d'une zone mémoire		
LSL	Décalage logique vers la gauche		
LSR	Décalage logique vers la droite		

Tableau 1. - Résumé du jeu d'instructions du 68000. * Avec et sans retenues.

Conclusion

Les prévisions technologiques révèlent que le cap du million de composants actifs sur la même puce sera vraisemblablement franchi dans moins d'une décennie.

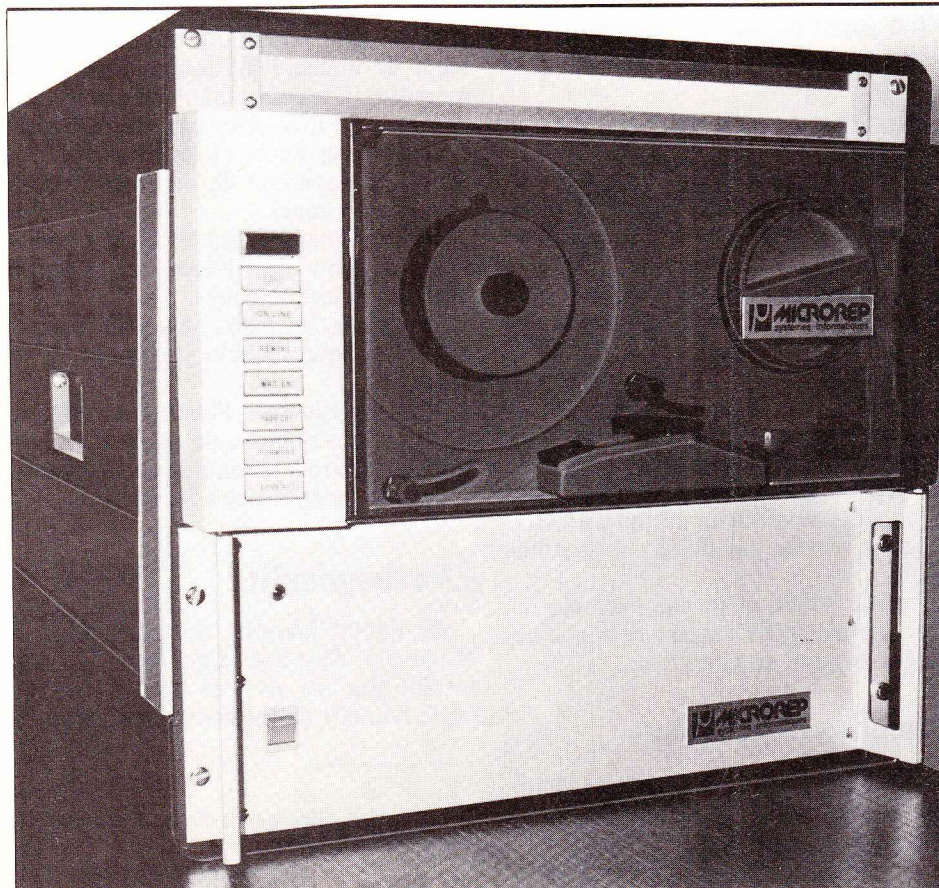
Mais quels circuits pourront alors exploiter pleinement cette extraordinaire intégration tout en conservant un caracté-

rière suffisamment universel en vue d'une production massive ?

De nombreuses idées répondent déjà à cette interrogation, preuve que si la matière a des limites, l'imagination de l'homme est infinie. ■

* Récurif: Programme qui peut s'appeler lui-même.

* Réentrant: Un programme qui peut être logé à n'importe quel endroit de la mémoire.



UN DÉROULEUR DE BANDE COMPATIBLE 800/1600 BPI

connectable à un micro ou à un système informatique disposant d'une sortie V24 ou parallèle.

MICROREP
systèmes informatiques

24, boulevard Anatole-France, 92190 Meudon.
tél. : 534.76.47.

La voiture-robot du Microtel-Club de Bordeaux

Vainqueur du 1^{er} Championnat international de Voitures-Robots

par le Microtel-Club de Bordeaux

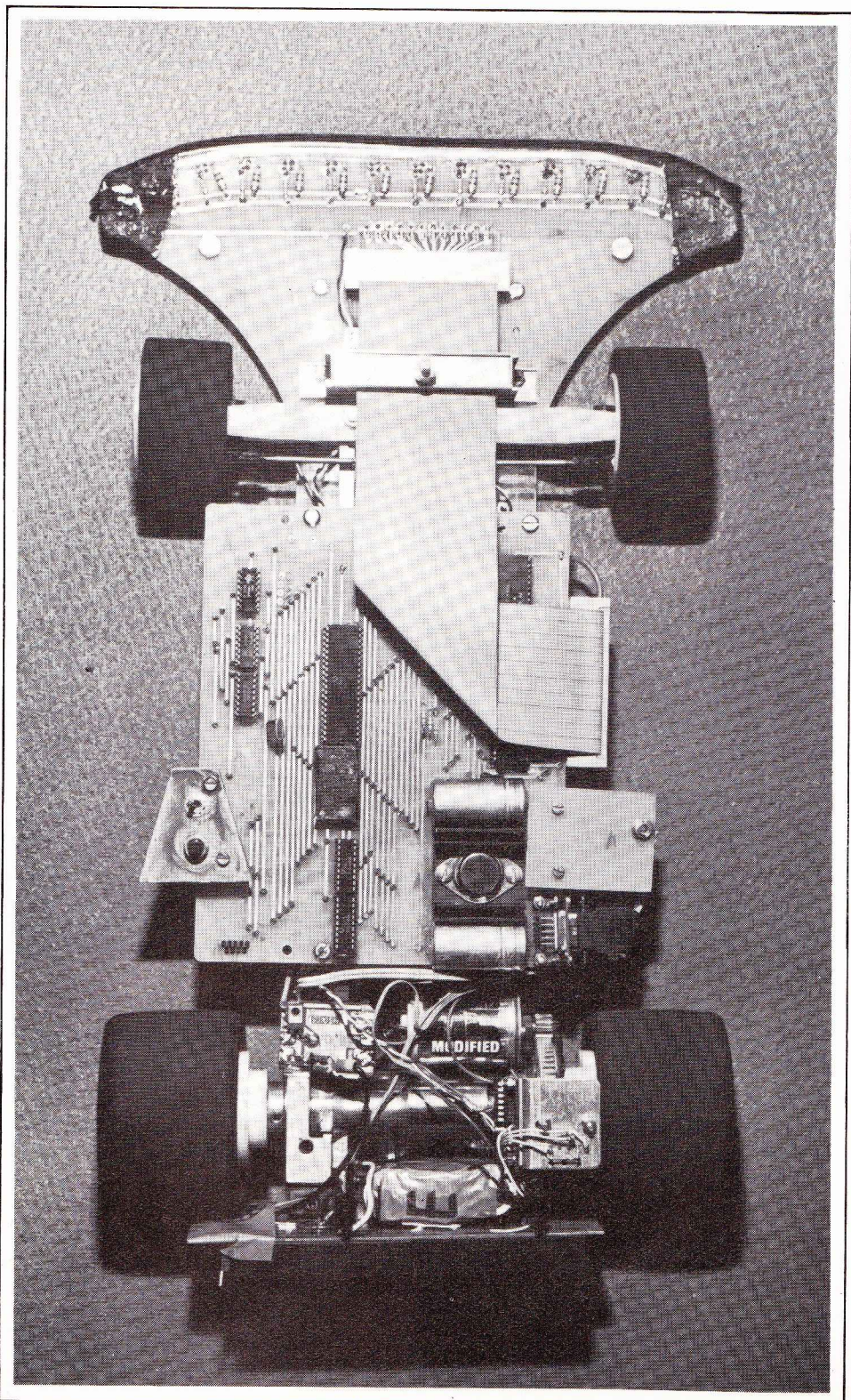


Photo 1. – La voiture du Microtel Club de Bordeaux sans sa carrosserie : une très belle réalisation.
(Photo SYGMA.)

Lorsque le Microtel Club de Bordeaux décida, en janvier 1980, de participer au premier championnat international de voitures-robots, l'équipe de volontaires qui s'est constituée était persuadée que huit mois ne seraient pas de trop pour mener à bien cette opération.

Rappelons la composition de cette équipe :

- Gaëtan Luong et Noël Monin chargés de l'électronique.
- Francis Dubois et Jean-Philippe Pujol chargés de l'informatique.
- Christian Banlier et Christian Cournut chargés de la mécanique auxquels se sont joints dans les moments les plus difficiles Jean-Jacques Massey représentant la société Composants S.A. qui fournissait les composants et Francis Gonzalez, informaticien, bien connu pour ces grandes performances sur 1 500 et 5 000 mètres.

Aucun de ces téméraires n'était familiarisé avec la robotique et réaliser une telle voiture était une aventure, un voyage dans l'inconnu...

La rédaction de Micro-Systèmes tient à féliciter particulièrement la sympathique équipe de ce club pour leur magnifique prestation qui souleva l'enthousiasme du public.

Nous voulons aussi remercier la société Texas Instruments, en la personne de Stéphane Chapotin, qui offre le premier prix de ce championnat : un ensemble informatique composé autour de l'ordinateur familial TI 99/4 d'une valeur de 15 000 F.

Une stratégie de réalisation

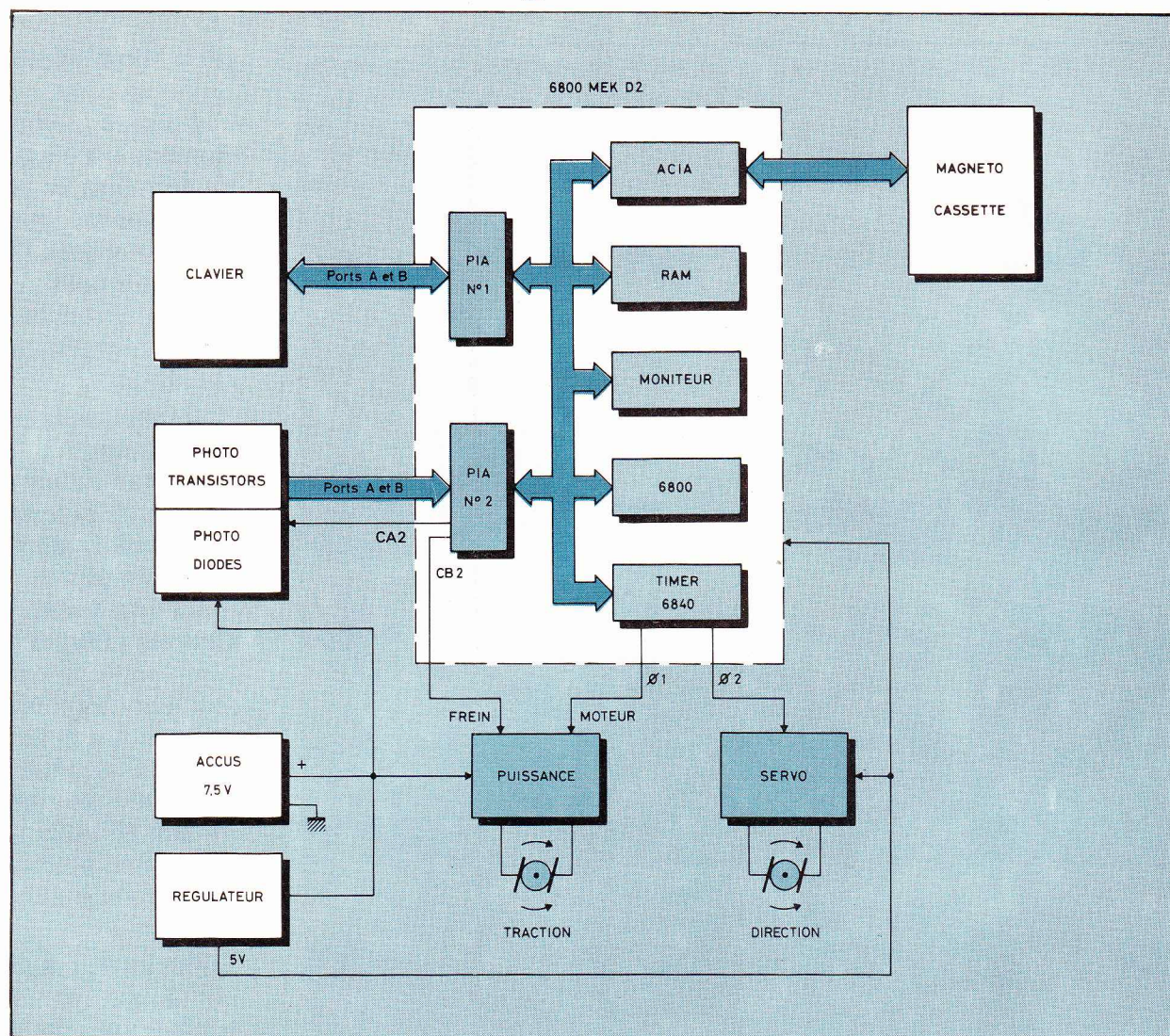


Fig. 1. – Les éléments du prototype équipé d'un kit 6800 MKD2.

Les adhérents du Microtel Club sont des lecteurs de Micro-Systèmes et le prototype présenté par la revue a fortement inspiré l'équipe qui a tout de suite adopté les mêmes capteurs et un châssis similaire. Nous avons voulu simplifier au maximum le travail de construction mécanique pour nous consacrer essentiellement à l'électronique et à l'informatique, activités de base du Microtel Club. C'est pourquoi le choix s'est porté sur un châssis au 1/8^e d'un modèle radiocommandé du commerce, relativement performant.

