

Réalisez votre micro-ordinateur « Micro-Systèmes I »

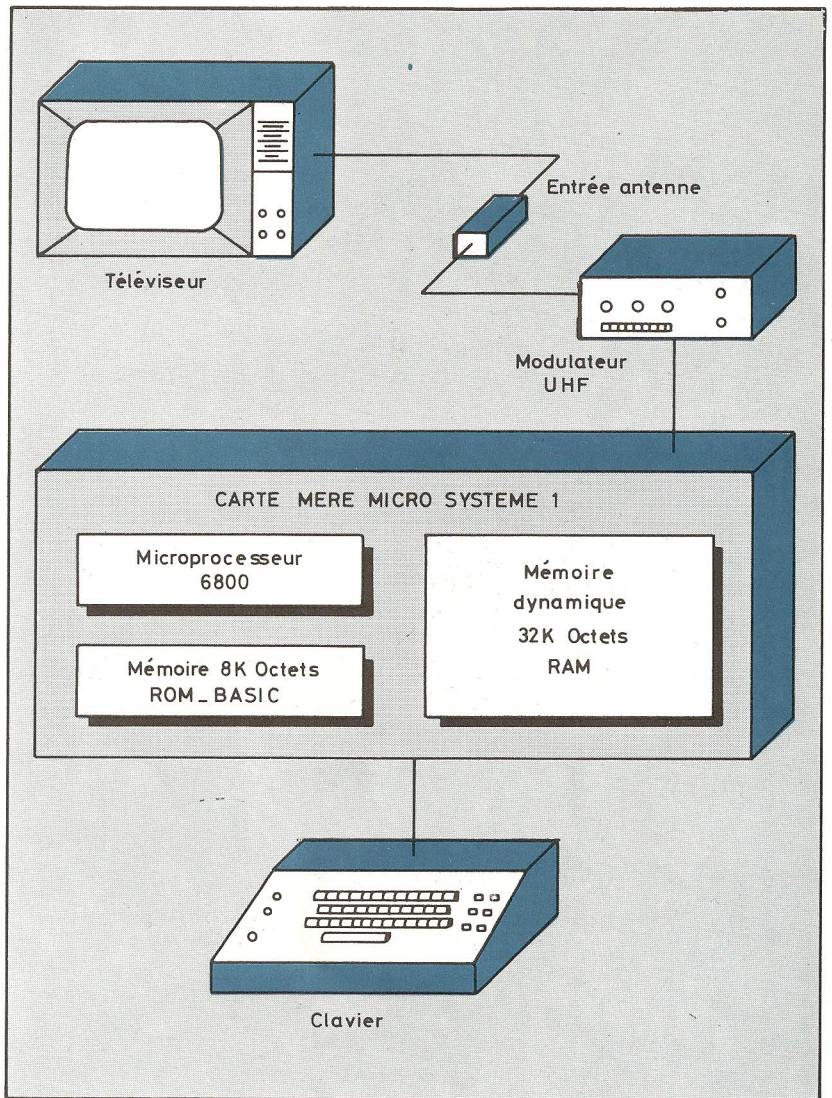
Outre le dialogue permanent que la revue veut instaurer entre ses lecteurs et les constructeurs, la vocation principale de Micro-Systèmes est de vous proposer et de vous assister dans la réalisation complète de micro-ordinateurs dont nous avons préalablement testé les performances. Entendons-nous sur la sémantique : certains considèrent comme « micro-ordinateur » tout dispositif capable de faire fonctionner des programmes, même s'il ne dispose que d'un langage-machine. Partant de là, tous les kits d'évaluation et systèmes de mise au point pourraient répondre à cette définition.

En fait cette appellation ne doit recouvrir que des machines disposant d'un langage évolué donc d'une mémoire vive capable de contenir des compilateurs ou interpréteurs de ces langages.

Lorsqu'à présent nous vous proposons la réalisation d'un micro-ordinateur, c'est à cette deuxième définition que nous nous référons. Nous en avons confié l'étude au Groupe Protéus International qui durant trois mois s'est appliqué à le rendre simple mais performant, facilement exécutable car exempt de tout réglage.

Ce micro-ordinateur est programmable en « Extended-BASIC », langage cher aux amateurs, car il s'apprend en quelques heures.

Un BASIC tel que celui que comporte « micro-systèmes 1 » permet les entrées/sorties en huit bits parallèles et la manipulation binaire d'appareillage électronique, les calculs à neuf décimales, le traitement de chaînes de caractères et de textes *, les appels de sous-programmes écrits par l'utilisateur directement en langage-machine, la programmation de jeux, la gestion, etc.



Le micro-ordinateur « Micro-Systèmes I » permet le dialogue homme-machine grâce à un clavier ASCII et à un téléviseur domestique.

Comment réaliser son micro-ordinateur ?

