

**LABORATOIRE d'INFORMATIQUE THEORIQUE  
& APPLICATIONS DE MARSEILLE  
L.I.T.A.M.**

Faculté des Sciences Economiques  
Université d'Aix-Marseille II

ISSN  
0291-5413

**INFORMATIQUE  
FONDAMENTALE  
&  
APPLICATIONS**

**BULLETIN N° 32**

**Comité de  
rédaction**

**SOMMAIRE**

**E. Bianco  
R. Cusin  
S. Hilala  
P. Isoardi  
J.M. Knippel  
J.P. Lehmann  
R. Stutzmann**

- P1 ... Editorial:  
Société et organisation.  
E. Bianco
- P5 ... Le processeur de la Procédure  
Formelle.  
E. Bianco
- P21 ... Vouzzavédibisar.

**Dépositaire**

**B.U. Sc. Eco.  
Aix-Mars. II**

**Juin 1992**

Adresse postale:

Faculté des Sciences Economiques  
LITAM

14 rue Puvis de Chavannes 13001 Marseille

Tel. 91 13 96 29 - (20, 21)

### Société et organisation.

L'Homme est un nouveau riche. Avec tout ce que l'attitude qui va avec comporte comme inculture, morgue imbécile et manque de clairvoyance. Il y a six mille ans à peine, frileux et démuné, il disputait encore sa part de HLM à l'ours des cavernes. De nos jours, par l'intermédiaire de ses éléments les plus obtus, il investit les derniers recoins un peu charmants qui avaient été épargnés, en s'y installant au beau milieu. De telle sorte que son bunker bétonné qui remplace la forêt de chênes s'appelle: "la forêt de chênes". Un peu comme un amateur de peinture qui s'achèterait une toile renommée, mettons les iris de Van Gogh, la mettrait à plat sur le sol et poserait dessus son gros cul, en déclarant qu'il adore se sentir au milieu des fleurs. Un retour à la terre en quelque sorte.

Il est navrant de constater combien la dépense de véritables trésors d'intelligence, aboutit à l'accumulation de masses grotesques d'imbécillités. On accumule chez soi des quantités d'objets, produits industriels extrêmement sophistiqués qui ne servent rapidement plus à rien. Dans un autre domaine, celui de l'organisation, il est pour le moins curieux d'observer comment évoluent les actes collectifs importants qui soutendent notre société. D'abord le phénomène surgit pour résoudre un problème grave, il se constitue et se conforte dans un milieu bien souvent hostile, puis à partir du moment où s'établit un fonctionnement de croisière, se met à dériver peu à peu vers une sordide caricature de sa propre réalité. C'est à peu près le cas de toutes les institutions, qui avaient au début un rôle à jouer, puis qui sont devenues des sortes de forteresses tenues par des castes. Et le jeu n'est plus d'attendre que de telles organisations offrent un quelconque service, et encore moins le genre de service qui était à l'origine de leur création, mais bien plutôt d'apprendre à jouer la comédie rituelle et servile qui a des chances de faire qu'on devienne un jour en son sein, ou qu'on espère y devenir un personnage important, dont le rôle sera essentiellement à son tour d'entraver la montée des nouveaux venus. Je ne citerai pas le syndicalisme, le dernier venu, qui, en période de crise aiguë, n'a pas hésité à venir au secours de son ennemi naturel, le pouvoir, sous prétexte qu'il n'était pas l'instigateur de l'agitation. Plus aucun syndicalisme ne pourra faire autre chose, désormais, que de la paperasserie. Et inutile de parler non plus de nos vieux, de nos très vieux corps constitués.

Toute la société moderne, au moins dans les pays riches, se structure de la sorte au moyen des pires critères d'inutilité. Plus une chose est formelle, vaine, détachée de la réalité la plus simple, plus elle a de chances de devenir un bon critère de sélection. Le

système ainsi établi semble créé sur une base très stable aussi surprenant que cela puisse paraître si l'on ne s'en tient qu'aux déclarations officielles. Mais en fait, le mécanisme est très ingénieux, et semble à toute épreuve, bien qu'il ne soit pas facile à dévoiler. Par instants, cependant, la machine se découvre. Le tout est de regarder au bon moment. Prenons une organisation quelconque par exemple, Les premiers qui sont dans la place, sont la plupart du temps des sortes de pionniers doublés de militants, des gens passionnés par leur entreprise. Puis quand la machine fonctionne, on remplace de plus en plus les organisateurs dynamiques par des gestionnaires mâtinés de comptables. Qui, à leur tour sont remplacés par des aventuriers du genre parasite. Parvenu à ce point l'organisme tourne sur lui-même sans plus rien produire, sinon une organisation inutile pour la société dans laquelle elle baigne, et nuisible dans la mesure où elle brise définitivement toutes les initiatives. Le système est générateur d'incapacité. Et l'incapacité pousse à la méfiance, qui pousse à la mesquinerie. Mais pour peu qu'elle ait atteint un poids relativement important, l'incapacité se met à sécréter sectarisme et intolérance pour son contraire. La perte des qualités qui sont le fleuron de l'ancienne éducation entraîne à la paranoïa, et la paranoïa à tous les sectarismes. Notre société dégénère car elle noie l'initiative, la personnalité, l'originalité dans une mer visqueuse de médiocrité et d'écoeurement.