Le prototype

Afin d'acquérir l'expérience qui nous manquait dans ce domaine, nous devons faire un certain nombre de tests avec un prototype de fortune sur un circuit de fortune avant de construire le modèle définitif, ne serait-ce que pour vérifier

le bien-fondé de nos options. On a donc installé un Kit MEK 6800 D2 (fixé avec des élastiques) sur un châssis au 1/8^e équipé d'un moteur de traction et d'un servomoteur de direction. Par ailleurs nous avons construit une mini piste découpée dans du papier goudronné sur laquelle des pointil-

lés blancs de scotch adhésif plastique furent collés. Cette piste a été installée dans le local du club.

Il a fallu évidemment adapter le kit en l'équipant d'un Timer 6840 et en doublant la taille mémoire RAM. La figure 1 représente le système électronique du prototype.

Le prototype, avec ses énormes radiateurs et son clavier ressemblait plutôt à un attelage agricole.

Voitures-robots

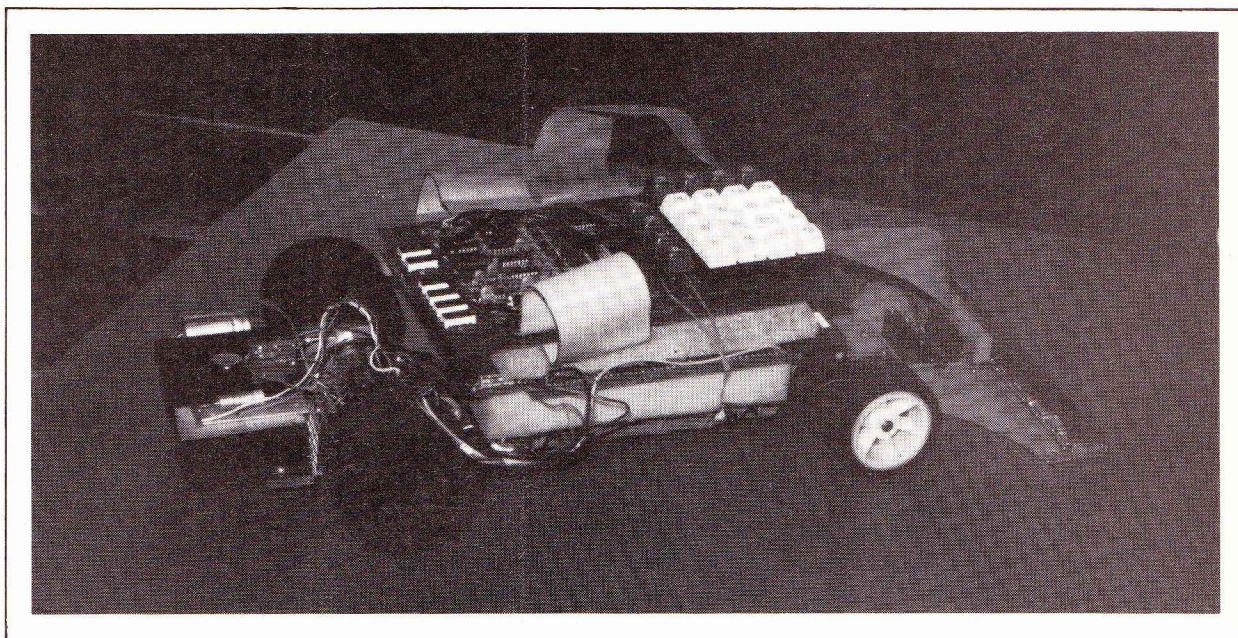
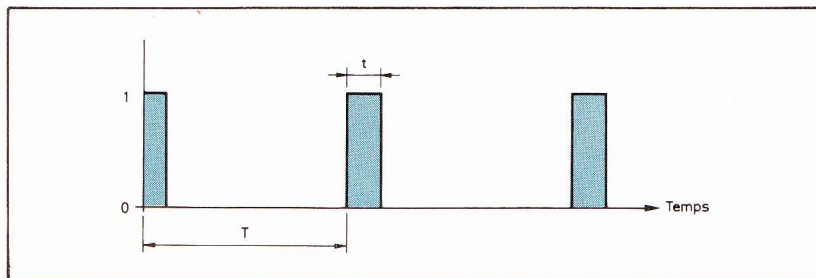


Photo 2. – Le prototype conçu autour du kit MKD2. Le clavier permettait de modifier en temps réel les paramètres de programmation.

Fig. 2. – Les créneaux de commande de direction, de traction, d'allumage des photodiodes et de frein :

Direction :	$T = 10 \text{ ms}$	$t = 1 \text{ à } 2 \text{ ms}$	Sortie $\varnothing 1$ du timer
Traction :	$T = 10 \text{ ms}$	$t = 0 \text{ à } 10 \text{ ms}$	Sortie $\varnothing 2$ du timer
Photodiodes :	$T = 2 \text{ à } 10 \text{ ms}$	$t = 2 \text{ ms}$	Sortie CA2 du PIA
Frein :	Tout ou rien		Sortie CB2 du PIA



Le Timer 6840, semblable à celui utilisé pour la voiture de Micro-Systèmes devait assurer les commandes de direction et de traction selon une méthode comparable à celle décrite dans la revue.

Ce prototype, avec son clavier MEK D2, sur mousse plastique, et d'énormes radiateurs pour les transistors de puissance, ressemblait plutôt à un attelage agricole. Mais nous lui devons tout. Il nous a fait découvrir les pièges à éviter et « la voie royale ».

Le MEK D2 comprend en version de base deux PIA, un pour la gestion du clavier et un pour des entrées sorties de service. Un

timer 6840 a d'abord été ajouté dans la partie extension (à câbler) du Kit. Le deuxième PIA a été relié directement aux détecteurs photo sensibles et le Timer aux commandes de mouvement de la voiture : un servo-moteur du commerce pour la direction, et un moteur de traction piloté par une électronique de puissance.

Nous n'insisterons pas sur ces systèmes de commande car ils ont déjà été décrits ici. Précisons simplement que des créneaux de longueur variable sont délivrés par le Timer 6840 sur les sorties $\varnothing 1$ et $\varnothing 2$ et une variation de la longueur de ces créneaux correspond à une variation soit de la vitesse du moteur de traction (pour $\varnothing 2$) soit de

l'angle de braquage (pour $\varnothing 1$). La fréquence de ces créneaux est de l'ordre de 100 périodes par seconde (fig. 2).

La mise au point

Le Kit a aussi l'avantage de posséder un interface cassette. La procédure de mise au point était donc la suivante :

Les programmes étaient écrits en langage assembleur sur un micro-ordinateur Goupil puis stockés sur magnétocassette. (Un interface cassette pour le standard Kansas City a été spécialement développé). La cassette permettait de réintroduire les programmes dans la mémoire du MEK D2. Les procédures étaient lancées à l'aide du clavier. Ce clavier permettait aussi de modifier en temps réel les paramètres de programmation pendant les évolutions du véhicule. Il fallait évidemment un peu de dextérité pour le suivre à la trace surtout lorsqu'il quittait la piste (c'est là qu'intervenait notre champion de course à pied !).

Ces essais ont ainsi permis pendant six mois d'affiner un logiciel de plus en plus performant et de définir un système électronique réduit à sa plus simple expression.

Le logiciel

Le logiciel a été conçu en plusieurs étapes. D'abord l'acquisition des données lors du premier tour de reconnaissance * : il fallait régler là un problème de cybernétique avec un asservissement de système à grande inertie. Ensuite le traitement informatique de ces données de manière à obtenir des commandes précises et infaillibles sur une trajectoire optimale. Enfin l'exécution de ces commandes à la plus grande vitesse possible avec là encore des problèmes d'inertie considérables.

L'acquisition

Le premier problème à résoudre concernait le positionnement de la voiture par rapport à la piste. Plusieurs solutions ont été envisagées :

1° Utiliser les lignes blanches continues. Cette solution avait l'avantage de placer la voiture bien à l'intérieur ou bien à l'extérieur des virages pour une conduite optimale. Mais dans ce cas, la voiture ne pouvait pas se situer par rapport à son point de départ. Pour cela il était possible de compter les tours de roue et la combinaison de ces deux méthodes semblait a priori satisfaisante.

2° Asservir la voiture au pointillé. Cette méthode semblait plus précise et elle permettait aussi de serrer à la corde dans la limite de la largeur de la voiture. Mais il était alors absolument interdit « d'oublier » de compter un pointillé. Les dérapages étaient exclus. Comme on le verra, toutes ces solutions ont été testées et les premières s'étant avérées aux essais trop imprécises, c'est la dernière qui fut adoptée.

Il fut donc décidé d'associer une « case » mémoire à chaque pointillé. L'adresse de cette case représenterait le numéro du pointillé et le contenu restait à définir.

Il fallait donc « encadrer » le pointillé à l'aide de photodiodes. L'utilisation de deux ou même quatre photodiodes telle que le faisait Micro-Systèmes nous est apparue à l'expérience, insuffisante. D'autant plus qu'avec la solution adoptée (ne pas perdre de vue le pointillé), si l'on voulait serrer sur les bords extérieurs du véhicule.

Ainsi, pour obtenir la plus grande précision possible dans le

positionnement de la voiture par rapport au pointillé, il a été décidé de placer une rangée de quinze diodes sur toute la largeur de la voiture (30 cm) le plus en avant possible de manière à anticiper au maximum la lecture (ceci en raison de l'inertie du système).

Il suffisait alors d'éclairer périodiquement les diodes et le résultat de la réflexion, blanc ou noir, fournissait un « zéro » ou un « un » à l'entrée correspondante du PIA. La commande d'allumage était un ordre microprocesseur fourni sur la sortie CA2 du PIA.

Le pointillé était ainsi parfaitement « placé » sous la voiture. Les diodes étant espacées de 2 cm, pour un pointillé de 4 cm, une ou deux diodes pouvaient se trouver au-dessus du pointillé et la voiture pouvait ainsi occuper vingt-neuf positions possibles sur la largeur de la piste.

L'image du circuit

Nous avons admis le principe suivant :

La longueur du créneau de commande est proportionnelle à la courbure (inverse du rayon de courbure) elle-même proportionnelle à l'angle de braquage. Les calculs ont montré qu'il s'agit là d'une approximation suffisante pour cette application. En conséquence, il suffisait de quantifier ce créneau de commande sur un octet (256 valeurs) et de relever cette valeur dans l'octet réservé à chaque pointillé. On obtenait ainsi une image fidèle de la courbure du circuit, à condition que les valeurs des angles de braquage soient relevées avec précision. En fait, nous nous sommes contentés de 128 positions de braquage.

Pour obtenir la plus grande précision possible, trois méthodes ont été employées : d'abord serrer de très près le pointillé de manière à avoir des écarts minima. C'était possible au premier tour à très faible vitesse. Ensuite, relever, sur un même pointillé, le plus de valeurs possibles et en faire la moyenne. Ainsi les diodes étaient allumées et le résultat lu toutes les 10 milli-secondes ce qui correspondait, pour une vitesse de 2 km/heure, à 25 lectures environ par pointillé.

Enfin, la troisième méthode, pour augmenter la précision, était d'effectuer au cours du traitement des données mémorisées, un lissage de ces valeurs. Ainsi, à la fin du premier tour la suite des valeurs enregistrées dans chaque octet de la mémoire, c'est-à-dire pour chaque pointillé, épousait bien la forme du circuit.

Le traitement

Comme on l'a vu, la première partie du traitement était un lissage des valeurs relevées en mémoire. Ce lissage fournissait une trajectoire qui centrait la voiture sur le pointillé. Mais cette trajectoire n'était pas la plus optimale afin d'obtenir des performances convenables. Il fallait donc encore modifier judicieusement les valeurs en mémoire pour que la voiture se situe à l'extérieur à l'entrée du virage, à l'intérieur au milieu du virage et de nouveau à l'extérieur à la sortie du virage. Cette opération, associée au lissage, consistait à placer à gauche ou à droite la diode dite « de référence », c'est-à-dire celle centrée sur le pointillé.

Grâce à un capteur photosensible placé à cheval sur une roue dentée solidaire du différentiel et

* Rappelons à ce sujet que la course se déroulait sur 3 tours : 1 tour de reconnaissance et 2 tours chronométrés.

A chaque pointillé de la piste correspondait une case mémoire dont l'adresse représentait le numéro du pointillé.

Voitures-robots

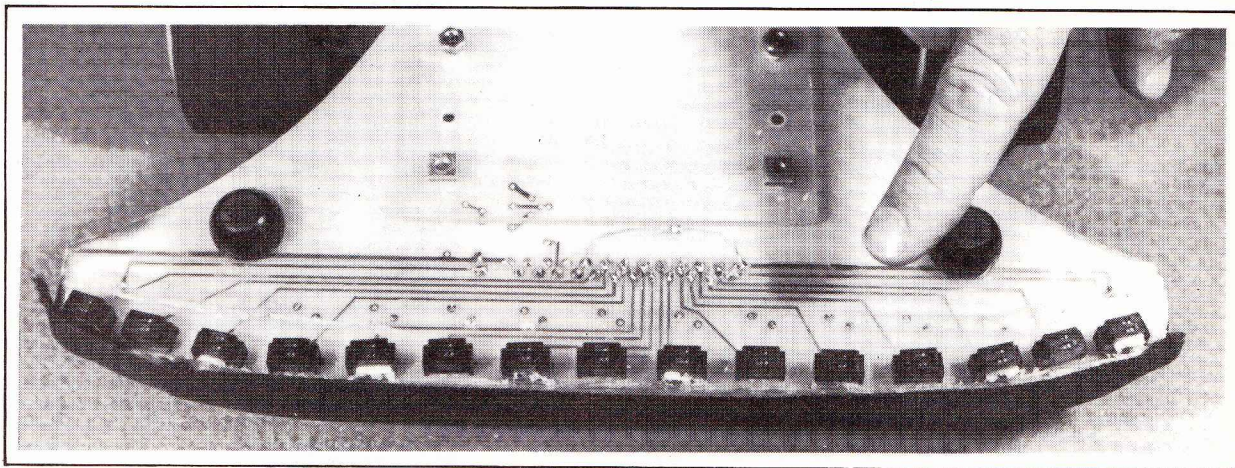


Photo 3. – Les 15 photodétecteurs placés à l'avant de la voiture sur un châssis en plexiglas.

une électronique de mise en forme très simple la vitesse du véhicule était lue à chaque instant. On utilisait pour cela l'entrée des interruptions prioritaires du 6800 « l'entrée NMI ».