A partir du moment où vous décidez de vous initier aux micro-processeurs, la construction d'un micro-ordinateur devient presque une finalité vers laquelle chaque lecteur a de bonnes raisons de vouloir prétendre.

Est-il possible de se lancer dans cette aventure seul ? Regardons ensemble les étapes qui mènent à l'aboutissement de ce projet.

Premier scénario

Donc, vous êtes chez vous, tout seul, la micro-informatique vous passionne et vous aimeriez avoir

votre micro-ordinateur construit selon votre étude personnelle.

La lecture de Micro-Systèmes vous a maintenant donné toute la confiance dont vous aviez besoin pour faire de ce rêve une réalité.

Vous prenez une feuille de papier, un crayon et votre matière grise commence à vous dicter les fonctions que devra comprendre votre système. De là vous recherchez autour de quel microprocesseur il sera organisé. A cela vous ajoutez quelques mémoires, un clavier, un affichage sur écran vidéo et en option un lecteur de cassette. Jusqu'ici tout va bien.

Les composants étant sélectionnés, il vous reste à étudier l'implan-

tation de la carte-mère. Ça devient difficile mais à force de patience on peut s'en sortir. Elle sera nécessairement du type double face à trous métallisés.

Pour l'approvisionnement de vos composants, nous vous faisons confiance. Vous pourrez toujours trouver un endroit où les acheter au meilleur prix. Mais attention, plutôt que de privilégier les composants bon marché, il est de loin préférable, pour ce type de réalisation, de rechercher les composants de qualité.

A présent, ça y est tout est là. C'est gagné...

Pas tout à fait, car pour en arriver à ce stade nous venons d'élu-

* Une chaîne de caractères et de textes représente un nom, un mot code, une phrase que la machine traite sous forme de suite d'octets binaires.

der un certain nombre de problèmes difficiles à surmonter. Lesquels ?

— La première grande difficulté apparaît dans l'étude de l'implantation de la carte-mère : faisable mais très laborieux.

— Deuxièmement sa réalisation pratique n'est pas envisageable avec les moyens dont dispose un amateur, de plus fabriquée à l'unité elle devient un composant de luxe.

Il faut ajouter à cela que de sa bonne exécution, dépend largement le fonctionnement de votre micro-ordinateur.

— Troisièmement, pour mettre au point un langage BASIC il faut pouvoir disposer d'un ordinateur du type IBM 360.

— Devant tous vos efforts, un quatrième problème va surgir : comment programmer la mémoire contenant l'interpréteur BASIC ?

Il faut d'abord définir le programme pour ensuite le stocker en mémoire.

Il est évident que ce scénario se terminera bien tristement.

Regardons le deuxième scénario

C'est simple, l'idée, votre idée, reste la même, mais à présent nous avons décidé de vous venir en aide avec « micro-systèmes I ».

« micro-systèmes I » n'est pas un kit.

« micro-systèmes I » est une **étude personnelle** assistée.

Que comporte cette assistance ?

a) La première chose que nous vous offrons c'est l'étude, nous l'avons faite faire pour vous par Protéus International.

Vous découvrirez la notice de ce système tout au long de cette rubrique.

b) La carte-mère : spécialement réalisée pour « micro-systèmes I », la revue sera le dépositaire exclusif de ce composant passif qui, marqué de l'estampille « micro-systèmes I », vous sera cédé au prix coûtant de 300 F (TTC).

Comportant au total 1400 trous métallisés, cette carte a été réalisée selon des normes professionnelles

avec comme objectif fondamental, l'obtention d'un taux de déchet nul. Pour parvenir à ce résultat l'implantation a été confiée à un ordinateur...

Ainsi à l'image des matériels professionnels, cette carte est conçue pour supporter l'ensemble des composants nécessaires, excepté le modulateur UHF en option 44 F (TTC) (signal vidéo-complexe sur la carte-mère) et le clavier ASCII encodé. Type 1648 de Key-Tronic.

Cette carte reçoit les huit bits parallèles provenant de ce clavier et fournit à un poste de télévision équipé d'un modulateur UHF, ou disposant d'une entrée vidéo, le signal nécessaire à la visualisation de 16 lignes de 64 caractères de texte ou graphiques.

c) Une mémoire morte ROM de 64 K bits (8 K - octets) contenant le BASIC, dont la fabrication relève des toutes dernières techniques d'intégration et pour laquelle le prix de vente est de 980 F (TTC).

Cette mémoire pourra être remplacée par quatre mémoires pré-programmées d'EPROM'S.2716 (2 K octets chacune) disposées sur une plaque de circuit imprimé terminé par un connecteur de 24 broches pouvant s'insérer dans le support réservé à la ROM-BASIC.

Cet ensemble de mémoires pré-programmées EPROM'S est disponible au même prix que la ROM-BASIC « micro-systèmes I ».