Les filtres administratifs sont de plus en plus sélectifs et efficaces. Là où, il y a seulement peu d'années, une entrevue suffisait pour choisir un candidat, il faut maintenant franchir le barrage de plusieurs commissions dont la constitution elle-même est d'une horrible complexité. Les candidats doivent faire la preuve de leurs capacités en inondant les examinateurs de masses de dossiers. Que plus personne n'a le temps de lire vraiment. Les gens sont donc, en fait, sélectionnés sur une phrase entrevue en feuilletant rapidement une liasse, sur un commentaire vaguement entendu, mais surtout sur l'opinion des petits copains. Aussi conseil aux jeunes: n'essayez surtout pas de faire montre de la moindre originalité, car d'abord, vous n'avez que peu de chance d'être lu si vous n'êtes pas 'aidé', et si vous l'étiez vous pourriez effrayer votre rapporteur, qui sait bien lui-même comment il a été sélectionné, et qui craint avant tout de se mouiller.

Voilà d'ailleurs encore un effet pervers de ce qu'on appelle l'exercice de la démocratie. En effet c'est une majorité qui va décider du choix, dans une commission. Le plus malin c'est donc celui qui passe le plus de temps à y placer ses hommes de main . Celui qui préfère s'isoler dans son travail, celui-là n'a aucune chance d'obtenir un jour une quelconque décision. Le tout est de savoir ce que l'on veut.

Nous voulons la médiocrité majoritaire, nous l'avons. La médiocrité a d'autres conséquences dramatiques. C'est le 'toujours davantage' qui caractérise le moteur de notre société. On pourrait imaginer qu'à partir du moment où le nécessaire et un certain superflu sont atteints, un autre but devrait guider l'ambition du monde. Par exemple profiter de l'acquis pour essayer de vivre. A partir d'un certain point qui se révèle de non retour, la possession nous rend insatiables. Avec toutes les catastrophes écologiques et sociales qui en découlent. Sans oublier que, dans ce domaine précis, les pays pauvres tentent d'imiter les pays riches.

Les famines, les épidémies, les catastrophes naturelles ou pas, les dictatures, les despotismes, les tyrannies en tous genres et autres saccages de la planète ont encore de beaux jours devant eux, sous l'œil bienveillant des Mages, des Sorciers, des devins, des Enchanteurs, des Astrologues, des Envoûteurs, des Pères Spirituels en tous genres, et autres marchands de dieux, de diables et de diableries.

Encore une question, combien de milliers de siècles faudra-t-il encore pour que cette pauvre cervelle humaine prenne conscience de la réalité ? Question subsidiaire, qui ira le plus vite, la compréhension des problèmes du moment et de ceux du futur, ou bien la complexification de ces mêmes problèmes au long de l'essor démographique ?

## **LE PROCESSEUR DE LA PROCEDURE FORMELLE**

**Subject classification informatics: D33, D34.**

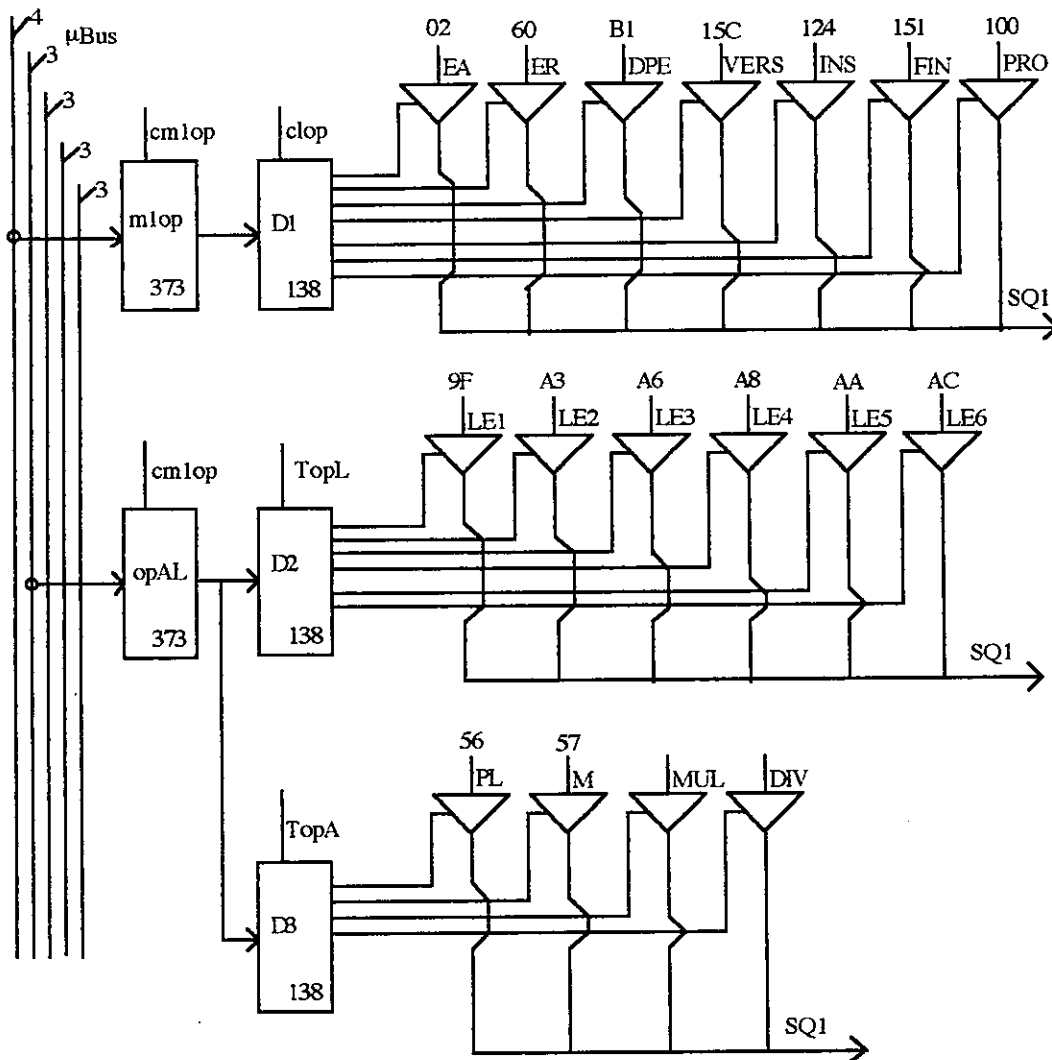
### Résumé.