Ainsi le microprocesseur pouvait ajuster la valeur du créneau de traction pour réguler une vitesse imposée. Et cette vitesse était maintenue quel que soit l'état de la piste, courbure ou déclivité.

Si l'on fait dépendre la vitesse de la courbure du circuit, il apparaît que les variations plus ou moins grandes de courbure provoquent des accélérations ou des décélérations plus ou moins brutales. Il fallait donc associer, à l'aide d'une table, une valeur de courbure (c'est-à-dire une valeur de créneau de direction) à une valeur de vitesse. La fonction courbure-vitesse ne nous a pas paru linéaire, c'est pourquoi, d'une part, cette table a été composée par l'expérience, d'autre part plusieurs tables ont été écrites dans des EPROM différentes pour s'adapter à des circuits de difficultés différentes.

Pour passer d'une vitesse à une autre, la voiture qui possède une grande inertie, met un certain temps. Un temps d'autant plus long que la variation de vitesse est importante. Il convenait donc **d'anticiper** la commande de vitesse. Et **l'anticipation devait dépendre de la grandeur de la variation de vitesse.**

Par exemple, si le véhicule trouvait au bout d'une longue ligne droite un virage serré et que sa vitesse devait passer de 25 km/heure en ligne droite à 6 km/heure dans ce virage, la commande de vitesse 6 km/heure était lancée 10 pointillés avant l'entrée du virage. Mais si la courbure du virage permettait une vitesse de 12 km/heure, la commande de ralentissement à 12 km/heure était lancée 6 pointillés avant l'entrée de ce virage.

L'exécution

Après tout ces calculs en mémoire, d'ailleurs exécutés au fil de l'eau au premier tour (le temps ne manquait pas !) et avec un carnet de route sous forme de diverses tables bien ajustées, le bolide était prêt à effectuer les tours suivants pied au plancher...

Mais d'autres difficultés nous attendaient.

Il aurait été naïf de croire que la voiture pouvait exécuter tous ces ordres parfaitement, sur chaque pointillé, à 20 km/heure sans corrections, donc sans asservissement.

Nous avons donc conservé le principe de l'asservissement du premier tour, mais il a fallu l'adapter aux circonstances et accélérer la lecture.

Ainsi, un allumage toutes les 10 millisecondes donnait à 20 km/heure moins de trois lectures par pointillé, ce qui était insuffisant

pour avoir une bonne sécurité. Le choix d'une fréquence d'allumage proportionnelle à la vitesse a donc été fait.

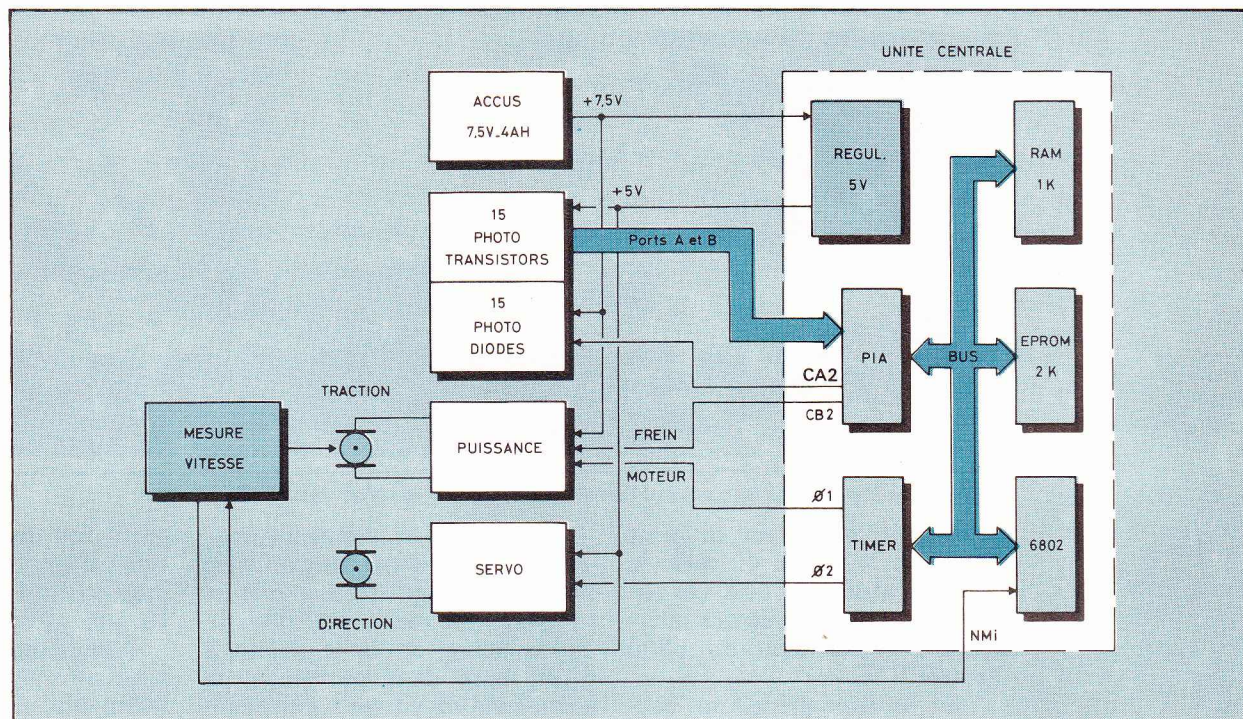
Ensuite, nous avons rendu l'asservissement plus « lâche ». C'est-à-dire que, en cas d'écart de trajectoire, la correction n'était décidée que si cet écart était suffisant à grande vitesse, il fallait à tout prix éviter le « pompage ». Par ailleurs, la correction ne devait surtout pas être proportionnelle à l'écart, à la rigueur proportionnelle à la variation de cet écart, et encore il fallait la pondérer en fonction de la vitesse du véhicule, à la manière d'un véritable pilote.

La quantité de photodiodes placées à l'avant était bien utile pour mesurer cette variation de l'écart, mais il faut avouer que ce fut là la partie la plus délicate du programme. Notre spécialiste, Francis Dubois, s'en est bien tiré.

En ce qui concerne la sécurité, nous avons vu qu'une perte de pointillé était fatale. Il fallait donc éviter de se « séparer » du pointillé. Pour cela deux procédures exceptionnelles ont été rajoutées au programme : d'abord un ralentissement d'autant plus important que la variation de l'écart de trajectoire était important, ensuite un rôle particulier des deux photodiodes extrêmes qui représentaient la « limite autorisée ». Il faut bien reconnaître que, lors de la compétition, la qualité de la piste était telle que ces procédures n'ont pas servi.

Le modèle de compétition

Fig. 3. — Le modèle de compétition est équipé d'un microprocesseur 6802.



Au bout de six mois d'essai à l'aide du prototype, le modèle définitif était fixé. Il ne restait plus qu'à le monter.

Le programme était pratiquement au point, encore fallait-il l'adapter à une nouvelle électronique et à une nouvelle mécanique. La construction du modèle de compétition a été entreprise un mois avant la date des épreuves. Nous n'avons donc pas eu beaucoup de temps pour vérifier avec précision ses performances, mais elles étaient meilleures que celles du prototype.

En effet, compte tenu des défauts constatés sur le véhicule d'essais plusieurs modifications ont été apportées.

Le châssis

Le châssis de base du commerce était monté sur une plaque d'aluminium très mince et donc très souple. Cette technique assure une sorte de suspension pour les véhicules radiocommandés qui peuvent évoluer sur des pistes bosselées. Mais pour notre cas c'était un inconvénient : la barre de photodiodes placées à l'avant était fixée à ce châssis et suivait donc ses déformations. En conséquence, la hauteur variable des capteurs diminuait la précision de lecture. Le support commun aux roues et aux capteurs a donc été fabriqué dans une plaque de plexiglass beaucoup plus rigide.

La masse

La masse du modèle a été considérablement diminuée. Lors

des essais, il était capital que le véhicule possède une grande autonomie. C'est pourquoi des accus imposants avaient été choisis (6 éléments au cadmium-Nickel de 8 ampères-heure). Pour la compétition, ces accus ont été remplacés par 6 éléments de 4 ampères-heure, deux fois moins lourds. De la même manière, les transistors de puissance étaient longuement sollicités et leurs radiateurs occupaient une grande partie du véhicule. La surface de ces radiateurs a également été réduite de moitié. Enfin, un système électronique a été spécialement conçu pour le modèle, sur un seul circuit imprimé de 15 cm sur 25 cm, avec un minimum de composants. Ce système a remplacé l'énorme Kit MKD2 et son clavier.

Ainsi allégé, le modèle était beaucoup plus à l'aise sur la piste.

Ses accélérations étaient plus franches et sa vitesse limite, dans les virages serrés, beaucoup plus grande. Nous avons encore amélioré son comportement en introduisant un système de freinage électronique.

Le freinage

L'électronique de puissance du moteur de traction permettait de faire tourner celui-ci dans les deux sens, marche avant ou marche arrière avec deux commandes logiques évidemment exclusives. Comme on l'a vu, la commande de marche avant était réalisée par un créneau variable fourni par la sortie Ø2 du Timer 6840. Pour freiner, lorsque le bolide était à grande vitesse, il suffisait de délivrer un créneau de commande sur l'entrée « marche arrière ». Ce créneau, fourni par la sortie CB2 du

L'anticipation de la commande de vitesse était fonction de la grandeur de la variation de vitesse.

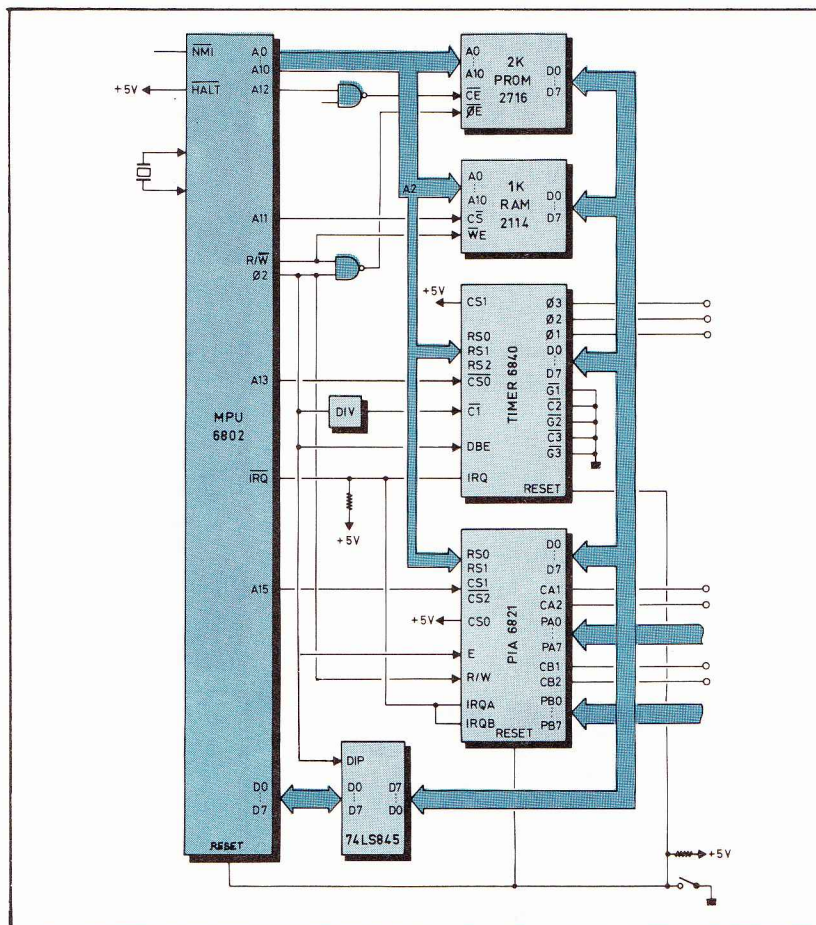


Fig. 4. - Schéma complet du modèle définitif.

PIA était maintenu tant que la nouvelle vitesse lue par le microprocesseur sur son entrée NMI n'était pas atteinte. Là aussi, évidemment, intervenaient les problèmes d'anticipation.

Le système

Le système définitivement adopté est décrit figures 3 et 4.

Le microprocesseur est un 6802, plus rapide, et plus simple à utiliser que le 6800, puisqu'il contient l'horloge et une mémoire interne en page zéro. Cette page est bien utile pour certaines instructions du 6800.

La mémoire associée au microprocesseur comprend 1 K de RAM et 2 K d'EPROM. La RAM sert exclusivement à enregistrer les informations lues au premier tour à raison d'un octet par pointillé. La longueur limite du tour de circuit est donc

de mille pointillés, soit environ 200 mètres. C'est plus qu'il n'en fallait puisque le circuit officiel mesurait moins de 60 mètres.

Rappelons que l'information enregistrée en RAM sur un octet est la longueur du créneau de direction qui varie en gros de 1 ms (pour braquage à gauche) à 2 ms (pour braquage à droite).

L'unité de valeur placée dans l'octet est donc de 1 ms/128, soit à peu près 8 microsecondes.

L'EPROM contient le programme et les tables paramètres. Là aussi nous avons vu large puisque, après les optimisations d'usage et de rigueur, tout le logiciel est contenu dans 1 K octet. (Rappelons qu'il est écrit exclusivement en assembleur).

La zone disponible peut aussi contenir la partie moniteur du MEK D2 qui gère le clavier de ce Kit. Nous allons voir son utilité.