Toutefois, il est à remarquer que cette dernière s'alimente uniquement en + 5 V alors que la batterie d'EPROM'S nécessite les tensions suivantes : + 5 V ; - 5 V ; + 12 V.

Maintenant que les quelques points, qui jusque-là demeuraient difficilement accessibles même aux amateurs chevronnés, ont trouvé une solution efficace et économique, il ne reste plus qu'à passer aux choses sérieuses.

Le reste des composants est classique et possède plusieurs sources d'approvisionnement.

Pour les mémoires dynamiques, il a été choisi des SV6726 pour leur grande rapidité. D'une capacité de 16 K x 1 bits, elles seront montées par quantité correspondant à des multiples de huit.

Le prix de l'ensemble des composants se situera dans la fourchette des 3000 à 4000 F (TTC) pour la version complète qui correspond à un micro-ordinateur dont les performances le placent dans le peloton de tête des systèmes actuellement disponibles sur le marché français.

« Micro-systèmes I » fonde sa supériorité sur la puissance de son langage et sur sa puissance de calcul.

Votre investissement initial pourra se porter sur l'acquisition de la carte-mère, tous les autres éléments pouvant être achetés au fur et à mesure de la progression de votre étude « micro-systèmes I ».

Pour la mémoire BASIC, qu'il s'agisse de la mémoire ROM ou de la batterie d'EPROM'S 2716 pré-programmées, ces deux composants constituent des fournitures spécifiques approvisionnées par Protéus International.

Un micro-ordinateur ! Pour quoi faire ?

Si dans un premier temps la motivation qui vous anime est l'étude et la réalisation de « micro-systèmes I », il n'en demeure pas moins vrai que ceci ne constitue pas une fin en soi et qu'en fait ce que vous visez réellement c'est l'utilisation de ce micro-ordinateur en réponse à des problèmes spécifiques ou généraux que vous rencontrez.

Certains diront : « oui mais pourquoi un micro-ordinateur plutôt qu'un système d'évaluation ou de mise au point de microprocesseurs travaillant en langage machine ou assembleur ? » Tout simplement parce qu'il les remplace avantageusement.

En électronique

A part les applications militaires ou industrielles, où le problème du gain, recherché au niveau de la microseconde se pose avec acuité un utilisateur disposant d'un peu de temps pour son application (il s'agit ici d'une vue de l'esprit car ce temps est microscopique et imper-

ceptible à l'échelle humaine) aurait avantage à calculer les paramètres de son expérience et à les programmer en langage évolué.

Ceci est concevable pourvu que la liaison avec les périphériques soit assurée comme dans tous les systèmes à microprocesseurs et soit accessible par le langage.

Prenons un exemple pour mieux comprendre : un ingénieur doit calculer « U » (tension crête), « I » (courant crête) dans une expérience sur les circuits fonctionnant en impulsions.

Il lui faut d'abord multiplier des données binaires de huit bits issues d'un convertisseur analogique-digital par des facteurs d'échelle pour ensuite afficher la valeur de l'impédance $Z = U/I$.

En plus de ces multiplications et de cette division binaire, les résultats doivent être affichés en décimal d'où une conversion finale Binaire-BCD.

Pour ce faire, il commence par travailler en binaire sur un kit d'initiation. A mi-chemin dans son projet, une discussion le met sur la voie d'un micro-ordinateur du type de celui que nous vous proposons d'étudier.

Ainsi, après l'acquisition de ce micro-ordinateur, son application se traite en quelques heures : avec saisie des données en binaire, calculs en BASIC et affichage sur écran vidéo au lieu d'afficheurs à sept segments. Ainsi dispose-t-il enfin d'un moyen **pratique, efficace et rapide** de travail, car la programmation et l'arithmétique hexadécimale binaires ne sont pas toujours des plus attrayantes

Dans la vie de tous les jours

« micro-systèmes I » constitue un véritable **outil informatique**. En effet, malgré le fait qu'il soit édifié sur des composants simples et limités en nombre, cet ordinateur ne mérite absolument pas ce préfixe au regard de sa puissance de calcul.

Il offre la possibilité de programmer : la gestion de son (ou ses) compte (s) en banque (s), les achats ménagers à effectuer pour la semaine à venir, des jeux éducatifs ou distractifs visualisés sur télévision, bref toutes ces applica-

tions qui ont déjà fait l'objet de nombreuses émissions télévisées.

En plus de l'aspect « Jeu », sa programmation offre des possibilités de « Gestion ».

Ainsi « micro-systèmes I » est tout aussi bien destiné aux « PME » qui ont de bonnes raisons de ne pas investir dans la grosse informatique compte tenu de la dimension de leurs problèmes. (On ne prend pas un marteau-pilon pour briser une noix).