La première partie essayait de donner une idée de l'organisation de la circuiterie. Sans entrer dans trop de détails car le dossier complet est très épais. La présente suite décrit au moins l'essentiel de ce qui fait le moteur de l'ensemble. La partie "calcul" du processeur est mue par un jeu d'instructions simples qui représentent les opérations élémentaires. La partie "pilote" est en fait un programme rédigé avec de telles instructions, et c'est ce programme qui décrit la fonction "séquenceur". Dans cette partie de l'article, en sont présentées les parties essentielles.

## LE PROCESSEUR DE LA PROCEDURE FORMELLE.

### TRAITEMENT DU CODE OPERATEUR.

Deux cellules: m1op et opAL, commandées également par cm1op, comme le montre le schéma qui suit, contiennent le code de l'opérateur récupéré en mémoire par la première instruction du programme pilote:  $cm1 := cM$ , qui charge également les cellules n1, n2, et n3.



Le premier aiguillage Aig  $cm1op$  qui émet la commande  $C1op$ , en fonction du code de l'opérateur rencontré, renvoie dans le programme pilote à l'une de ces adresses:

EA, ER, DPE, VERS, INS, FIN, et PRO,

Les buffers génèrent directement les valeurs de ces adresses en binaire (ou hexadécimal) qui attaquent le circuit du séquenceur en adresse suivante par SQ1. Ce sont les nombres hexadécimaux portés dans le schéma en entrée des buffers.