Les entrées-sorties

Les entrées-sorties étaient constituées d'un PIA et d'un timer 6840.

Le timer, comme on l'a vu à plusieurs reprises génère les créneaux de commande et est relié directement aux électroniques de traction et de direction.

Le PIA possède plusieurs fonctions :

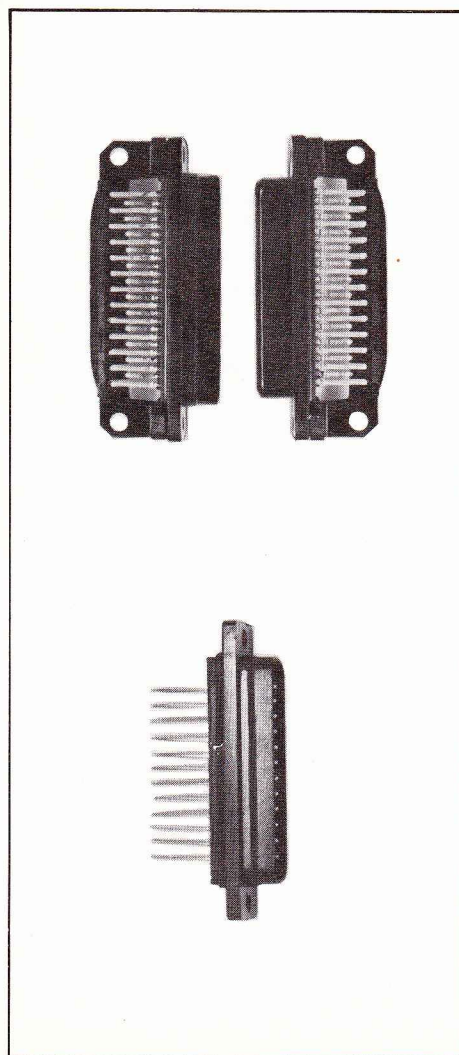
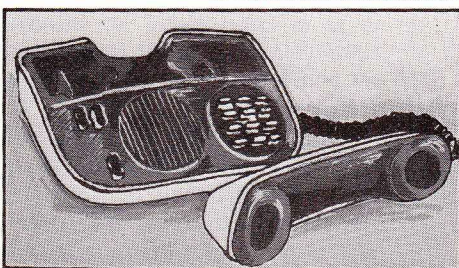
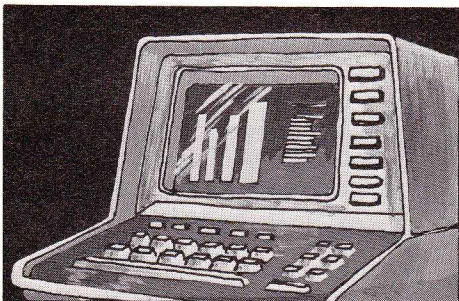
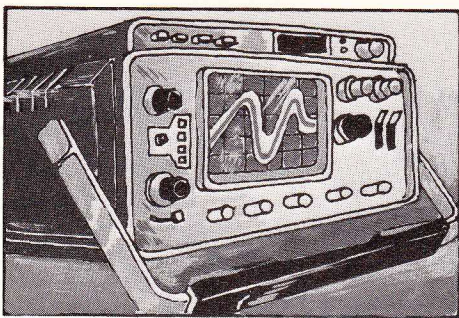
- d'une part il reçoit sur ses ports A et B (sauf PB7) les informations (blanc ou noir) fournies par les photodiodes à la cadence imposée par la sortie CA2.

- D'autre part les sorties CA2 et CB2 sont utilisées pour l'allumage des photodiodes et pour la commande de frein. Un connecteur relie tous les points aux organes correspondant de la voiture. Mais il est possible de débrancher ce connecteur et de brancher à la place le connecteur du clavier du MEK D2. L'unité centrale réalisée pour la voiture se comporte alors comme le Kit grâce à sa partie moniteur logée en EPROM et il est possible de modifier de façon dynamique des tables de paramètres installées alors en RAM. Cette procédure évite d'utiliser un programmeur d'EPROM (on n'en a pas toujours sous la main) pour modifier par exemple avant la course une table de vitesses. Elle a permis d'affiner les derniers essais du modèle.

Conclusion

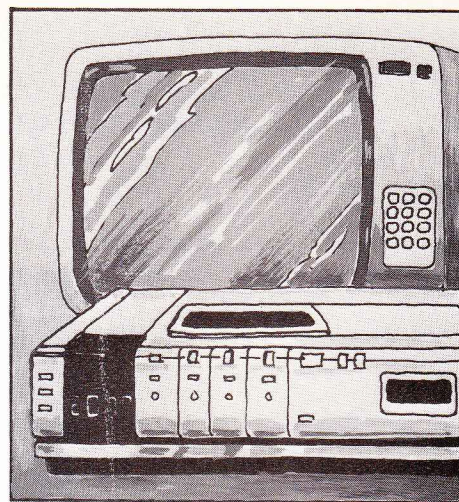
Il y aurait des quantités de choses à raconter, notamment les soirées ou les nuits passées à chercher des pannes qui n'existaient pas, la journée d'affolement passée la veille de notre départ pour Paris alors que la voiture ne marchait plus. Après plusieurs heures passées le nez dans les programmes, c'était les accus qui étaient à plat.

La formule μ vient de naître et elle a un long avenir devant elle. Nous retrouverons avec plaisir l'année prochaine nos concurrents d'un jour qui doivent déjà rêver de revanche. ■ Claude PRUNET



VOUS FABRIQUEZ CECI

Télécommunication privée,
HiFi-Vidéo, instruments de
musique, jeux électroniques,
micro-informatique, novotique
(télématique, bureautique,
informatique, robotique),
instrumentation, etc..



VOUS AVEZ BESOIN DE CELA 2000-D

le connecteur "D Subminiature"
tout plastique, disponible
dans les arrangements de
contacts classiques des
normes HE 501 et 502, dans
les versions mâles
et femelles.

CONNECTEURS CANNON
B.P. 20 F. 31770 COLOMIERS
Tél. (61) 78 53 33 + Télex 531600



Nom _____

Fonction _____

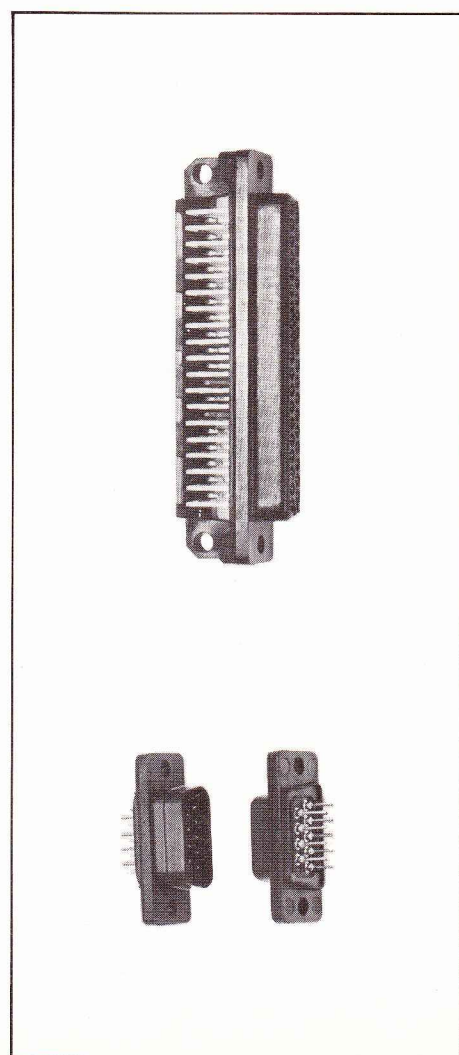
Société _____

Adresse _____

désire recevoir

- une documentation sur le connecteur 2000-D
- la visite d'un ingénieur technico-commercial
- un échantillon gratuit

MS



Le langage PASCAL

Modularité et Hash-Coding

Les techniques traditionnelles de programmation décomposent les programmes en deux parties. La première comprend les éléments de contrôle des traitements, et la seconde les données sur lesquelles portent ces traitements.

Généralement, la « structuration » des programmes consiste à bien organiser les procédures, fonctions, sous-programmes et autres traitements, et à considérer les données comme un aspect secondaire du problème. Ainsi dans certains programmes, la modification d'une simple ligne, d'une seule instruction peut demander plus de temps que la réécriture totale du programme.

Pour remédier à cet état de fait il est nécessaire, non seulement de décomposer les étapes de traitement, mais aussi d'en rendre indépendantes ses différentes parties. En effet la grande difficulté de la composition d'un projet informatique tient beaucoup moins aux rapports entre les éléments de traitement qu'à la gestion et la communication des informations.

Cette prise de conscience conduit à la création de « modules » dont l'élément de base est une structure de données autour de laquelle se greffent les sous-programmes permettant d'accéder à cette structure.

Pour l'environnement de ce module, c'est-à-dire le reste du programme, la façon dont les informations sont agencées à l'intérieur de celui-ci, devient sans importance, seule la manière d'y accéder est spécifiée.*

Cet article illustre ce type de conception par le biais d'un programme effectuant une mini-gestion de stock. Bien que le problème ait été très simplifié, le lecteur ayant bien assimilé les techniques que nous vous présentons n'éprouvera aucune peine à les utiliser dans des cas plus complexes.

Un exemple de gestion...

Il est souvent nécessaire en programmation de pouvoir associer des informations de natures différentes. Un cas courant correspond, par exemple, à l'association du nom d'une entité avec diverses informations relatives à celle-ci.

Ainsi, dans un système de gestion du personnel d'une entreprise, on souhaite associer au nom de chaque employé toutes les informations le concernant : âge, salaire, situation familiale, numéro de sécurité sociale, adresse etc. On désire aussi, retrouver aisément ces informations, les changer ou les détruire en utilisant le nom de l'employé comme référence.

Le problème consiste à organiser les informations de manière à être en mesure, à partir du nom de l'employé, de déterminer une « zone de mémoire » où sont rangées les informations le concernant.

De nombreuses méthodes existent pour résoudre les difficultés relatives à la gestion des informations. L'une de ces méthodes est très efficace : le **hash-coding**.*

Afin d'illustrer ce problème, nous considérerons un cas spécifique de gestion de stock dans le-

quel chaque article est caractérisé par son nom et sa quantité.

Le but étant de présenter quelques méthodes de programmation, les problèmes spécifiques de gestion de stocks ont été ici considérablement simplifiés.

En réalité, le cas que nous vous présentons correspond à une gestion de tables des variables où chacune d'entre elles est définie par son nom et une valeur qui lui est associée. De telles gestions de tables se rencontrent dans un compilateur ou un interpréteur, lors de l'analyse syntaxique.

Ici, deux opérations seulement peuvent être effectuées sur le stock :

- L'état de stock ou quantité disponible pour chaque article ;
- La mise à jour de ces quantités.

Le programme

Le programme effectuant cette gestion de stock (fig. 1) s'intitule GESTION. Ses sous-programmes peuvent être répartis en trois groupes :

■ Traitement des communications avec l'utilisateur :

INTERACTION, INTERROGATION, MISEAJOUR et LI-RENOM.

■ Gestion des informations en mémoire :

INITIALISER, CHANGER, CHERCHER, HOMONYME

■ Traitement des erreurs :
ERREUR

Le rôle du programme principal consiste seulement à initialiser le système, à imprimer quelques messages et à appeler la procédure INTERACTION dont le but est de traiter le dialogue entre l'utilisateur et le système.

Les procédures MISEAJOUR et INTERROGATION réalisent la lecture des informations tapées au clavier et l'impression des réponses.

Celles-ci délèguent le travail de manipulation de ces données aux procédures CHANGER et CHERCHER.

Le stock est défini par les informations contenues dans le tableau ETAT, dont chaque élément décrit un article particulier. Le nombre d'articles différents est donné dans la variable NBARTICLES.

Le rôle du sous-programme CHERCHER consiste à retrouver la quantité disponible en fonction du nom de l'article.

La procédure CHANGER est plus complexe. En effet si son rôle

* L'importance de la modularité a été reconnue puisqu'elle a été incorporée dans la structure même de la plupart des langages récents (Modula ou Ada par exemple).

* Hash-Coding : littéralement, hachage de l'information.

Il faut organiser les informations de manière à être en mesure de déterminer une zone mémoire contenant toutes les informations relatives à un même élément.