Pour une mise de fonds minimum, il gère toute une comptabilité et peut assurer un travail de secrétariat tel que l'impression de **courrier** répétitif, remplaçant de la sorte certaines machines de gestion de textes très onéreuses. Il vous apporte la solution que vous attendiez pour composer automatiquement vos **numéros de téléphone**, il peut même, après avoir obtenu votre correspondant transmetteur du courrier, des données, etc.

Ce micro-ordinateur peut tenir une **caisse**, traiter des **calculs** qu'on lui soumet en un langage clair et dont tout le déroulement est suivi en permanence sur l'écran de votre téléviseur.

Mais attention, ceci n'est en rien comparable avec les calculatrices programmables les plus sophistiquées.

Ajoutons à tout cela qu'il est capable d'imprimer ses résultats sous forme de tableaux ou sous forme de graphique si on le lui demande.

A qui nous adressons-nous ?

Nos efforts s'adressent à la grande majorité, enthousiaste et à la fois surprise par l'ampleur de ce phénomène qui conduit à une démocratisation de l'informatique.

La construction du « micro-systèmes I » pourrait être perçue comme un simple exercice de câblage nécessitant un fer à souder, de la soudure, des composants et le circuit imprimé double face à trous métallisés. Certes, il est permis de s'en tenir à cette solution pour qui n'est pas familiarisé avec

l'électronique, mais rappelons qu'il s'agit d'abord d'entreprendre l'étude de ce micro-ordinateur pour lequel nous serons là pour vous assister.

D'un autre côté être ingénieur électronicien ne constitue pas une nécessité car beaucoup plus que la réponse transitoire en impulsions des composants, il importe de se soucier de leur sens d'implantation.

La récompense finale de tout votre travail sera matérialisée par l'apparition sur l'écran de votre téléviseur du « Ready » du BASIC.

Quant à la notice de montage qui commencera dans le numéro prochain, elle sera accompagnée des principes de fonctionnement, des particularités du micro-processeur et des mémoires utilisées dans ce système.

Ce qui veut dire, qu'à l'instar de l'équipe qui a conçu « micro-systèmes I », vous découvrirez les raisons qui ont prévalu dans leur choix ainsi que la justification de l'architecture de ce micro-ordinateur.

En résumé, nous traiterons de :

A : la présentation de la fiche technique du micro-ordinateur.

B : la description de son organisation générale accompagnée d'une analyse du matériel (hardware).

C : la réalisation pratique de la carte-mère.

D : des explications détaillées sur le fonctionnement de ce micro-ordinateur ainsi que la réponse aux questions suscitées par son montage.

E : la programmation de nombreuses applications avec des programmes en Basic.

A ce sujet le « volume de la sphère », la « visualisation de courbes par points » qui vous ont été présentées précédemment au chapitre « programmation » sont des programmes qui « tournent » déjà sur « micro-systèmes I ».

Fiche technique de « MICRO-SYSTEMES I »

HARDWARE

Présentation :

Une plaque-mère, reliée à un clavier encodé ASCII et un écran TV.

Processeur :

6 800 (SFF 96 800) ; 0,894 MHz (2 à 4 cycles/instruction en moyenne).

Rom :

Standard BASIC-8k résident.

Option assembleur, éditeur de texte sur cartes enfichables.

Ram :

32 k octets sur supports, soudée sur la plaque-mère.

16 k octets en option sur des cartes enfichables.

Entrées/Sorties :

Standard TTY, entrée/sortie série TTL, en boucle de courant 20 mA et standard RS 232, trois versions résidentes en permanence sur la plaque-mère.

Minicassette en standard Kansas City (1 200 Hz × 4 cycles pour un zéro et 2 400 Hz × 8 cycles pour un « 1 » logique). Options 4 cartes enfichables sur supports bénéficiant des bus tamponnés ainsi que d'un décodage d'entrée/sortie, permettant d'installer des PIA-s.

Option Floppy Disk :

Un contrôleur de DMA sur la plaque-mère et une carte sur support permettent de gérer 3 minidisquettes Pertec double face et double densité (2,6 Mbits de capacité totale chacune). DOS chargé en RAM (16 k octets) à partir de la disquette.

Alimentation :

+ 5 V ± 10 % ; 3 A

+ 12 V ± 10 % ; 1,5 A
- 12 V ± 10 % ; 1 A

Non fournie. Une alimentation résidente sur la plaque, dérivée de - 12 V.

Clavier ASCII :

Un côté « B » de PIA (PB₀-PB₇, CB₁, CB₂) reçoit les 7 bits, un « Strobe » reçu sur le front descendant, un « Break » et « Here Is », d'un clavier encodé ASCII-standard.

Ecran Vidéo

La plaque fournit un signal Vidéo-Complex pouvant gérer un moniteur vidéo ou un écran TV ordinaire muni d'un modulateur U.H.F.

Le circuit de visualisation de caractères graphiques, résident sur la plaque possède sa propre mémoire d'écran. Les caractères sont « fondus » dans une EPROM-2708 et peuvent être modifiés à souhait par l'utilisateur. La version standard contient le jeu de caractères ASCII.