**Correction.**

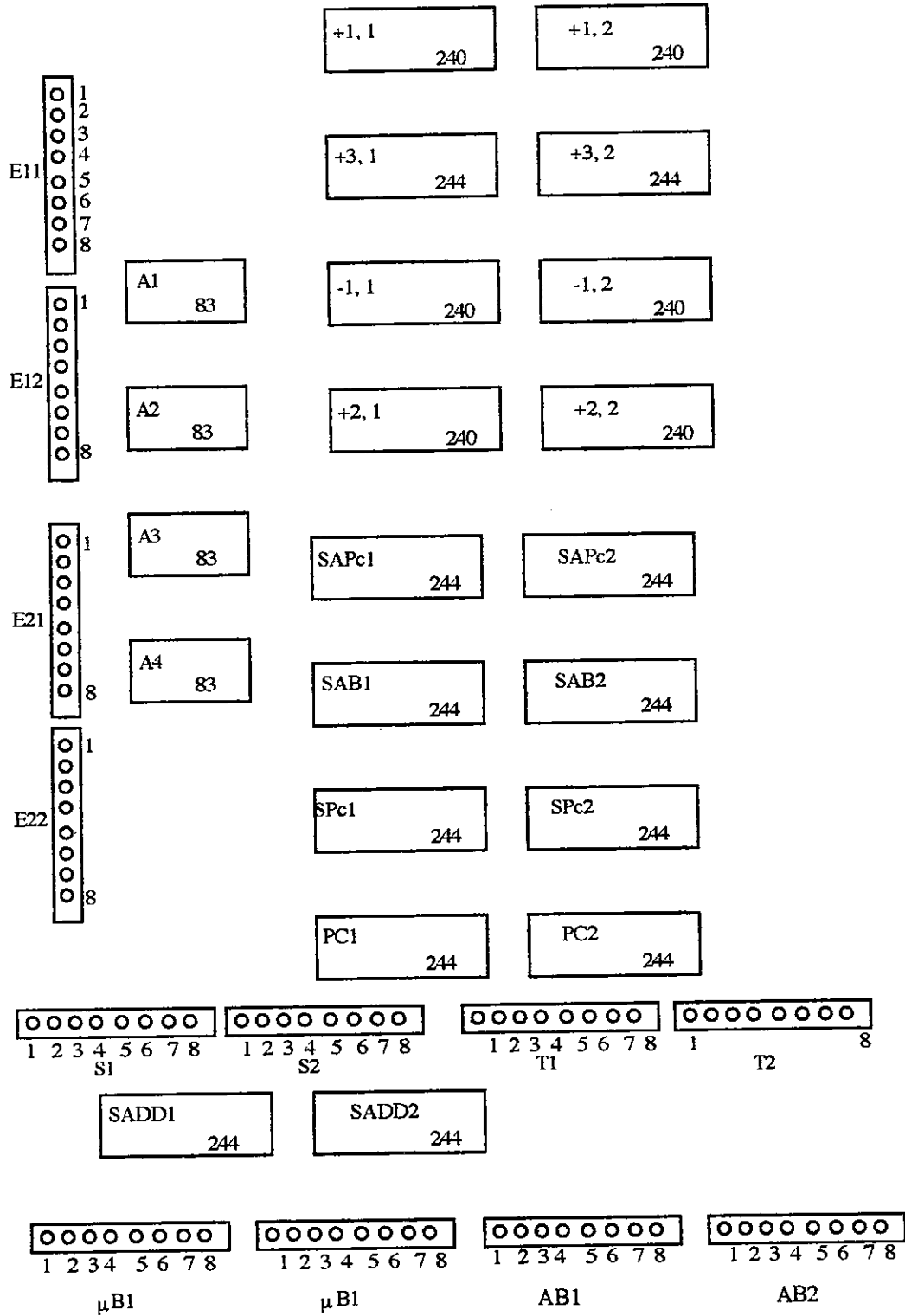
Dans le schéma de la page 19 du précédent numéro, entre le point F2 et la deuxième entrée du comparateur, il faut rajouter un buffer dont la commande est 'comV2', .

**LES LANGAGES DE DESCRIPTION DE CIRCUITS.**

Dans la première partie, je montre le principe du processeur et de son environnement avec des schémas simplifiés dans la forme, qui permettent d'un coup d'œil d'embrasser l'idée de principe de l'organisation. Tout bus y est représenté par un trait unique quel que soit le nombre de moments qui le composent. Ce nombre est d'ailleurs précisé par une oblique et un coefficient. Un latch représenté par un rectangle sera peut-être traduit par deux circuits sur la carte, si ce latch est à seize bits. Il devient donc nécessaire de donner une représentation de la carte en termes de répartition de circuits intégrés sur une surface, représentation à laquelle il faut ajouter les tableaux de câblage. Ce sont ces tableaux qui décrivent l'essentiel du travail de "wrapping". Pour chaque circuit, on y retrouve la liste des broches et, en face, le nom de la broche, ou des broches auxquelles chacune d'entre elles est reliée.

Les schémas des pages 17 et 19 du N° 31 sont traduits sur quatre cartes séparées, qui sont reliées au reste du processeur par trois bus: commandes, adresses et données. Je donne le schéma de répartition des circuits intégrés sur la carte qui matérialise le calcul d'adresse.

## CARTE CALCUL D'ADRESSE.



Les lignes de broches: E11, E12, E21 et E22 sont des bus de communication d'une carte sur l'autre. En effet les deux accès de l'additionneur d'adresses, les points E1 et E2 sont également



attaqués par les sorties des latch  $d$  et  $pc''$  pour E1, et des latch  $a$  et  $pc'$  pour E2. Sur la carte elle-même, c'est SAPC qui attaque directement E2, et les émetteurs des constantes +1, -1, 3 et 2 qui attaquent E1.

Les lignes de broches  $\mu B1$ ,  $\mu B2$ , AB1 et AB2 sortent sur le bus qui passe par le fond de panier. Sur cette carte, les circuits intégrés sont maintenus par les commandes: cPC, sPC, sAB, sAPC, sADD, un, mun, trois et deux. Ce bus de commande est connecté à la carte du séquenceur par un câble plat.