Le programme GESTION

```

program GESTION(INPUT,OUTPUT);
  label
  1;
  const
    MAXNOM =10;
    MAXETAT =300;
  type NOMS =array[1..MAXNOM]of CHAR;
  ARTICLE =
    record
      NOM:NOMS;
      QUANTITE:INTEGER
    end;
  ETATSTOCK=array[1..MAXETAT]of ARTICLE;
  var TERMINE:BOOLEAN;
  ETAT:ETATSTOCK;
  NEARTICLES:INTEGER;

  procedure ERREUR(NUMERO:INTEGER);
  begin
    WRITE('***** ERREUR: ');
    case NUMERO of
      1: WRITE('Trop d'articles. ');
      2: WRITE('Article inconnu. ');
      3: WRITE('Commande inconnue. ');
    end;
    Writeln;
    goto 1;
  end (*ERREUR*);

  procedure INITIALISER;
  begin
    NEARTICLES:=0;
  end (*INITIALISER*);

  procedure CHANGER(NOM:NOMS;QUANTITE:INTEGER);
  var I:INTEGER;
  begin
    I:=0;
    repeat
      I:=I+1;
      if I>NEARTICLES then
        (*Nouvel article*)
        if NEARTICLES>=MAXETAT then ERREUR(1)
        else begin
          NEARTICLES:=NEARTICLES+1;
          ETAT[NEARTICLES].NOM:=NOM
        end
      until ETAT[I].NOM=NOM;
      ETAT[I].QUANTITE:=QUANTITE
    end (*CHANGER*);

  function CHERCHER(NOM:NOMS):INTEGER;
  var I:INTEGER;
  begin
    I:=0;
    repeat
      I:=I+1;
      if I>NEARTICLES then ERREUR(2)
      until ETAT[I].NOM=NOM;
      CHERCHER:=ETAT[I].QUANTITE
    end (*CHERCHER*);

  procedure LIRENOM(var NOM:NOMS);
  var I,K:INTEGER;
  C:CHAR;
  begin
    repeat
      READ(C)
    until C<>' ';
    I:=1;
    NOM[I]:=C;
    while(I<MAXNOM)and not EOLN(INPUT)do
      begin
        I:=I+1;
        READ(NOM[I])
      end;
    for K:=I+1 to MAXNOM do NOM[K]:= ' '
  end (*LIRENOM*);

  procedure MISEAJOUR;
  var NOM:NOMS;
  QUANTITE:INTEGER;
  begin
    Writeln('Article: ');
    READLN;
    LIRENOM(NOM);
    Writeln('Quantite: ');
    READLN;
    READ(QUANTITE);
    CHANGER(NOM,QUANTITE)
  end (*MISEAJOUR*);

  procedure INTERROGATION;
  var NOM:NOMS;
  QUANTITE:INTEGER;
  begin
    Writeln('Article: ');
    READLN;
    LIRENOM(NOM);
    QUANTITE:=CHERCHER(NOM);
    Writeln('Quantite = ',QUANTITE)
  end (*INTERROGATION*);

  procedure INTERACTION;
  var C:CHAR;
  begin
    READLN;
    READ(C);
    if C in['>','?','.'] then
      case C of
        '>': MISEAJOUR;
        '?': INTERROGATION;
        '.': TERMINE:=TRUE
      end
    else ERREUR(3)
  end (*INTERACTION*);

  (* Corps du programme GESTION *)
  begin
    Writeln('Bonjour...');
    Writeln('Les commandes disponibles sont:');
    Writeln(' > pour mettre a jour la quantite disponible d'un article');
    Writeln(' (il peut s'agir d'un nouvel article);');
    Writeln(' ? pour demander la quantite actuellement disponible',
      ' d'un article;');
    Writeln(' . pour terminer votre session. ');
    Writeln;
    INITIALISER;
    TERMINE:=FALSE;
    repeat
      INTERACTION;
    until TERMINE;
    Writeln('Au revoir...')
  end (*GESTION*)

```

Fig. 1. - Le programme GESTION effectuée de manière totalement modulaire une mini-gestion de stock. Ainsi les modifications à l'intérieur d'un « module » n'entraînent aucune transformation sur les autres parties du programme.

se borne à modifier la valeur d'un article lorsque celui-ci existe déjà, elle doit, dans le cas où le nom spécifié ne correspond à aucun article, créer un nouvel article et lui associer la quantité correspondante.

En pratique l'analyse séquentielle effectuée par les procédures CHERCHER et RANGER pour accéder aux informations n'est pas très efficace. Il faut en effet lire en moyenne la moitié des noms avant de trouver l'article recherché. Pour de grands stocks pouvant contenir plusieurs milliers d'articles, ce processus se révèle long et coûteux en temps machine.

Nous allons donc utiliser une autre méthode de rangement de l'information qui nous permettra des accès plus rapides.

Généralement une modification concernant la structure des données entraîne une réorganisation de tout le programme. Mais ici notre programme a été écrit soigneusement en séparant totalement dans des modules distincts les différentes phases du traitement. Il est de ce fait possible de modifier les sous-programmes de dialogue avec l'utilisateur sans transformer ceux qui gèrent les informations en mémoire, l'inverse étant aussi réalisable.

Cette séparation des responsabilités que l'on nomme techniquement « modularité » permet de modifier certaines parties du traitement tout en évitant les interactions parasites entre les sous-programmes.

C'est une des clés de la programmation fiable.

En ce qui concerne notre programme nous n'aurons donc à modifier que les trois procédures INITIALISER, CHERCHER et CHANGER. Il est intéressant de noter que la procédure INITIALI-

Fig. 2. - Deuxième version du sous-programme de gestion des informations. Cette version accélère les recherches en utilisant la technique du « Hash Coding ».

Fig. 3. - Exemple d'utilisation du programme GESTION. Les lignes décalées d'un caractère vers la droite correspondent aux données introduites par l'utilisateur.

```

procedure INITIALISER;
var I:INTEGER;
begin
for I:=1 to MAXETAT do ETAT[I].NOM[1]:=' '
end (*INITIALISER*);

function HACHER(NOM:NOMS):INTEGER;
var SOMME,RESTE,I:INTEGER;
begin
SOMME:=1*ORD(NOM[1])+13*ORD(NOM[2])+11*ORD(NOM[3])+7*ORD(NOM[4]);
for I:=5 to MAXNOM do SOMME:=ORD(NOM[I])+SOMME;
RESTE:=SOMME mod MAXETAT;
HACHER:=RESTE+1
end (*HACHER*);

procedure CHANGER(NOM:NOMS;QUANTITE:INTEGER);
var HCODE,I:INTEGER;
begin
HCODE:=HACHER(NOM);
I:=HCODE;
while ETAT[I].NOM<>NOM do
if ETAT[I].NOM[1]=' ' then
(*Nouvel article*)
ETAT[I].NOM:=NOM
else begin
if I=MAXETAT then I:=1 else I:=I+1;
if I=HCODE then ERREUR(1)
end;
ETAT[I].QUANTITE:=QUANTITE
end (*CHANGER*);

function CHERCHER(NOM:NOMS):INTEGER;
var HCODE,I:INTEGER;
begin
HCODE:=HACHER(NOM);
I:=HCODE;
while ETAT[I].NOM<>NOM do
begin
if ETAT[I].NOM[1]=' ' then ERREUR(2);
if I=MAXETAT then I:=1 else I:=I+1;
if I=HCODE then ERREUR(2)
end;
CHERCHER:=ETAT[I].QUANTITE
end (*CHERCHER*);
    
```

Fig. 2

```

gestion

Bonjour...
Les commandes disponibles sont:
> pour mettre a jour la quantite disponible d'un article
   (il peut s'agir d'un nouvel article);
? pour demander la quantite actuellement disponible d'un article;
. pour terminer votre session.

>
Article:
navet
Quantite:
60
>
Article:
orange
Quantite:
45
?
Article:
navet
Quantite =          60
?
Article:
pomme
***** ERREUR: Article inconnu.
>
Article:
navet
Quantite:
55
?
Article:
navet
Quantite =          55
.
Au revoir...
    
```

Fig. 3

La « modularité » permet de modifier certaines parties du traitement tout en évitant les interactions parasites entre les sous-programmes.

```

function HOMONYME(NOM1,NOM2:NOMS):BOOLEAN;
var I:INTEGER;
    PAREIL:BOOLEAN;
begin
  PAREIL:=TRUE;
  I:=1;
  while(I<=MAXNOM)and PAREIL do
  begin
    PAREIL:=(NOM1[I]=NOM2[I])and PAREIL;
    I:=I+1
  end;
  HOMONYME:=PAREIL
end (*HOMONYME*);

```

Fig. 4. - Dans le cas où les comparaisons de tableaux ne sont pas possibles, la fonction HOMONYME doit être créée afin de pouvoir comparer deux noms.

SER qui nous semblait inutile au premier abord, va nous permettre de ne pas modifier le programme principal.

Le Hash-Coding

Accéder directement dans le tableau ETAT à l'information associée à un nom, nécessiterait d'indiquer le tableau avec ce nom.

Malheureusement, il n'est pas permis en PASCAL d'indiquer un tableau par un autre tableau. En outre, les variables de types NOMS peuvent prendre un très grand nombre de valeurs différentes.

Il est possible en effet, de composer plus de 26^{10} mots différents de taille inférieure ou égale à 10 avec les 26 lettres de l'alphabet, ce qui empêche toute utilisation d'un mot comme indice d'un tableau.

La technique, dite hash-coding, consiste à associer un indice du tableau à chaque nom.

Cette opération est effectuée dans le programme par l'intermédiaire de la fonction HACHER qui accepte un NOM comme paramètre et fournit un entier utilisé pour indiquer ETAT.

Comme le nombre de noms possibles est beaucoup plus important que le nombre d'indices, nous obtiendrons des « collisions » c'est-à-dire des indices identiques provenant de noms distincts.

La répartition des noms dans ETAT dépend de la fonction HACHER. Une fonction HACHER bien construite fournira une répar-

tition uniforme sur l'ensemble des indices.

Le nombre de collisions obtenues dépend en réalité des taux de remplissage du tableau. Un tableau plus faiblement rempli ne provoquera que peu de collisions; mais celles-ci étant de toutes manières inévitables il y a lieu d'en tenir compte et de leur prévoir un traitement particulier.

Nous avons jusqu'à présent considéré l'entier fourni par la fonction HACHER, appelons-le H-CODE, comme l'adresse de l'élément dans le tableau ETAT. En réalité, afin d'éviter les collisions, ce nombre peut être pris comme le point de départ d'une recherche séquentielle. S'il n'y a pas de collision, alors l'entier H-CODE correspond bien à l'adresse de l'élément, en revanche dans le cas d'une collision il y aura lieu d'effectuer une recherche supplémentaire dont l'origine coïncidera avec cet entier.

Pour placer un nouvel élément dans le tableau il suffit de calculer l'entier H-CODE associé à ce nom.

Si l'élément correspondant à cet indice est vide il est alors possible de placer un nouvel article, dans le cas contraire une recherche séquentielle doit être effectuée pour trouver un élément inutilisé.

Modifications du programme

Grâce à l'architecture modulaire, la modification du pro-

gramme GESTION n'implique qu'une réécriture des sous-programmes INITIALISER, CHANGER et CHERCHER et l'écriture de la fonction HACHER.

Ces modifications sont présentées figure 2.

La fonction CHERCHER a été transformée de manière à effectuer une recherche séquentielle d'un article à partir de l'indice fourni par la fonction HACHER. Lorsque, la fonction échoue dans sa recherche, (elle ne trouve pas le nom désiré), la procédure ERREUR est alors appelée.

Le sous-programme CHANGER procède de façon similaire. Lorsqu'un élément inutilisé est rencontré au cours de la recherche séquentielle, un nouvel article y est placé.

La figure 3 présente un exemple d'application du programme de gestion. Initialement les commandes disponibles sont affichées.

Le dialogue programme/ utilisateur commence par l'introduction d'un article et de sa quantité.

De nombreuses implémentations de Pascal ne permettent pas de comparer les tableaux, ce qui a lieu chaque fois qu'il faut comparer des noms dans les procédures CHERCHER et CHANGER.

Dans ce cas il est possible d'effectuer cette comparaison en utilisant la fonction HOMONYME présentée figure 4. Cette fonction accepte des noms comme paramètre et retourne la valeur vraie si deux noms sont identiques et faux dans le cas contraire.

Une autre modification doit être apportée dans le cas du PASCAL U.C.S.D. En effet celui-ci ne permet pas d'effectuer un GOTO à l'extérieur des procédures. Vous pouvez surmonter cette difficulté en déclarant tous les sous-programmes à l'intérieur de la procédure INTERACTION et en remplaçant l'instruction GOTO 1 dans la procédure ERREUR par l'instruction EXIT (INTERACTION). ■

B. LANG*

* B. LANG est chercheur à l'INRIA (Institut National de Recherche en Informatique et Automatique).

Le « Compact Disc Digital Audio » : une nouveauté en matière de reproduction sonore

Les derniers développements de la technologie ont permis d'obtenir des matériels de stockage et de reproduction sonore de haute qualité : le Compact Disc Digital Audio, issu des laboratoires d'études Philips, utilise le système d'enregistrement digital et un procédé de lecture optique d'un disque par rayon laser.



Photo A. – Le Compact Disc digital audio et son disque de 12 cm. Notez la simplicité de sa face avant.

Un disque de poche

Réalisé en chlorure de polyvinyle, le disque ne mesure que 120 mm de diamètre et 1,2 mm d'épaisseur.

L'information n'est pas inscrite à la surface mais à l'intérieur du disque sous forme d'une spirale enroulée à partir du centre. Le faisceau lumineux issu d'une diode laser GaAlAs est focalisé au niveau du plan où sont gravées les informations.

L'enregistrement digital est constitué d'une succession de « trous » et de « bosses » représentant les bits d'information. Pour un simple disque, la digitalisation du signal audio représente l'inscription de plus de 5 billions de bits. Les impulsions binaires sont regroupées en « mots » de 16 bits dans un système de codage P.C.M. (Modulation par impulsions codées).

Ce procédé digital d'enregistre-

ment permet en outre d'inscrire des informations relatives à une programmation des pistes : ainsi, celles-ci peuvent être sélectionnées en « accès direct ». Il est de même envisageable, par une série d'informations supplémentaires, de visualiser sur un écran certaines indications concernant l'enregistrement : nature d'une composition musicale, nom du chef d'orchestre, etc.

Une seule face du disque est enregistrée mais procure un temps d'écoute identique à celui obtenu avec les deux faces d'un 33 tours actuel.

Le principe de lecture

La lecture est réalisée par une tête optique (fig. 1), utilisant un mini-laser. Le rayon lumineux balaie la piste digitale de l'intérieur vers l'extérieur du disque à une vitesse linéaire constante. La succession des « trous » est détectée à

la vitesse de $4,3 \cdot 10^6$ bits par seconde. La réponse se fait à la vitesse de la lumière, donc beaucoup plus rapidement qu'avec le système traditionnel de la pointe sur le sillon. Chaque mot est lu à vitesse constante en moins de $10 \mu\text{s}$. Le résultat de la lecture optique se traduit par un flux d'impulsions exprimé dans le système PCM à 16 bits. Un convertisseur digital/analogique décode le flux mot par mot et délivre le signal audio-conventionnel.