SOFTWARE

Structure de l'espace adressable

Les divers éléments occupent les adresses indiquées dans le tableau I.

On constate une occupation continue des premiers 32 k octets, de 0000 à 7 FFF. Ces adresses contiennent la RAM dynamique logée intégralement sur la carte-mère.

Les 16 k octets suivants, allant de 8 000 à BFFF sont réservés à un coupleur d'entrée/sortie parallèle (le « PIA »-MC 6820) et série (l'« ACIA »-MC 6850) et à la mémoire où les coupleurs d'entrée/sortie additionnels qui prendront place sur des cartes enfichables.

BASIC

La mémoire ROM-8 k octets contient un BASIC étendu permettant toutes les fonctions du FULL-BASIC plus des fonctions spéciales. Nous les avons résumées dans le tableau II.

Tableau I

FFFF	ROM BASIC ou Amorce pour le chargement du DOS
E000	
DFFF	8 k-ROM BASIC
C000	
BFFF	LIBRE
8000	
7FFF	32 k de RAM sur la plaque-mère
000	

TABLEAU II

Le jeu d'instructions BASIC de MICRO-SYSTEMES I

Commandes :

RUN : exécution immédiate
 NEW : efface la mémoire de programme
 CONT : après un STOP, permet de continuer
 TRACE ON : démarre l'exécution en pas à pas en BASIC
 TRACE OFF : arrête le pas à pas
 FORT : définit le port (canal, ou ligne) d'E/S
 LOAD : charge le contenu d'une cassette, disquette, etc. en mémoire.
 APPEND : comme LOAD sauf qu'il n'efface pas (NEW) l'ancienne zone
 pgme
 SAVE : sauvetage d'un pgme sur cassette, ruban perforé ou disquette
 ER : rend la main au moniteur, DOS, etc. s'ils existent
 LIST : affichage de la liste d'instructions du programme
 INCR : numérotage automatique par pas de INCR des lignes
 LINE : permet de réserver une ligne de ligne caractères
 DIGITS : choisit le nombre de décimales désirées
 SPEED : choix, par programme de la fréquence d'horloge-système
 STRING : introduction d'une chaîne de caractères (plus de 18)
 CTRL C : STOP de l'exécution
 CTRL X : effacement d'une ligne en cours d'entrée
 CTRL O : effacement du dernier caractère tapé
 RUBOUT : numérotation automatique

Directives :

REM : remarque-ligne non-exécutable, de commentaire
 LET : « soit » affectation (facultative) LET X = Y + Z
 PRINT : « imprimer » utilisé pour l'impression de texte et résultats
 INPUT : « entrer » utilisé pour l'entrée de texte et données
 GOTO : « va en » saut inconditionnel
 END : fin du programme
 STOP : arrêt
 IF THEN : saut logique ou exécution conditionnelle
 FOR : début de boucle
 NEXT : fin de boucle
 STEP : pas de boucle
 GOSUB : saut à un sous-programme
 RETURN : retour d'un sous-programme

ON GO : aiguillage inconditionnel
 ON GOSUB : aiguillage vers des sous-programmes
 DIM : tableau
 DATA : introduction de données
 READ : lecture des données précédentes
 RESTORE : manipulation du pointeur du READ
 DEF : définition des fonctions propres du programmeur

Fonctions :

SQR : racine carrée $SQR(9) = \sqrt{9}$
 ABS : valeur absolue
 INT : valeur entière
 SGN : signe : positif = 1 ; négatif = -1 ; nul = 0
 RND : produit des nombres aléatoires
 DLOG : logarithme décimal
 LOG : logarithme naturel
 EXP : exponentielle
 SIN : sinus
 COS : cosinus
 TAN : tangente
 ATAN : arctangente
 DEG : conversion radians-degrés
 RAD : conversion degrés-radians
 TAB : tabulation - déplace le pointeur
 POS : position du pointeur
 LEN : longueur d'une chaîne de caractères
 VAL : valeur du nombre = chaîne de caractères
 STR\$: variable numérique dans une chaîne de caractères
 ASC : donne la valeur décimale d'un caractère
 CHR\$: chargement d'un caractère
 LEFT\$: nombre de caractères, partant de la gauche
 MID\$: nombre de caractères, au milieu
 RIGHT\$: nombre de caractères, partant de la droite
 PEEK : lecture d'un mot mémoire (en décimal)
 POKE : écriture dans une case mémoire
 USEP : appel de sous-programmes en langage machine
 FN : création de fonctions

Organisation générale

Générateur d'horloge

Le synoptique de la figure 1 donne une idée de l'architecture d'un système micro-ordinateur classique. L'unité Centrale représente bien peu de chose. Elle reçoit ses signaux d'horloge de la part d'un circuit qui génère en même temps les signaux garantissant le D.M.A. (Direct Memory Access), le rafraîchissement de la mémoire dynamique et crée la fréquence d'horloge d'un diviseur-générateur de bauds, nécessaire aux transmissions série.