#### TABLEAUX DE CONNECTIONS.

A4	A3	A2	A1
74HC83	83	83	83
1 E12(8)	1 E12(4)	1 E11(8)	1 E11(4)
2 S2(7)	2 S2(3)	2 S1(7)	2 S1(3)
3 E12(7)	3 E12(3)	3 E11(7)	3 E11(3)
4 E22(7)	4 E22(3)	4 E21(7)	4 E21(3)
5 VCC	5 VCC	5 VCC	5 VCC
6 S2(6)	6 S2(2)	6 S1(6)	6 S1(2)
7 E22(6)	7 E22(2)	7 E21(6)	7 E21(2)
8 E12(6)	8 E12(2)	8 E11(6)	8 E11(2)
9 S2(5)	9 S2(1)	9 S1(5)	9 S1(1)
10 E12(5)	10 E12(1)	10 E11(5)	10 E11(1)
11 E22(5)	11 E22(1)	11 E21(5)	11 E21(1)
12 masse	12 masse	12 masse	12 masse
13 A3(14)	13 A2(14)	13 A1(14)	13 masse
14 $\omega$	14 A4(13)	14 A3(13)	14 A2(13)
15 S2(8)	15 S2(4)	15 S1(8)	15 S1(4)
16 E22(8)	16 E22(4)	16 E21(8)	16 E21(4)

Cet additionneur de seize bits par seize bits sur seize bits et un report est connecté en entrée sur E11, E12 et E21, E22, et en sortie sur S1, S2.

Le latch PC, lui, est relié en entrée sur le micro-bus et en sortie sur T1, T2. Ce sont les buffers SAB, SPC, et SAPC qui sont connectés en entrée sur T1 et T2.

PC1

74HC373	
1	masse
2	T1(1)
3	$\mu$ B1(1)
4	$\mu$ B1(2)
5	T1(2)
6	T1(3)
7	$\mu$ B1(3)
8	$\mu$ B1(4)
9	T1(4)
10	masse
11	cPC,PC2(11)
12	T1(5)
13	$\mu$ B1(5)
14	$\mu$ B1(6)
15	T1(6)
16	T1(7)
17	$\mu$ B1(7)
18	$\mu$ B1(8)
19	T1(8)
20	VCC

PC2

373	
1	masse
2	T2(1)
3	$\mu$ B2(1)
4	$\mu$ B2(2)
5	T2(2)
6	T2(3)
7	$\mu$ B2(3)
8	$\mu$ B2(4)
9	T2(4)
10	masse
11	cPC,PC1(11)
12	T2(5)
13	$\mu$ B2(5)
14	$\mu$ B2(6)
15	T2(6)
16	T2(7)
17	$\mu$ B2(7)
18	$\mu$ B2(8)
19	T2(8)
20	VCC

sPC1

74HC244	
1	sPC,sPC2(1)
2	T1(1)
3	$\mu$ B1(8),PC1(18)
4	T1(2)
5	$\mu$ B1(7),PC1(17)
6	T1(3)
7	$\mu$ B1(6),PC1(14)
8	T1(4)
9	$\mu$ B1(5),PC1(13)
10	masse
11	T1(5)
12	$\mu$ B1(4),PC1(8)
13	T1(6)
14	$\mu$ B1(3),PC1(7)
15	T1(7)
16	$\mu$ B1(2),PC1(4)
17	T1(8)
18	$\mu$ B1(1),PC1(3)
19	sPC,sPC2(19)
20	VCC

sPC2

244	
1	sPC,sPC1(1)
2	T2(1)
3	$\mu$ B2(8),PC2(18)
4	T2(2)
5	$\mu$ B2(7),PC2(17)
6	T2(3)
7	$\mu$ B2(6),PC2(14)
8	T2(4)
9	$\mu$ B2(5),PC2(13)
10	masse
11	T2(5)
12	$\mu$ B2(4),PC2(8)
13	T2(6)
14	$\mu$ B2(3),PC2(7)
15	T2(7)
16	$\mu$ B2(2),PC2(4)
17	T2(8)
18	$\mu$ B2(1),PC2(3)
19	sPC,sPC1(19)
20	VCC

Le buffer sPC est connecté en entrée sur la sortie de PC par l'intermédiaire des lignes de broches T1 et T2, et il est connecté en sortie sur le micro-bus. Le buffer sAPC sert de porte entre la sortie de PC et l'entrée E2 de l'additionneur. Le buffer sAB conduit la sortie de PC sur le bus adresse qui, via le fond de panier, aborde la mémoire.

sAPC1

74HC244	
1	sAPC
2	T1(1)
3	E21(8),A2(16)
4	T1(2)
5	E21(7),A2(4)
6	T1(3)
7	E21(6),A2(4)
8	T1(4)
9	E21(5),A2(11)
10	masse
11	T1(5)
12	E21(4),A1(16)
13	T1(6)
14	E21(3),A1(4)
15	T1(7)
16	E21(2),A1(7)
17	T1(8)
18	E21(1),A1(11)
19	sAPC
20	VCC

sAPC2

244	
1	sAPC
2	T2(1)
3	E22(8),A4(16)
4	T2(2)
5	E22(7),A4(4)
6	T2(3)
7	E22(6),A4(7)
8	T2(4)
9	E22(5),A4(11)
10	masse
11	T2(5)
12	E22(4),A3(16)
13	T2(6)
14	E22(3),A3(4)
15	T2(7)
16	E22(2),A3(7)
17	T2(8)
18	E22(1),A3(11)
19	sAPC
20	VCC

