

**INITIATION**  
**AUX MINICALCULATEURS**  
**ET MICROPROCESSEURS**

PRATIQUE DE L'INFORMATIQUE



**INITIATION**  
**AUX MINICALCULATEURS**  
**ET MICROPROCESSEURS**

par

**Arpad BARNA**

*Hewlett-Packard Laboratories  
Palo Alto-Californie*

et

**Dan I. PORAT**

*Stanford Linear Accelerator Center  
Stanford University-Stanford-Californie*

TRADUIT DE L'ANGLAIS PAR

**Jean-Marc KERILIS et Christian SAGUEZ**

*Ingénieurs des Arts et Manufactures*

ÉDITIONS EYROLLES

61, boulevard Saint-Germain - 75005 PARIS

1977

BIBLIOTHEQUE DU CERIST

*Traduction autorisée de l'ouvrage anglais*

**Introduction to Microcomputers and Microprocessors**

©1976 by John Wiley and Sons Inc. (Tous droits réservés)

## AVANT-PROPOS

*Le nombre sans cesse croissant de types de minicalculateurs et de microprocesseurs a conduit à une grande diversité dans leurs applications. Ainsi la création de systèmes utilisant un microprocesseur demande la connaissance de plusieurs domaines tels que la logique, les systèmes digitaux, l'architecture d'un calculateur, la programmation et à un degré moindre la conception des circuits et la technologie des semiconducteurs. Ce livre d'initiation est écrit pour les personnes qui n'ont pas de connaissances précises dans ces domaines et qui désirent en acquérir suffisamment pour faire une bonne utilisation des minicalculateurs et des microprocesseurs.*

*Nous avons procédé en trois étapes. Les trois premiers chapitres donnent une vue générale du hardware et du logiciel élémentaires. Les cinq chapitres suivants détaillent le fonctionnement du minicalcuteur et le dernier chapitre présente des compléments. Il y a un minimum de renvois entre les différents chapitres pour que le lecteur puisse sans problème lire seulement les parties qui l'intéressent.*

*Les 120 exemples et problèmes inclus dans le texte rendent le livre particulièrement apte à une étude personnelle, en fournissant une base solide pour la compréhension des caractéristiques de la plupart des minicalculateurs. Les réponses à certains problèmes sont données à la fin du livre.*

## TABLE DES MATIÈRES

AVANT-PROPOS . . . . .	5
Liste des abréviations utilisées . . . . .	9
<b>1. Introduction</b> . . . . .	<b>11</b>
<b>2. Structure de base des minicalculateurs et microprocesseurs</b> . . . . .	<b>14</b>
2.1. Le bloc entrée-sortie . . . . .	14
2.2. L'unité centrale de traitement . . . . .	16
2.3. La mémoire centrale . . . . .	19
2.4. Les microprocesseurs . . . . .	19
<i>Problèmes</i> . . . . .	20
<b>3. Introduction aux techniques de programmation</b> . . . . .	<b>21</b>
3.1. Instructions en langage machine. . . . .	21
3.2. Instructions en langage d'assemblage . . . . .	23
3.3. Langages évolués de programmation . . . . .	25
3.4. Sous-programmes . . . . .	26
3.5. Organigrammes . . . . .	27
<i>Problèmes</i> . . . . .	29
<b>4. Entrées et sorties.</b> . . . . .	<b>30</b>
4.1. Instructions d'entrée-sortie . . . . .	30
4.2. Bloc entrée-sortie . . . . .	30
4.3. Interruptions . . . . .	37
4.4. Accès direct en mémoire . . . . .	38
<i>Problèmes</i> . . . . .	38
<b>5. Opérations arithmétiques</b> . . . . .	<b>39</b>
5.1. Systèmes de numération . . . . .	39
5.2. Représentation des nombres en octal et hexadécimal . . . . .	46
5.3. Codage . . . . .	49
5.4. Représentation en virgule flottante et règles de calcul . . . . .	52
<i>Problèmes</i> . . . . .	53

<b>6. Circuits arithmétiques et logiques</b> . . . . .	55
6.1. Additionneurs et soustracteurs . . . . .	55
6.2. Multiplicateurs et diviseurs . . . . .	61
6.3. Accumulateur et unité arithmétique et logique . . . . .	62
<i>Problèmes</i> . . . . .	64
<b>7. La mémoire centrale.</b> . . . . .	65
7.1. Mémoires à semiconducteurs . . . . .	65
7.2. Structure de la mémoire . . . . .	70
7.3. Registre de décalage . . . . .	71
7.4. Registres auxiliaires . . . . .	71
7.5. Circuit de restauration pour les RAM MOS dynamiques . . . . .	72
7.6. Mode d'adressage . . . . .	73
7.7. Adressage indirect . . . . .	76
<i>Problèmes</i> . . . . .	77
<b>8. L'unité de commande</b> . . . . .	79
8.1. Mise en séquence . . . . .	79
8.2. Synchronisation . . . . .	85
8.3. Chemins de données et structure des canaux . . . . .	86
8.4. Microprogrammation . . . . .	88
8.5. Schéma fonctionnel d'un minicalcateur . . . . .	89
<i>Problèmes</i> . . . . .	90
<b>9. Compléments</b> . . . . .	92
9.1. Assembleurs . . . . .	92
9.2. Chargeurs . . . . .	95
9.3. Structures des données . . . . .	96
9.4. Liens de sous-programme . . . . .	100
9.5. Simulation . . . . .	103
9.6. Partage hardware . . . . .	103
9.7. Système d'exploitation . . . . .	104
<i>Problèmes</i> . . . . .	105
<b>INDEX ALPHABÉTIQUE</b> . . . . .	111