Les amplificateurs de ligne

Le synoptique ne fait pas apparaître les amplificateurs, ou « tampons » de ligne dont est munie l'Unité Centrale.

Ses bus de données et d'adresses ainsi amplifiés, peuvent supporter une bonne dizaine de charges TTL, circuits sur la carte-mère, ou sur les supports des cartes enfichables de l'utilisateur.

La ROM-BASIC

L'Unité Centrale est reliée à une mémoire de programme-ROM qui contient en codes-machine les programmes nécessaires à la gestion de la carte (sortie

vers un magnétophone mini-cassette ou TTY, entrées d'interruption, ou de données, etc.) ainsi que les programmes de calcul et d'interprétation des lignes BASIC. Au niveau du synoptique, certains codes-parallèle huit bits arrivant du coupleur PIA, ou codes-série venant de l'ACIA*, constituant autant de caractères de la syntaxe du BASIC ou des chiffres en décimal, sont stockés, ligne par ligne. Une ligne représente pour la machine ni plus ni moins qu'une suite de caractères ASCII, représentés en binaire et se terminant par le code « Retour Chariot » (RC). Le programme interpréteur BASIC fait entrer en mémoire un caractère après l'autre, tout en veil-

* ACIA :
 Asynchronous
 Communications
 Interface Adapter.

lant s'il n'est pas le « R.C. ». A l'arrivée de cet ordre (matérialisé par l'encodage sur le clavier d'une combinaison binaire comme les autres), le programme effectue une recherche syntaxique et sémantique de la chaîne de caractères reçue et s'il retrouve un ordre connu, il l'exécute. Par exemple, si l'on envoie l'ordre BASIC-direct :

GO TO 20 C.R.

c'est-à-dire la suite d'octets (correspondant au code ASCII) :

G = 79₁₀ = 4F₁₆
 O = 71₁₀ = 47₁₆
 blanc = 32₁₀ = 20₁₆
 T = 84₁₀ = 54₁₆
 O = 71₁₀ = 47₁₆
 blanc = 32₁₀ = 20₁₆
 CR = 10₁₀ = 0A₁₆

Les choses se passeront de la manière suivante

Le « Strobe » généré par le clavier à chaque enfoncement de touche fera saisir par la PIA les huit bits parallèles que lui présente l'encodage du clavier, cet octet sera comparé à l'encodage du code représentant le retour chariot et sera rangé dans une mémoire de ligne. Ainsi cette mémoire tiendra sur plusieurs mots consécutifs : 47, 4F,...

A l'arrivée du Retour Chariot, le programme commence à analyser le premier caractère, puis les deux premiers caractères, etc., et les compare avec une table : Est-ce un « RUN » ?, un « FOR », un... etc. En trouvant « GO TO » qui est syntaxiquement correct à l'insu d'un « GO TA » par exemple, il branchera le processeur au programme de saut inconditionnel qui contiendra un code-machine.

BRA Adresse relative
 (branchement)
 ou
 JMP Adresse absolue (saut)

Il est important de signaler que le temps passé à interpréter les ordres pour la gestion des codes (qui peuvent arriver par TTY, en série-également), est incomparablement plus long que le temps d'exécution

effective de l'instruction de saut. Néanmoins, un utilisateur humain est relativement insensible à une différence de temps d'exécution d'une dizaine de millièmes de seconde (BRA, JMP) ou d'un millième de seconde (GO TO).

Telles sont les grandes lignes du travail de la ROM-BASIC.

Le Contrôleur DMA

Pour faciliter les transferts rapides de données vers la mémoire, le système est muni d'un boîtier capable d'analyser les instructions présentes sur les bus et le cycle en cours. Ce boîtier d'accès direct à la mémoire est référencé DMA (Direct Memory Access). Les bus du microprocesseur se trouvent dans leur état actif durant les brefs moments, puis sont mis à l'état haute-impédance. Les rapports des temps de bus actifs/inactifs diffèrent d'une instruction à l'autre. Quoi qu'il en soit, pendant que le microprocesseur n'utilise pas ses bus, un autre processeur pourrait se servir de ses ressources (ROM-RAM, coupleurs, etc.) Mise à part les configurations multi-processeurs, la technique de l'accès direct à la mémoire est utilisée lors de la manipulation d'informations stockées sur les disques magnétiques. La cadence de ces transferts étant incomparablement plus rapide que celle des canaux TTY mini-cassette ou autres, il est parfois impératif de charger ces informations en mémoire sans passer par la technique traditionnelle : la PIA, va interrompre le processeur, prendre possession du bus de données et transférer un octet dans l'Unité Centrale qui le rangera à l'adresse mémoire « X ».