sAB1

244	
1	sAB
2	T1(1)
3	AB1(8)
4	T1(2)
5	AB1(7)
6	T1(3)
7	AB1(6)
8	T1(4)
9	AB1(5)
10	masse
11	T1(5)
12	AB1(4)
13	T1(6)
14	AB1(3)
15	T1(7)
16	AB1(2)
17	T1(8)
18	AB1(1)
19	sAB
20	VCC

sAB2

244	
1	sAB
2	T2(1)
3	AB2(8)
4	T2(2)
5	AB2(7)
6	T2(3)
7	AB2(6)
8	T2(4)
9	AB2(5)
10	masse
11	T2(5)
12	AB2(4)
13	T2(6)
14	AB2(3)
15	T2(7)
16	AB2(2)
17	T2(8)
18	AB2(1)
19	sAB
20	VCC

Pour compléter l'exemple, je donne les tableaux de câblage du buffer sADD et de la constante -1. Les tableaux des constantes +1,

+2 et +3 s'en déduisent immédiatement, je ne les reproduis donc pas.

sADD1		sADD2		2 <sub>1</sub>		2 <sub>2</sub>	
74HC244		244		244		244	
1	sADD	1	sADD	1	deux	1	deux
2	S1(1)	2	S2(1)	2	masse	2	masse
3	μB1(8)	3	μB2(8)	3	E11(8)	3	E11(8)
4	S1(2)	4	S2(2)	4	VCC	4	VCC
5	μB1(7)	5	μB2(7)	5	E11(7)	5	E11(7)
6	S1(3)	6	S2(3)	6	masse	6	masse
7	μB1(6)	7	μB2(6)	7	E11(6)	7	E11(6)
8	S1(4)	8	S2(4)	8	masse	8	masse
9	μB1(5)	9	μB2(5)	9	E11(5)	9	E11(5)
10	masse	10	masse	10	masse	10	masse
11	S1(5)	11	S2(5)	11	masse	11	masse
12	μB1(4)	12	μB2(4)	12	E11(4)	12	E11(4)
13	S1(6)	13	S2(6)	13	masse	13	masse
14	μB1(3)	14	μB2(3)	14	E11(3)	14	E11(3)
15	S1(7)	15	S2(7)	15	masse	15	masse
16	μB1(2)	16	μB2(2)	16	E11(2)	16	E11(2)
17	S1(8)	17	S2(8)	17	masse	17	masse
18	μB1(1)	18	μB2(1)	18	E11(1)	18	E11(1)
19	sADD	19	sADD	19	deux	19	deux
20	VCC	20	VCC	20	VCC	20	VCC

Ces tableaux sont importants pour le contrôle du câblage. Chaque fois qu'une broche est wrappée, on coche la ligne qui désigne la broche. Par exemple, la broche 16 du sADD1 sera cochée, ainsi que la broche 2 du μB1, dès que le fil qui les relie sera mis en place. Les broches 1 et 19 des quatre circuits ci-dessus sont, elles, reliées aux commandes 'sADD' et 'deux', qui sont émises par les PROM P1 à P10 du séquenceur. Ce sont ces commandes qui mettent chaque circuit en activité, lorsqu'elles sont produites par le programme pilote du séquenceur.

## LE PROGRAMME PILOTE.

### Les instructions.

La batterie de PROM: P1, P2, P3, ... , P10 se comporte comme un tableau de valeurs binaires de 32K lignes dont chaque ligne comporte 10 X 8 bits. J'ai utilisé des R PROM 2732A dont la technologie est hélas un peu dépassée. On ne trouve déjà plus sur le marché que des 2764A, et encore jusqu'à quand ? Ces derniers circuits sont des 28 broches, contre seulement 24 pour les 2732. Cela représente beaucoup de câblage inutile alors que je n'utilise qu'environ 500 lignes.

Je vais maintenant définir un langage de plus, celui qui sert pour le programme pilote. Chaque ligne de chaque PROM est un octet, dont chaque bit est une commande qui permet d'établir des communications dans l'ensemble de la circuiterie du processeur.

J'en donne la liste pour chaque PROM, les noms qui leur sont attribués sont ceux qu'on retrouve dans les différents schémas, en commande des circuits.

P10	P9	P8	P7	P6	P5	P4	P3	P2	P1
un	sAB	spc'	rep	cv1	ca3	com1	clop		ct1dpe
mun	sADD	ca	Erep	sv1	sa3	com2		cn2A	ct1A
deux	se2p'c	sa	Srep	cv2	cpc	com3	TopL	cn2L	M
trois	sV1a	cd	sADV	svn1A	spc	com4	TopA	cn2dpe	dM
zéro	saa	sd	EV1	ca1	cAD	com5			R/W
cPC	seid	cpc''	EV2	sa1	sAD	com6	cn1A	cn3A	cVS
sPC	selpc''	spc''	Mev2	ca2	comV1	com7	cn1L	ct1	VS1
sARC	cpc'	co	dun	sa2	comV2	cm1op	cn1dpe		VS2

Ces commandes sont normalement à la valeur 1, qui est la valeur généralement admise (nous en verrons des exceptions) pour bloquer tout circuit. Donc, pour une instruction déterminée parmi ces 80 commandes seules celles qui sont destinées à débloquer le ou les circuits correspondants, sont à la valeur 0.