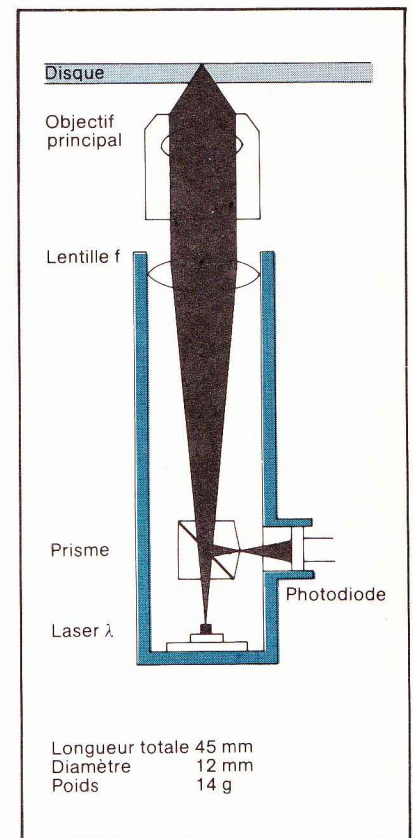


Fig. 1. – La tête de lecture optique de la platine Compact Disc. Par l'intermédiaire d'une lentille, le faisceau divergent issu du laser est converti en un faisceau parallèle : celui-ci est acheminé vers l'objectif principal où il est focalisé au niveau de la couche du disque contenant les informations digitales. La lumière réfléchie, modulée, est dirigée vers la photodiode par un prisme réfléchissant. Un coin intercalé entre le prisme et la photodiode permet de décomposer en deux parties différentes le faisceau réfléchi parvenant sur la photodiode. Le traitement des courants de sortie résultant de la diode permet alors de récupérer le signal audio et les informations d'erreurs de piste et de mise au point.

La platine Compact Disc

La taille du disque a permis de réduire les dimensions du lecteur :



Photo B. - Un microsillon 33 tours habituel et le « Compact Disc ». Le même temps d'écoute...

la platine Compact Disc a ainsi des dimensions équivalentes à celles d'une mini-platine cassette. Elle constitue une unité indépendante où sont intégrés tous les circuits électroniques nécessaires au raccordement sur un amplificateur HiFi. Sa manipulation se résume à la commutation de trois

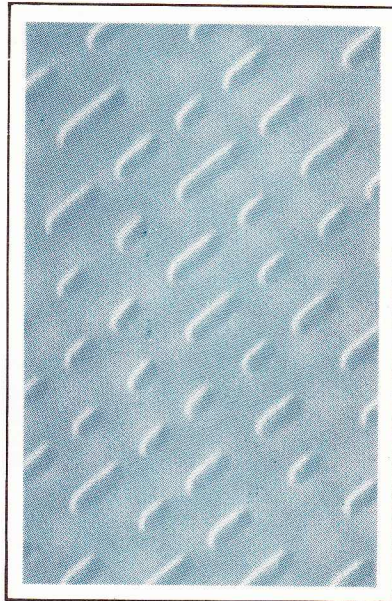
fonctions : lecture, arrêt et recherche automatique.

Le Compact Disc Digital Audio permet grâce à sa technologie, la reproduction d'un signal original avec une grande précision : la distorsion est éliminée et le pleurage inexistant. De plus, le rapport signal/bruit n'est plus fonction que

du contenu des « mots » utilisés pour codifier le signal original.

Le prix de ce système devrait se situer aux environs de 2 000 F (et 50 F pour le disque) au moment de sa commercialisation en 1983. ■

Photo C. - Les pistes du Compact Disc au microscope (grossissement x 5000).



JAXTON INFORMATIQUE S.A.

La Levratte 18 1260 Nyon/Suisse
Tél. 022/61 77 33 Télex 289 198 ICCU CH

IMS INTERNATIONAL MARKETING SERVICE

Rue de Vintimille 22 75009 Paris/France
Tél. 526 40 42 Télex 640 282



INSAC CONSEIL

ETUDES
ANALYSES
CAHIER DES CHARGES
AUDITS D'INSTALLATION
PLAN DE FINANCEMENT

INSAC PRODUITS

SERIE 2000
64 K mémoire de 2,4 à 20 MB
3 ports de sortie
SERIE 3000 A
Jusqu'à 256 K mémoire,
jusqu'à 100 MB.
5 ports de sortie

INSAC APRÈS-VENTE

EUREP
EULOG

SERVICE « PLUS »

INSAC PROFESSION

COMPTABILITÉ, STOCK
FACTURATION, SALAIRES
GESTION ADMINISTRATIVE
AVOCAT 80

ISS
INFORMATIQUE SYSTEME SERVICE
89, Boulevard de Sébastopol
75002 **PARIS** Tél. (1) 233.58.51

ETABLISSEMENT CHATILLON
25, Rue de Versoix
01210 **FERNEY-VOLTAIRE**
Tél. (50) 40.62.34

DISTRIBUTEURS AGRÉÉS :

Recherchons distributeurs
France-Suisse-Belgique

Pour plus de précision cerchez la référence 154 du « Service Lecteurs »

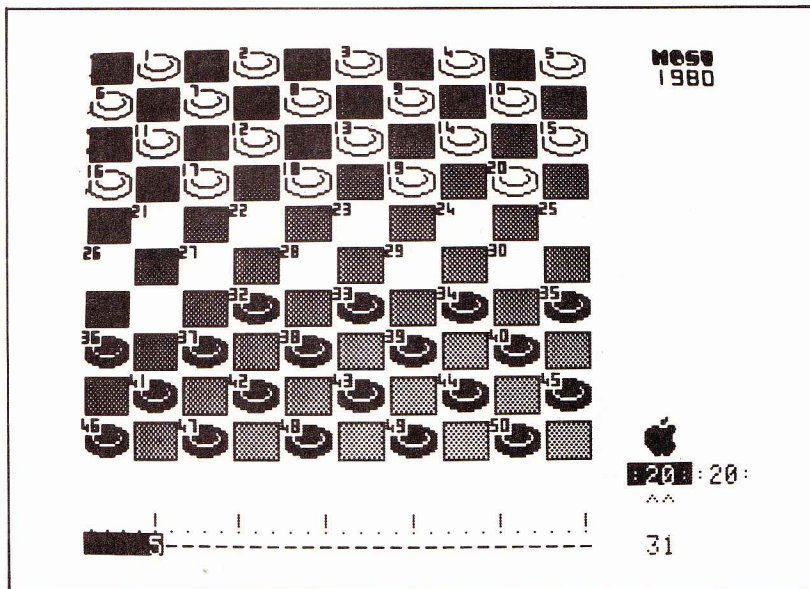
SOPHEL INFORMATIQUE
20, Rue Agent Galay
13012 **MARSEILLE**
Tél. (91) 93.11.13

SOKEL
Allmendstrasse 11
3052 **ZOLLIKOFEN**
Tel. (031) 57.64.22

Dames Challenger

au banc d'essai

Un programme de jeu de Dames appelé Dames-Challenger a été créé en France par une équipe d'ingénieurs du nom de MOST. Il est disponible actuellement sur trois micro-ordinateurs : Apple II, TRS 80 niveau II et PET Commodore.



Le jeu de dames tel qu'il apparaît sur l'écran de votre téléviseur.

Contrairement aux échecs le jeu de Dames à 100 cases ne semble retenir que depuis peu l'attention des informaticiens. Il est vrai que ce jeu n'existant pas outre-atlantique (aux U.S.A. on joue aux **Checkers**, version simplifiée du jeu de Dames qui se présente sur un échiquier de 64 cases), il n'a donc pu disposer de la puissance de vente du marché américain.

Nous avons eu tout récemment l'occasion de tester une de ces machines joueuses aux dames et le résultat nous a très agréablement surpris.

Le programme autorise plusieurs niveaux de jeu (numérotés de 1 à 9). Après quelques essais, nous nous sommes attardés au niveau 4 qui correspond, approximativement, à une cadence de tournoi.

Deux parties nous ont permis d'évaluer ce programme.

Pour en rendre compte, nous utiliserons la notation traditionnellement adoptée par les damistes ainsi que par Dames-Challenger.

Les coups sont notés en donnant d'abord le numéro de la case de départ puis celui de la case d'arrivée en les séparant par une

croix (X), s'il s'agit d'une prise et par un tiret dans le cas contraire. En outre, les coups joués par les noirs sont mis entre parenthèses. La numérotation des cases est donnée figure 1.

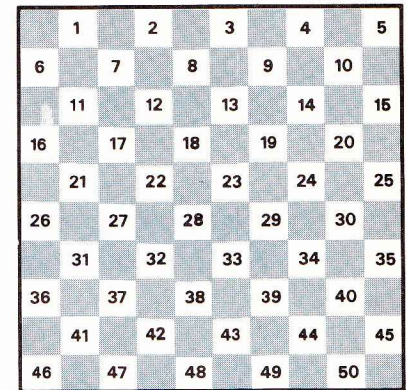


Fig. 1. - Cette numérotation est universellement adoptée par les damistes, et c'est celle que Dames Challenger utilise.

La première partie

La suite des coups joués durant la première partie opposait le programme (noirs) à Jérôme Spick (blancs), Champion de France junior 1980, ici le programme Dames-Challenger était sur le niveau de force 4.

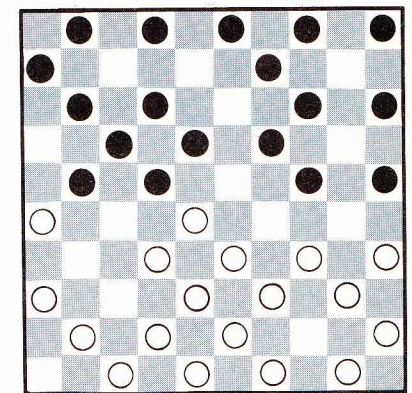


Fig. 2. - Après le 12^e temps des blancs, les noirs jouent et vont à dames.

	Blancs	Noirs		Blancs	Noirs
1.	32 - 28	(17 - 22)	8.	31 - 26	(16 - 21)
2.	28 x 17	(11 x 22)	9.	32 - 28	(19 - 24)
3.	37 - 32	(12 - 17)	10.	37 - 32	(13 - 19)
4.	41 - 37	(7 - 11) (a)	11.	40 - 34	(10 - 14)
5.	46 - 41	(8 - 12)	12.	44 - 40 ?? (b)	(24 - 29)
6.	34 - 29	(20 - 24)	Fig. 2		
7.	29 x 20	(14 x 25)			

a - La réponse théorique est 8-12 avec comme suite possible: 46 - 41 (6 - 11) 34 - 29 (19 - 23) 40 - 34 (14 - 19) 45 - 40 (10 - 14) 32 - 28 (23 x 32) 37 x 28 (16 - 21) 41 - 37 (11 - 16) 38 - 32 (5 - 10) 43 - 38 (3 - 8) 49 - 43 (1 - 6) 50 - 45 (7 - 11).

b - Une erreur énorme que la machine va correctement réfuter. Il était nécessaire de jouer 41 - 37 et 37 - 31 avec avantage pour les blancs qui menacent d'enchaîner l'aile droite des noirs.

BORDEAUX

BOUTIQUE MICRO

PET CBM ATOM
OHIO-SCIENTIFIC ACORN
NASCOM COMPUTEACHER
DISQUES - K 7 - PAPIER - DIVERS

PROGRAMMES

PET SOFT jeux et utilitaires
GESTION sur mesure
GESTION standard économique
AUTOMATISME industriel

LIBRAIRIE MICRO

SYBEX - OI - PSI

SERVICE APRES VENTE

LES PRIX :

ACORN	1.100.00 F HT	1.300.00 F TTC
SUPER BOARD		2.500.00 F
CBM 2001	4.650.00 F HT	5.450.00 F TTC
CBM 3001	21.900.00 F HT	25.750.00 F TTC
CBM 8001	34.850.00 F HT	41.000.00 F TTC

• Un exemple pour les programmes de gestion :

Comptabilité Générale	950.00 F HT	1.117.20 F TTC
PAIE	950.00 F	
Traitement de Texte	950.00 F HT	1.117.20 F TTC
Gestion Fichier/mailling	650.00 F HT	764.40 F TTC

Jeux de 60.00 F à 250.00 F TTC

FORMATION

ETUDE & DEVIS

Pour plus de précision cercelez la référence 155 du « Service Lecteurs »



Aquitaine Micro Informatique

134. Bd Président F.-Roosevelt
33800 BORDEAUX
Tel. (56) 91.78.74
entre Barrières Toulouse
et Saint-Genes

	Blancs	Noirs		Blancs	Noirs
13.	33 × 13	(22 × 44)	34.	48 - 42 (i)	(32 - 37)
14.	13 × 22	(17 × 46)	35.	42 × 31	(26 × 37)
15.	26 × 8	(3 × 12)	36.	25 - 20 ? (j)	(1 - 6)
16.	50 × 39	(46 - 19) ? (e)	37.	33 - 28 (k)	(34 - 39)
17.	38 - 33	(19 - 41)	38.	28 - 22	(39 - 43)
18.	43 - 38	(25 - 30) ? (d)	39.	22 - 18	(43 - 49) (l)
19.	34 × 25	(9 - 13)	40.	20 - 14	(19 × 10)
20.	40 - 34	(11 - 16)	41.	47 - 42	(37 × 48)
21.	42 - 37	(41 × 43)	42.	12 - 7	(2 - 11)
22.	49 × 38 (e)	(6 - 11)	43.	19 - 13 (m)	(48 - 30) ? (n)
23.	36 - 31	(16 - 21)	44.	35 × 24	(49 - 35)
24.	34 - 39	(21 - 26)	45.	13 - 9	(35 - 19)
25.	31 - 27	(13 - 19)	46.	9 - 3	(19 - 23)
26.	38 - 32	(15 - 20) (f)	47.	3 - 21	(10 - 14)
27.	39 - 34	(12 - 18)	48.	21 - 3	(4 - 9)
28.	34 - 30	(19 - 23)	49.	3 - 21	(14 - 19)
29.	30 - 24	(23 × 34)	50.	21 - 3 ?? (o)	(19 - 46)
30.	24 × 15	(11 - 17) (g)			
31.	32 - 28	(14 - 19) (h)			
32.	27 - 21	(18 - 23)			
33.	31 × 12	(23 × 32)			