## LISTE DES ABRÉVIATIONS

ALU	unité arithmétique et logique (arithmetic-logic unit).
ASCII	American standard code for information interchange.
BCD	décimal codé binaire (binary-coded decimal).
CPU	unité centrale (central processor unit).
DMA	accès direct en mémoire (direct memory access).
FET	transistor à effet de champ (field-effect transistor).
I/O	entrée-sortie (input-output).
MAR	registre d'adresse mémoire (memory address register).
MDR	registre de données mémoire (memory data register).
MOS	métal-oxyde-silicium (metal-oxyde-silicon).
MPX	multiplexeurs d'entrée et de sortie (I/O multiplexer).
MUX	multiplexeurs d'entrée et de sortie (I/O multiplexer).
PC	compteur ordinal (program counter).
PLA	zone logique programmable (programmable logic array).
pROM	mémoires mortes programmables (programmable read-only memory).
RAM	mémoire à accès sélectif (random-access memory).
ROM	mémoires mortes (read-only memory).

## INTRODUCTION

L'une des étapes les plus importantes dans l'évolution des calculateurs digitaux a été l'introduction des *calculateurs à programme mémorisé* (1). Contrairement à un boulier ou à un calculateur actionné manuellement, la suite des opérations dans un calculateur à programme mémorisé est commandée par un programme interne.

**Exemple 1.1.** La circulation automobile à l'intersection d'une route principale et d'une route secondaire est régulée par un contrôleur de circulation qui a une période de 60 secondes. Les feux de la route principale sont verts pendant 30 secondes puis oranges pendant 5 secondes et rouges pendant 25 secondes. Aussi simple qu'il est, ce contrôleur de circulation peut être considéré comme un calculateur à programme mémorisé.

Cependant, suivant la conception habituelle, un calculateur à programme mémorisé a une possibilité supplémentaire : il est capable de choisir entre différentes parties de son programme. Un tel choix, ou *prise de décision*, peut être commandé par le résultat de calculs divers ; il peut être aussi commandé par l'information reçue d'une *unité d'entrée* du calculateur.

**Exemple 1.2.** Le contrôleur de circulation de l'exemple 1.1 est développé pour recevoir deux capteurs de véhicule connectés comme unités d'entrée du calculateur. Les capteurs placés sur la route secondaire indiquent si un véhicule attend le changement de couleur du feu. A la fin des 30 secondes de feu vert de la route principale le contrôleur interroge les capteurs et ne change la couleur des feux que si un véhicule attend sur la route secondaire.

---

(1) Les termes nouveaux sont en italique.

Les calculateurs digitaux à programme mémorisé se sont largement développés au cours des deux dernières décennies. La première cause en était les progrès technologiques tels que l'introduction des transistors qui envahissent maintenant toutes les parties du calculateur, les améliorations des éléments de stockage utilisés dans la *mémoire*, la fiabilité accrue des *unités périphériques* électromécaniques et l'utilisation toujours croissante de *circuits intégrés*. Les calculateurs digitaux d'aujourd'hui se composent d'*ordinateurs spécialisés* conçus pour un seul usage et d'*ordinateurs à usages multiples* utilisés dans différents domaines tels que le contrôle, le traitement de données et les calculs scientifiques.

En parallèle avec la fiabilité accrue, les possibilités de calcul et la facilité d'utilisation des ordinateurs à usages multiples survinrent les miniordinateurs à usages multiples qui, bien que limités du point de vue possibilités de calcul, étaient plus petits et moins coûteux. En grande partie à cause de leur prix plus bas, les miniordinateurs s'imposèrent souvent dans ce qui était initialement le domaine exclusif des ordinateurs spécialisés. Le dernier fossé séparant les ordinateurs à usages multiples des contrôleurs et des ordinateurs spécialisés a été comblé par le dernier et le plus petit ordinateur à usages multiples, le *minicalcateur*.

Les premiers minicalculateurs étaient des calculatrices. A l'heure actuelle les minicalculateurs remplacent aussi et complètent beaucoup de petits ordinateurs et d'ordinateurs spécialisés, en particulier les *contrôleurs câblés* spécialisés.

**Exemple 1.3.** Des aiguillages sont installés dans un système de transit rapide. Un aiguillage particulier est installé pour chaque « bloc » de la voie, cet aiguillage guidant les trains entrant et sortant du « bloc ». Dans l'étude initiale chaque aiguillage utilisait un contrôleur câblé spécialisé; cependant, à cause des « cas particuliers » correspondant aux différentes parties de la voie, les contrôleurs ne pouvaient pas être identiques. Pour la réalisation pratique, les contrôleurs câblés ont donc été remplacés par des minicalculateurs et les cas particuliers sont traités par des programmes adéquats.

La simplicité et le coût réduit qui permettent une large utilisation des minicalculateurs ont aussi pour conséquence une complication des techniques de programmation par rapport aux ordinateurs. En outre, le côté *hardware* est souvent plus lié au côté programmation ou *logiciel* dans un minicalcateur que dans un ordinateur. Donc, en dépit du fait que le

travail est souvent divisé entre des spécialistes hardware et des spécialistes logiciel, la création d'un système qui utilise un miniordinateur réclame une connaissance élémentaire des deux domaines. Pour cette raison ces deux domaines sont imbriqués dans la plus grande partie de ce livre, de façon à favoriser une introduction équilibrée au hardware et au logiciel des minicalculateurs.