Par exemple: l'instruction

**Aig cm1op**

qui dérive vers une adresse particulière pour chaque valeur du code de l'opérateur, comporte la commande clop à zéro dans P3, ce qui donne pour la ligne de P3:

0111 1111

soit en hexadécimal:

7 F

sur la même ligne mais pour les autres PROM on n'aura que des: FF.

	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1
Aig cm1op	FF	FF	FF	FF	FF	FF	FF	7F		
Aig TopA								EF		
Aig cn1dpe								FE		
Aig cn1A								FB		
Aig cn1L								FD		
Aig cn2dpe									EF	
Aig cn2A									BF	
Aig cn2L									DF	
Aig cn3A									FB	
Aig ct1A										BF
Aig TopL								DF		
Aig ct1dpe										7F
cm1(op,n1,n2,n3):=cM							FE			CF
si cV1=cV2 vers VRAI						FC	7F			
si cV1<cV2 vers VRAI						FC	BF			
si cV1>cV2 vers VRAI						FC	DF			
si cV1>cV2 vers FAUX						FC	EF			
si cV1<cV2 vers FAUX						FC	F7			
si cV1=cV2 vers INSF	FF					FC	FB			
si cV1=cV2 vers PROF	FF	FF	FF	FF	FF	FC	FD	FF	FF	FF

Les casés en blanc contiennent toutes la valeur: FF. Donc, seules apparaissent les cases qui contiennent des zéros, c'est-à-dire des commandes actives.

Les douze premières instructions correspondent au fonctionnement des circuits d'analyse des statuts: n1, n2 et n3, dont les schémas sont en page 24, 25 et 26 de ce même précédent numéro. Les sept dernières sont les images de l'exploitation des valeurs des trois bits que fournit le comparateur lorsqu'on lui injecte les contenus de V1 et V2.

	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1
cPC := cpc'	FB		7F				FF	FF	FF	FF
cPC := cpc' + 1	7B	9F								
cPC := ca1	FB				FB					
cPC := ca2	FB				FE					
cPC := ca3	FB					BF				
cPC := ca + cpc''	FB	B5								
cPC := ca + 2	DB	B7								
cPC := cd	FB		F7							
cPC := ca + cd	FB	B3								
cPC := ca + 1	7B	B7								
cPC := cpc	FB					EF				
cpc' := cPC + 1	7E	BE								
cpc' := cPC	FD	FE								
cpc' := cd	FF	FE	F7							
cpc' := ca - 1	BF	B6								
ca := cM	FF		BF							CF
ca := ca1			BF		FB					
ca := ca2			BF		FE					
ca := cV1 + cV2			BE	63						
ca := cV1 + (cV2)			BE	65						
ca := cV1 + 1			BE	66						
ca := cd			B7							
ca := cAD			BF			FB				
ca := ca3			BF			BF				
cd := cV1			EF		BF					
cd := ca			CF							
cd := ca + 1	7F	B7	EF							
cd := cpc'	FF	FF	6F	FF	FF	FF	FF	FF	FF	FF

Les instructions de communication de registre à registre, sont classées par registre affecté. Les instructions les plus complexes de la procédure formelle, comme l'insertion, par exemple exigent le plus de manipulations à base de ces opérations élémentaires. C'est le cas, en particulier, pour les boucles de traitement des listes de paramètres. C'est ce fait qui impose l'assez grand nombre de registres: soit 11 au total, sans compter ni PC ni  $\omega$ . L'abondance des émetteurs de constantes ( -1, +1, +2, +3 ), procède de la même raison, cela permet un calcul d'adresses plus facile.

	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1
cpc <sup>n</sup> := cM	FF	FF	FB	FF	FF	FF	FF	FF	FF	CF
cpc <sup>n</sup> := cpc			FB			EF				
cpc <sup>n</sup> := cPC	FD		FB							
cpc <sup>n</sup> := ca + 1	7F	B7	FB							
cpc <sup>n</sup> := ca2			FB		FE					
cpc <sup>n</sup> := ca3			FB			BF				
cAD := cM						F7				CF
cAD := ca - 1	BF	B7				F7				
ca1 := ca + cd		B3			F7					
ca1 := ca + cV1		A7			F7					
ca1 := cM					F7					CF
ca1 := ca3					F7	BF				
ca1 := ca + cpc <sup>n</sup>		B5			F7					
ca1 := ca + 3	EF	B7			F7					
ca1 := ca + 1	7F	B7			F7					
ca2 := ca + cd		B3			FD					
ca2 := ca + cV1		A7			FD					
ca2 := ca + (cV1)		B7			ED					
ca2 := ca + 1	7F	B7			FD					
ca2 := cM					FD					CF
ca2 := ca - 1	BF	B7			FD					
ca3 := ca + cd		B3				7F				
ca3 := ca + cpc <sup>n</sup>		B5				7F				
ca3 := ca			DF			7F				
ca3 := ca + 2	DF	B7				7F				
ca3 := ca + 1	7F	B7	FF	FF	FF	7F	FF	FF	FF	FF