Les blancs abandonnent

c - Il fallait jouer (14 - 20) 46 - 10 et (9 - 14) et la dame noire, cachée derrière ses pions est imprenable.

d - (1 - 7) gagnait encore et si les blancs capturent la dame par 42 - 37 (41 × 43) 49 × 38 les noirs restent avec un pion de plus ce qui est théoriquement gagnant.

e - Les blancs ont bien rétabli la situation. Les fautes de l'ordinateur s'expliquent par le fait que, au niveau 4, celui-ci ne « voit » pas les combinaisons exigeant plus de 4 demi-coups sans prise.

f - Les noirs qui ont déjà le désavantage accroissent leurs difficultés. (15 - 20) est un coup plus que douteux stratégiquement.

g - Pare l'attaque 35 - 30 par (17 - 22) 30 × 39 (22 × 31) avec égalité.

h - L'attaque 35 - 30 n'est pas toujours possible : si 35 - 30 (18 - 23) 30 × 39 (23 × 21).

i - sur 35 - 30 suivrait (2 - 7) 30 × 19 (7 × 18) 48 - 42 avec égalité.

j - Le meilleur était bien sûr 35 - 30 (2 - 7) 30 × 39 (7 × 18) avec égalité.

k - Les blancs sont contraints de laisser les noirs aller à dame.

l - Les noirs menacent (37 - 42) 47 × 38 (49 × 27 × 9 × 25) aussi simple que terrible...

m - Les blancs ont payé très cher le privilège d'aller à dame.

n - Un coup encore gagnant mais plus radical était (10 - 14) 13 - 8 (48 - 26) 8 - 3 8 - 2 (26 - 42) 2 × 16 (42 - 38) 16 × 43 (49 × 16) 11 - 17 (3 × 25) 4 - 9 (25 × 21) 26 × 8 N+.

o - Une faute monumentale sans laquelle les blancs auraient pu résister encore longtemps... mais la partie est, de toute façon perdue.

La deuxième partie

Cette deuxième partie oppose Dames Challenger niveau 4 (blancs) à Luc Guinard (noirs).

	Blancs	Noirs
1.	34 - 29	(17 - 22)
2.	31 - 27 (a)	(22 × 31)
3.	37 × 26	(11 - 17)
4.	40 - 34	(19 - 23)
5.	33 - 28	(14 - 19)
6.	36 - 31	(7 - 11)
7.	39 - 33	(1 - 7)
8.	43 - 39	(10 - 14)
9.	44 - 40	(5 - 10)
10.	42 - 37 (b)	(17 - 22) !
	fig. 3	
11.	28 × 17	(11 × 22)
12.	32 - 27 ? (c)	(19 - 24) (d)
13.	41 - 36 (e)	(14 - 19)
14.	26 - 21 ? (f)	(10 - 14)
15.	35 - 30 (g)	(24 - 44)
16.	49 × 40	(19 - 24)
17.	40 - 35	(7 - 11)

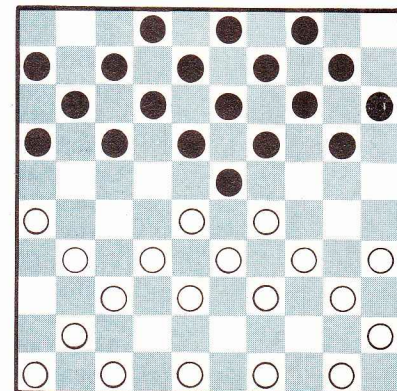


Fig. 3. - Après le 10^e temps des blancs, les noirs jouent et prennent un avantage positionnel.

	Blancs	Noirs		Blancs	Noirs
18.	50 - 44 ? ! (h)	(22 - 28)	35.	45 x 34	(23 - 28)
19.	33 x 22	(24 x 42)	36.	27 - 22	(28 - 33)
20.	47 x 38	(11 - 17) !	37.	38 x 29	(17 x 28)
21.	22 x 11	(6 x 26) (i)	38.	29 - 23	(28 - 33)
22.	35 - 30	(20 - 25)	39.	23 x 14	(9 x 20)
23.	39 - 33	(14 - 19)	40.	36 - 31	(33 - 38)
24.	33 - 29	(2 - 7)	41.	31 - 27	(38 - 43)
25.	38 - 33	(4 - 10)	42.	46 - 41	(43 - 48)
26.	29 - 24	(10 - 14)	43.	27 - 21	(16 x 27)
27.	33 - 29	(23 - 28)	44.	41 - 36	(48 x 31)
28.	48 - 43	(18 - 23)	45.	30 - 24	(20 x 40)
29.	29 x 18	(12 x 23)	46.	35 x 44	(31 - 48)
30.	27 - 22	(28 x 17)	47.	44 - 40	(25 - 30)
31.	31 - 27	(7 - 11)	48.	40 - 35	(30 - 34)
32.	43 - 38	(14 - 20)	49.	36 - 31	(26 x 37)
33.	44 - 40	(20 x 29)	50.	35 - 30	(34 x 25)
34.	40 - 35	(29 x 40)			

a - L'ordinateur ne joue pas de manière classique. En voici une nouvelle fois la preuve. La suite usuelle est 40 - 34 (11 - 37) 45 - 40 (6 - 11) 50 - 45 (1 - 6) 31 - 26 (16 - 21) 32 - 28 (19 - 23) 28 x 19 (14 x 23) 34 - 30 (10 - 14) 30 - 24 (23 - 28) avec beaucoup de tension de part et d'autre.

b - Une faute par laquelle les blancs vont se trouver très rapidement en difficulté. Meilleur était : 41 - 36.

c - 41 - 36 sans être extraordinaire était le meilleur coup de défense.

d - Les noirs enchaînent le centre noir.

e - Le seul coup noir pour ne pas perdre le pion.

1) 38 - 32 ou 37 - 32 (22 - 28) 33 x 22 (24 x 44) 50 x 39 (16 - 21) 26 x 17 (12 x 21) 27 x 16 (18 x 36)

2) 26 - 21 (22 - 28) 33 x 22 (24 x 42) 47 x 38 (23 - 28) 22 x 33 (18 - 22) 27 x 18 (16 x 47) (N+).

3) Sur 50 - 44 49 - 44 49 - 43 48 - 43 (22 - 28) 33 x 22 (24 x 42) 47 x 38 (16 - 21) 27 x 16 (18 x 47).

4) Sur 48 - 42 ou 47 - 42 toujours (22 - 28) et (16 - 21).

f - 50 - 44 ou 47 - 42 perd moins de matériel que le coup joué.

g - Rien de mieux contre la menace de perdre deux pions (22 - 28) 33 x 22 (24 x 44) 50 x 39 (23 - 29) 19 x 26.

h - Un coup « vicieux » de la machine. Elle spéculait sur (22 - 28) 33 x 22 (24 x 42) 47 x 38 (13 - 19)?? 24 x 22 (20 x 49) 46 - 41 (49 x 32) 37 x 10 (B+).

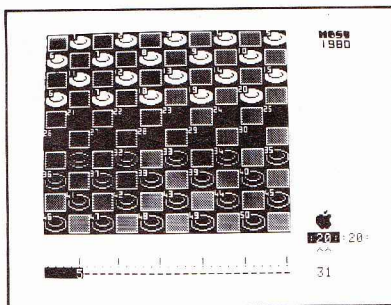
i - Avec deux pions d'avance, la partie est facilement gagnée. Mais la machine va continuer jusqu'au dernier...

Ces deux parties montrent bien la force tactique de ce programme. Même un joueur expérimenté ne peut se contenter de jouer négligemment (car la machine le rappelle rapidement à l'ordre) comme on a pu le constater au cours de la première partie. Celle-ci est en effet capable de calculer, et de jouer des combinaisons très complexes et devient donc redoutable lorsqu'elles comportent un grand nombre de prises.

Ainsi sur le plan tactique pur, ce type de programme est certainement capable de battre la quasi-totalité des joueurs humains.

Cependant, celui-ci ne possédant aucune notion stratégique véritable, un joueur pourra sans doute, après quelques parties, apprendre à exploiter ces faiblesses dans ce domaine, surtout en fin de partie.

Le problème reste donc, pour un joueur d'arriver jusque-là, sans avoir subi la force tactique brutale de Dames-Challenger.



Autre représentation du damier.

Le jeu de Dames est loin d'être uniquement le jeu que l'on pratique dans les arrières-salles des cafés, et nous espérons que Dames-Challenger* contribuera à en rendre conscient ceux qui, possédant un micro-ordinateur, sont en mesure de jouer contre ce programme ■

Luc GUINARD
(Candidat Maître National)

* Le programme Dames-Challenger est distribué par la société SIVEA au prix de 195 F.

**PKW
7000**

**LE PROGRAMMATEUR
UNIVERSEL**
EPROM E EPROM PROM



**UNIVERSEL : modules enfichables
PUISSANT : utilise le 8085
PETIT : 290 x 190 x 65 mm
LEGER : 1,8 kg**

connectable avec tous
types de calculateurs
ou outil de développement
RS 232C.

**UN GAGE DE FIABILITE
POUR UN RAPPORT
QUALITE/PRIX EXCEPTIONNEL.**

ce programmeur
fait partie de la série
des programmeurs
PECKER.



électronique

20/22, rue des quatre frères
Peignot
75015 Paris
Tel: (1) 575 53 53

« Pour plus de précision cerchez la référence 156 du « Service Lecteurs » »

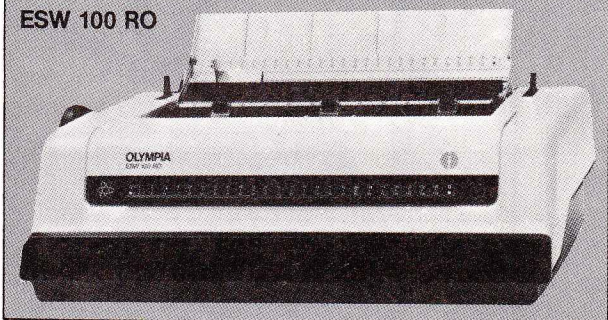
Imprimantes pour vos micros :

l'écriture "Direction" des Olympia ES 100

Les imprimantes Olympia ESW 100 KSR et ESW 100 RO ont la même écriture que les machines à écrire électronique de bureau Olympia ES 100 : une écriture "Direction." Grâce à la roue à caractères, l'écriture est interchangeable.



ESW 100 KSR



ESW 100 RO

ESW 100 KSR

Machine à écrire électronique connectable : EIA RS 232 C, V 24. Boucle de courant 20 mA. Déjà connectée sur : Canon, Zénith, Digital Equipment, Hewlett Packard,

Commodore, Apple, Sord, Lomac, Tandy, etc...

ESW 100 RO

A interface parallèle compatible Centronics. Déjà connectée sur : BOSS (Olympia), MICRAL (R2E)

Distributeurs agréés :

- INFOR/ELEC
7 rue Traversière
92100 BOULOGNE
Tél (1) 621.23.07
- MICROSCOP
39 rue de la Figairasse
34100 MONTPELLIER
Tél (67) 27.53.09
- DELCROIX
17 rue du Cdt Defontaines
80000 AMIENS
Tél (22) 92.29.70

- COMPTA FRANCE
3 Route de la Reine
92100 BOULOGNE
Tél (1) 603.76.40

Pour vous informer :
Olympia France
Département O.E.M.
10 av. Réaumur
92142 CLAMART Cedex
Tél : 630.21.42
Poste 181



**Olympia International
Machines et Systèmes de Bureau**

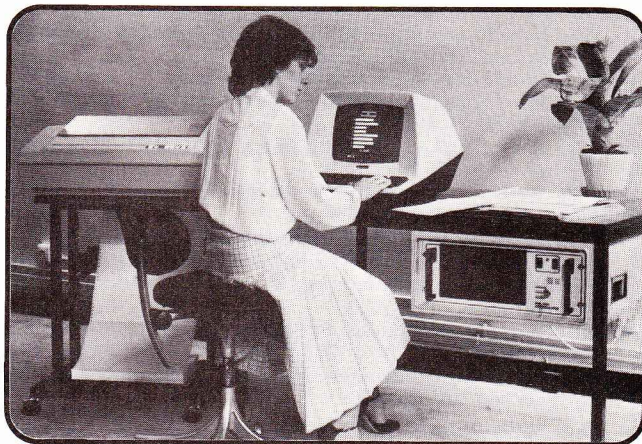
MICROMATIQUE ●●●● Europe s.a.

82/84 boulevard des batignolles 75017 Paris - tél. 387.59.79+

PM.E. / PM.I.

Cabinet Expert Comptable

POUR VOTRE GESTION PROTEUS III E "SIMPLICITE"



De 1.700,00 à 3.500,00 F HT/mois
"CLE EN MAIN"

VOTRE INDEPENDANCE SAUVEGARDEE

Entretien garanti par contrat
(Paris, Région Parisienne)

MICROMATIQUE Europe s.a.
82/84 boulevard des batignolles 75017 Paris - tél. 387.59.79+

Contre retour de ce bon une DOCUMENTATION
COMPLETE vous sera adressée.

Société _____ Nom _____

Adresse _____

Ville _____ Tél. _____

Pour plus de précision cercler la référence 158 du « Service Lecteurs »

2164 : une mémoire RAM de 64 K × 1 Bit

Intel annonce la première mémoire RAM de 64 K × 1 bit : la 2164.