Ce procédé demande 20 à 50 μ s pour le rangement d'un seul octet alors que pendant ce même laps de temps, avec des mémoires faites pour des cadences du MHz on peut y ranger 20 à 25 octets lors d'un accès direct.

Les disques souples

Le système utilise un contrôleur de disques souples pouvant gérer trois minidisquettes double face et double densité. Ce sont des supports magnétiques sur lesquels les informations sont écrites bit par

bit, sans signaux d'horloge, comme c'était le cas des disques à simple densité. En effet, dans les anciens systèmes, les bits de données se trouvaient situés dans l'intervalle entre deux tops d'horloge.

La présence d'un top au milieu signifierait un « 1 » son absence - un « 0 » logique. Par les techniques modernes, liées à l'emploi des boucles de phase, on peut se passer des tops d'horloge et synchroniser néanmoins le récepteur à l'aide de préambules du type 10101010... etc., qui lancent les oscillateurs des boucles de phase à la bonne fréquence, avant l'arrivée du bloc de données. C'est un peu acrobatique, mais ça marche sans faille et l'on stocke deux fois plus d'informations sur les mêmes longueurs de piste magnétique du disque.

Le décriptage et l'écriture des informations sur le disque demandent non seulement de l'électronique, mais aussi des programmes, pour l'organisation des données en blocs, le calcul des parités et des codes correcteurs d'erreur, etc.

Ces programmes sont stockés dans une PROM que nous avons appelée AMORCE (tableau 1). Les anglo-saxons l'appellent **Bootstrap**. En initialisation, l'AMORCE permet de charger en mémoire un programme situé sur le support magnétique lui-même.

En particulier, elle charge un programme de gestion des disquettes : le D.O.S. (disk operating system). Ce programme permet de faire des recherches, des copies et transferts de fichiers, etc.

Le minicassette

Un autre support magnétique que peut utiliser le système est le magnétophone minicassette ordinaire.

Dans ce cas les informations binaires sortant en série du coupleur d'entrée/sortie série (ACIA), sont traduites en cycles sinusoïdaux, aptes à être enregistrées sur bande comme toute oscillation « musicale ». Pas besoin d'une électronique compliquée pour obtenir ce genre de formattage. Le standard Kansas City d'enregistrement utilisé, veut qu'un « 1 » logique soit représenté par huit cycles à 2400 Hz et un « 0 » par 4 cycles