D'ici 1985, le marché de ce type de mémoire est évalué à 1 milliard de dollars (chiffre à comparer au marché estimé pour 1981 : 100 millions de dollars).

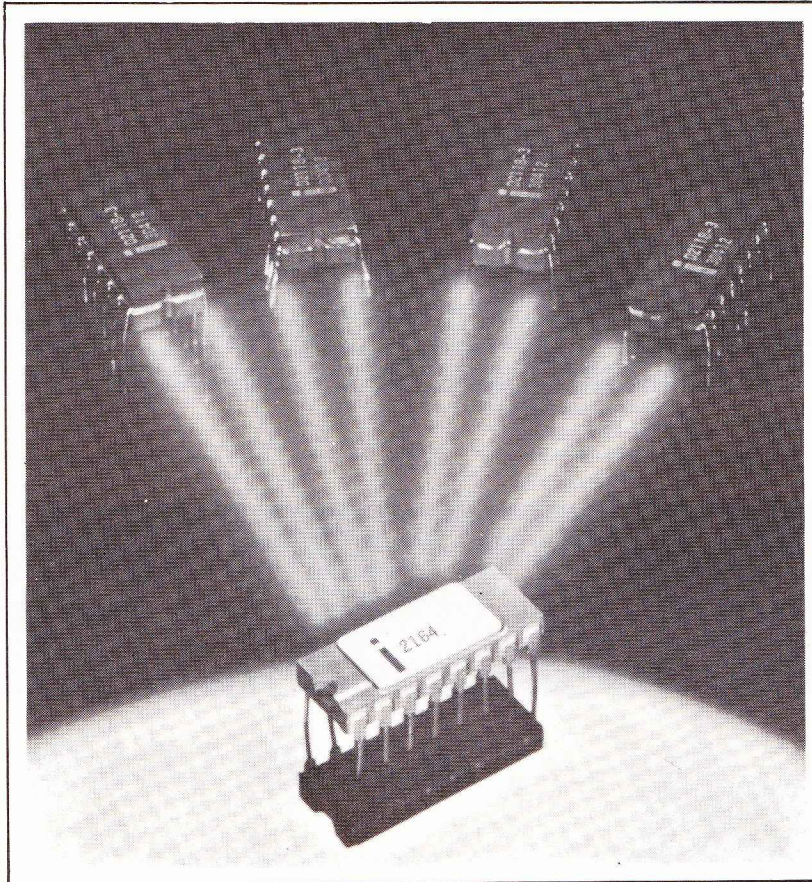


Photo 1. - La 2164 a une capacité de 64 K, soit l'équivalent de quatre boîtiers 2118.

Développée en technologie H-MOS (High performance Metal Oxide Semiconductor), la 2164 est une mémoire RAM dynamique de 65 536 mots de 1 bit. Livrée en boîtier DIP de 16 broches, cette mémoire est alimentée par une tension unique de 5 V ($\pm 10\%$). De plus, elle est compatible avec la 2118 (16 K) et le sera avec la future mémoire RAM de 256 K.

La 2164 ainsi que la RAM statique 2167 (16 K) sont les premiers produits fabriqués par INTEL selon une méthode dite « redondante » qui devrait grandement accroître leur disponibilité.

En effet, cette méthode consiste à intégrer 4 lignes et 4 colonnes de mémoire supplémentaires, en principe inutilisées dans la fonction de base.

Ainsi, lorsqu'un défaut est détecté au moment du test de la « puce », la colonne ou la ligne dé-

fectueuse est remplacée par son homologue « redondante ».

Cette technique vise à accroître de façon importante le rendement de fabrication (YIELD) conduisant directement à l'augmentation de la disponibilité et à une diminution sensible des délais de livraison.

En outre, pour les ingénieurs « système », INTEL développe un module mémoire bâti autour de boîtiers 2164.

Compatibilité

De par sa densité d'intégration (4 fois plus élevée que la 2118), la 2164 doit être logiquement le produit qui sera amené à la remplacer. Le brochage de la 2164 apparaît figure 1.

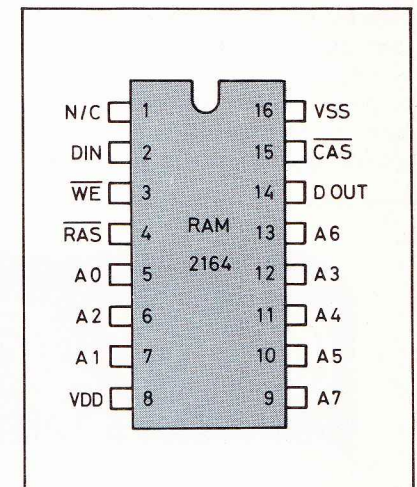


Fig. 1. - Brochage de la mémoire 2164.

L'expérience acquise dans le développement et la production de la 2118 a aussi apporté un « savoir faire » qui a considérablement aidé la conception et la fabrication de la 2164.

Le constructeur a laissé libre la broche 1 pour assurer la compatibilité avec les mémoires à venir (256 K).

Ainsi, l'utilisateur peut dès à présent, incorporer des 2164 dans ses matériels sans souci d'obsolescence.

Tableau 1. - Spécifications de chacun des trois types de 2164 commercialisés.

Type	Temps d'accès max	Courant max (actif)	Courant max (au repos)	Prix \$
2164-15	150 ns	60 mA	5 mA	81,20
2164-20	200 ns	55 mA	5 mA	64,95
2164-25	250 ns	50 mA	5 mA	52

Caractéristiques

Une des caractéristiques les plus intéressantes de cette mémoire liée à la technologie, est l'unique tension d'alimentation.

La tension négative nécessaire à la polarisation du substrat est générée de façon interne.

En ce qui concerne l'adressage, le multiplexage des 16 bits d'adresses permet de n'utiliser que 8 broches d'adresses (A0 - A7) et 2 lignes de sélection; RAS (Row Address Strobe) et CAS (Column Address Strobe).

La **figure 2** représente l'organisation interne de cette mémoire.

La famille des mémoires 2164 comporte 3 modèles (2164-15; 2164-20; 2164-25) dont les spécifications concernant les temps d'accès et la puissance dissipée sont résumés **tableau 1**.

Selon les références, les prix proposés par le constructeur se situent entre 50 et 80 \$ par boîtier pour des quantités de 100 unités. ■

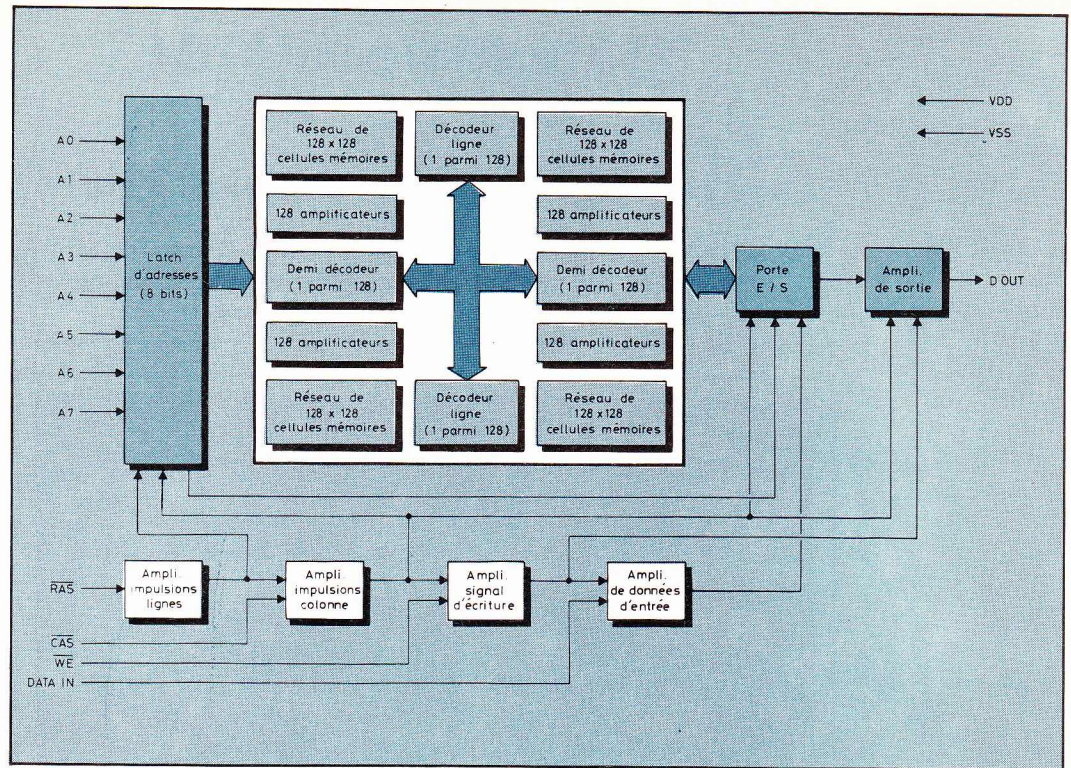


Fig. 2. - Structure interne de la 2164.



Cartes et Systèmes à Microprocesseurs

B.P. 84 - 38503 VOIRON Cedex



945 F TTC
le kit

(prix au 1.11.80)

JUNIOR COMPUTER

Micro-ordinateur monocarte basé sur le 6502, programmable en hexadécimal.

Mémoire : 1 K ROM avec moniteur + 1 K RAM.

Circuit d'interface 6532 (2 ports E/S + timer + 128 octets RAM).

Absolument complet avec alim., transfo., connecteurs.

En KIT : 945 F TTC

Monté : 1095 F TTC

Manuel de montage et de programmation : 50 F TTC.

Support altuglas formé, sérigraphié, colonnettes laiton chromé, visserie noir mat, housse de protection : 150 F TTC

Pour plus de précision cercele la référence 148 du « Service Lecteurs »

Vente par correspondance :

— Commande supérieure à 300 F : franco de port - sinon + 5 F

— Contre-remboursement : + 25 F

Commandes téléphonées et renseignements : (76) 50.05.31 De 13 h à 17 h

Matériels disponibles
sur stock



votre micro-informatique!

Vous êtes industriel, chercheur, enseignant, commerçant, particulier, membre de profession libérale.

Nous sommes **constructeurs** et **distributeurs**, nous avons une position de leaders sur le marché de la micro-informatique.

Nous avons une expérience unique en matière **d'installation et maintenance de systèmes**.

Nous **études et réalisons à la demande** le matériel et le logiciel de systèmes.

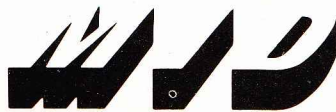
Nous sommes faits pour nous entendre.

• **Systèmes** : Systèmes MID 7924. Systèmes multipostes et multitâches. Systèmes d'acquisition et de traitement en temps réel. Contrôle de processus. Automates industriels.

• **Micro-ordinateurs** : Apple II, Apple III, Commodore, Pertec, Superbrain, etc.

• **Périphériques** : Floppys, disques durs, imprimantes, terminaux intelligents, tables traçantes, tables à digitiser.

• **Interfaces** : Entrées/Sorties parallèles et séries (TTL, V24 RS 232C, boucle de courant). Entrées analogiques multivoies, multigammes. Sorties analogiques. BUS IEEE-488. Entrées/Sorties BCD. Carte Horloge temps réel. Calcul rapide. Digitalisation d'image vidéo, etc.



Micro Informatique Diffusion
60, AVENUE DE LA RÉPUBLIQUE
75011 PARIS - TÉL. : 357.83.20 +

Ouvert de 9 h à 12 h et de 14 h à 19 h. Sauf le dimanche.

Pour plus de précision cerchez la référence 103 du « Service Lecteurs »

concurrente

LOCASYST



DISTRIBUTEUR NORTH-STAR

Ordinateurs : North-star, Dynabyte.

Ecrans : Televideo, Qwerty & Azerty.

Disque lourd (Multiposte) : North-star, Corvus jusqu'à 72 Mb.

Logiciel Micropro : Word-star, Data star, Super sort.

Logiciel Locasyst : Comptabilité, Paye & Salaires, Inventaires, Créateur de fichiers.

A.C. SYSTEMES

3, rue Viala,
79000 Niort
(49) 24.55.19

BUREAUVISION INFORMATIQUE

117, rue de la Croix-Nivert
75015 Paris
533.53.86

C.A.P. INFORMATIQUE

10, rue de Montezat
64010 Pau Cedex
(59) 27.54.98

C.V.S. INFORMATIQUE

5, rue Dormoy
42000 Saint-Etienne
(77) 23.43.96

Ets TOULOKOWITZ

44, rue Voltaire, B.P. 4018
10013 Troyes Cedex
(25) 79.13.01

FORMATTEL

10, rue Jean-Jaurès
Tour Letwin
92806 Puteaux La Défense
775.69.40

Distributeurs régionaux

GEMO INFORMATIQUE

Voie n° 8 - Z.I. de Jarry
B.P. 54
97152 Pointe-à-Pitre Cedex
(590) 26.60.18

INTERDIS

94, avenue de la Croix-du-Sud
34000 Montpellier
(67) 52.28.63

MIDI MICRO-INFORMATIQUE

26, rue Maurice-Fonvielle
31000 Toulouse
(61) 23.68.50

NORD-MICRO-SYSTEMES

25, rue Saint-Jacques
59800 Lille
(20) 31.08.96

PYTHAGORE INFORMATIQUE

588, rue Haute
45590 St-Cyr-en-Val
(38) 65.22.52

SEREC

36, rue de Metz
54000 Nancy
(08) 332.12.60

SOPROGA BOOLE INFORMATIQUE

14, rue Le Corbusier
13090 Aix-en-Provence
(42) 59.14.83

Locasyst 183, rue de Courcelles, 75017 Paris. Tél. : 622.42.58

Pour plus de précision cerchez la référence 104 du « Service Lecteurs »