figure 1

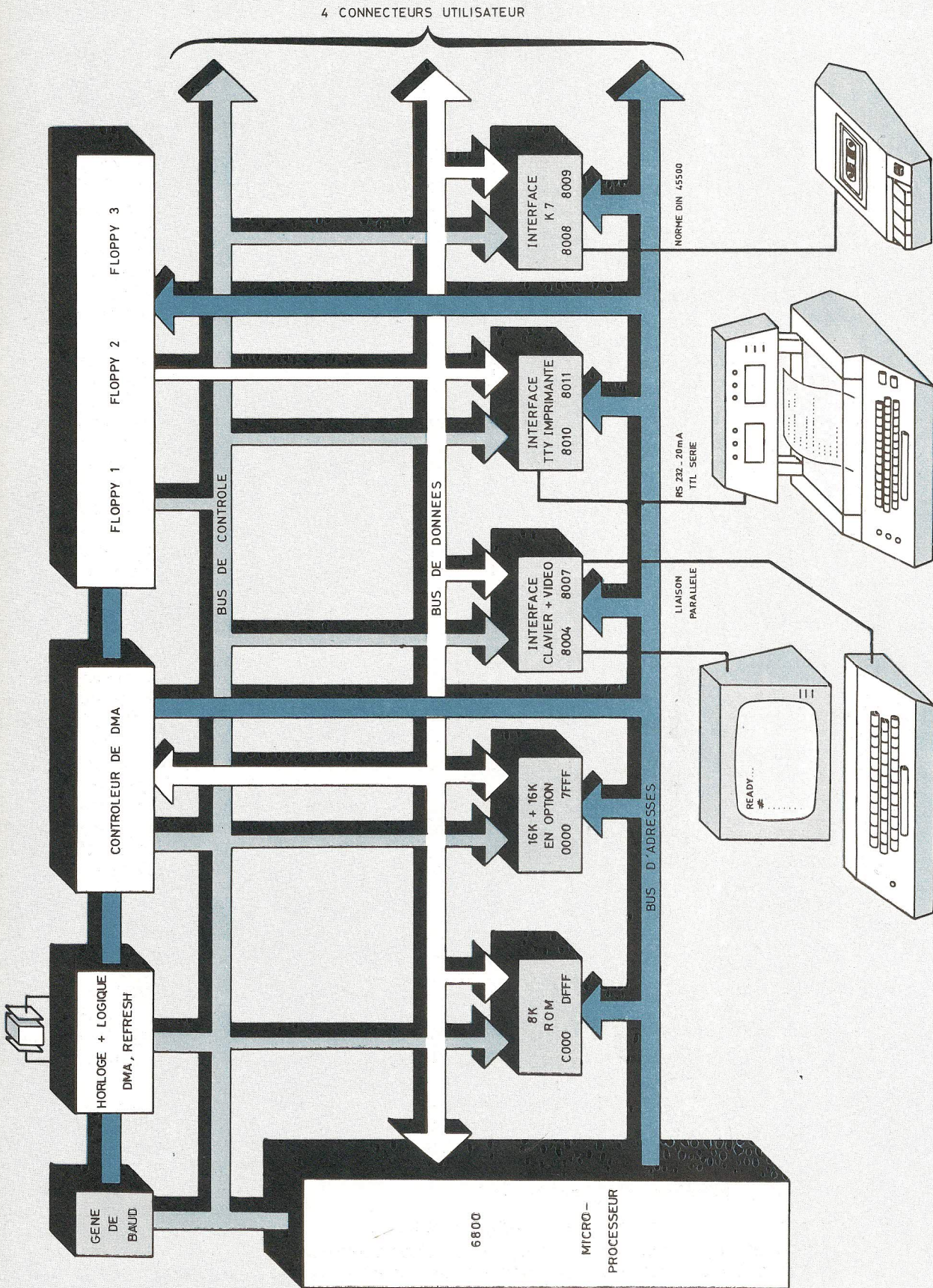


Fig. 1. - Synoptique générale du micro-ordinateur.
Le microprocesseur choisi est un 6800, notez la présence de la mémoire ROM de 8 k-octets nécessaire au langage BASIC.

de 1200 Hz. A la réception, deux filtres digitaux (à monostables) discriminent les fréquences enregistrées et les transforment en niveaux TTL logiques, envoyés dans l'entrée série de l'ACIA.

Les données à lire ou à enregistrer sont organisées en blocs de 256 octets séparés de portions de « 0 » logiques par programme, de sorte qu'un utilisateur peut avoir une idée du stade de l'enregistrement ou de la lecture en dénombrant les blocs. Le même programme organise les adresses d'enregistrement, en les situant en tête du bloc, etc.

La console de visualisation et clavier

Comme pour le magnétophone minicassette, la visualisation du texte (poste T.V. ordinaire muni d'un modulateur U.H.F. de jeux T.V., par exemple) et le clavier encodé ASCII sont pris en charge par le programme, contenu dans la ROM-BASIC. Le clavier peut

fournir huit bits parallèles accompagnés de signaux de validation « strobe » ou d'arrêt « Break », etc.

La visualisation du texte sur l'écran utilise un boîtier « Contrôleur CRT » bien connu le SF.F 96364 de la Sescosem qui rafraîchit sur l'écran le contenu d'une mémoire de 1 k octets, selon le jeu de caractères se trouvant dans une EPROM-2708. Une microprogrammation permet à ce contrôleur d'écran cathodique de gérer le texte (passage à la ligne à l'arrivée d'un retour chariot, déplacements du curseur, saut de page, rouleau, etc.) à partir de codes-caractère qu'il reçoit en parallèle.

Les supports de cartes enfichables

Ils sont au nombre de quatre et restent à la disposition de l'utilisateur qui pourra y loger jusqu'à quatre cartes enfichables obtenues par câblage ou par wrapping et dont l'adjonction lui permettra de personnaliser son micro-ordinateur en

fonction d'une application précise qu'il souhaite traiter.

Par exemple un aveugle aura toutes les facilités à utiliser « micro-systèmes I » s'il y ajoute une carte de réponse vocale et une carte d'entrée vocale.

Comment doit procéder l'industriel qui souhaite tenir tête à une raffinerie de pétrole ou automatiser un complexe industriel ? Très simple. Il lui suffit d'enficher 4 k octets d'adresses de PIA ce qui lui fait 1 000 PIA disponibles, chaque PIA pouvant être adressé et lu par le BASIC.

Ajoutons à cela qu'un PIA correspond à 16 fils d'entrée/sortie de travail, chacun de ces fils pouvant actionner une électrovanne par exemple : vous conviendrez qu'il devient aisé de s'offrir les grandes eaux de Versailles modulées par la musique d'un somptueux son et lumière.

André DORIS

Jean-José WANÈGUE

NOUVEAU
A PARIS

un FORUM MICRO-INFORMATIQUE



- Des ingénieurs pour résoudre vos problèmes de matériel et de logiciel ;
- Des produits adaptés à vos besoins (micro-ordinateurs, périphériques...)
- Un service documentation
- Des calculateurs en libre-service ;
- Des cours d'initiation, des journées à thème.

185 avenue de Choisy
75013 PARIS. Tél. 581-51-21