Dans le tableau suivant, les instructions:

vers VERSN  
vers INSN  
vers FINN  
vers PRON

permettent le changement de banc de mémoire. Elles font passer du banc 0 au banc 1, par changement de valeur du bit N°9. En effet, le bus du séquenceur est à neuf moments, ce sont les latch L1 et L2 qui gèrent les huit poids faibles et les latch SL1 et SL2 commandés par cVS et  $\phi_2$ , et alimentés par VS1, qui gèrent le poids fort complémentaire. Le programme pilote tient actuellement sur 500 lignes, mais grâce à SL1 et SL2, il pourrait occuper davantage de place. L'instruction

vers Entrée

ramène le bit N°9 à la valeur zéro, ce qui permet de revenir dans le banc 0.

	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1
cV2 := cM					DF		FF	FF	FF	CF
cV2 := cω				DF	DF					
cV2 := cd			F7		DF					
cV2 := 0	F7				DF					
cV2 := ca1					DB					
cM := cV1					BF					D7
cM := ca			DF							D7
cM := ca3						BF				D7
cM := cpc"			FD							D7
cM := cpc'			7F							D7
cω := cV1			FE	BF	BF					
cω := ca			DE	BF						
cpc := cPC	FD					DF				
cpc := cpc" + ca		B5				DF				
vers entrée										F9
vers VERSN										FB
vers INSN										FB
vers FINN										FB
vers PRON										FB
ct1 := 0	F7								FD	
ct1 := 1	7F								FD	
ct1 := 2	DF								FD	
cV1 := cM					7F					CF
cV1 := cω				DF	7F					
cV1 := cd			F7		7F					
cV1 := ca			DF		7F					
cV1 := cAD					7F	FB				
cV1 := ca2	FF	FF	FF	FF	7E	FF	FF	FF	FF	FF

Le taux d'utilisation de chacune de ces instructions est très inégal, certaines sont courantes, apparaissent fréquemment dans le programme, d'autres telles par exemples les changements de bancs cités plus haut n'apparaissent qu'une fois. Je donnerai en exemple seulement quelques fractions du programme pilote qui est assez long, pour en montrer simplement le principe.

### Le Programme Pilote.

Le premier aiguillage, qui s'applique sur le contenu du registre m1, dérive vers les traitements des diverses classes d'opérateurs. (cf. P.9 du B31). La classe des opérateurs arithmétiques qui comporte quatre membres (en réalité deux seulement, pour l'instant), est traitée en adresse EA (02H). Ensuite se fait pour chacun des opérandes l'analyse des statuts n1, n2 et n3. Selon, par exemple que n1 = 0, 1, 2 ou 3, on calcule l'adresse qui correspond à la forme d'opérande, soit [a, soit [a,b, soit [ω, soit [I. En fait, les deux derniers cas ne correspondent pas à un vrai calcul d'adresse car il n'est fait appel à aucune valeur fournie par le code, on n'a affaire qu'à des registres: ω et I, c'est-à-dire aux cellules ω et d. Il suffit d'imposer une valeur caractéristique à la cellule t1, soit 1, soit 2, de sorte que, les trois opérandes ayant été analysés, l'opérateur soit appliqué soit sur une adresse, soit sur un registre.

Pour les deux opérandes émetteurs, définis par n2 et n3, dès que les adresses sont évaluées, les valeurs sont placées en V1 et V2 (y compris si les valeurs sont à prendre en  $\omega$  ou en d).

Quand l'opérateur est de type logique, la classe comporte six membres, on traite les opérandes de la même manière que les deux opérandes émetteurs du cas précédent. Le calcul se déroule à partir de l'adresse ER (60H), et s'achève en adresse LE (8EH) où l'aiguillage TopL dirige vers l'un des six cas possibles: LE1, LE2, LE3, LE4, LE5 ou LE6. C'est en ces six points qu'on exploite les trois valeurs binaires émises par le comparateur de la page 19 du B31.

Les opérateurs de procédure et d'insertion, qui renvoient en PRO (100H) et en INS (124H), exigent le déroulement de boucles pour:

- 1) Constituer la partie locale de la configuration, mettre en place son volume, et calculer les adresses des variables locales,
- 2) Faire apparaître la configuration de la procédure à insérer, en portant dedans, le volume de la précédente, l'accès au retour dans le code de l'algorithme, et la liste des paramètres, puis faire évoluer [I].

Dans la présentation du programme, les deux premières colonnes contiennent le numéro de ligne de l'instruction, et le numéro de ligne suivante.

#### PROGRAMME.

```

00 01  Entrée:      cm1(op,n1,n2,n3) := cM
01 --  Aig cm1op(EA,ER,DPE,VERS,INS,FIN,PRO)
02 03  EA:        cpc' := cPC + 1
03 04          cPC := cpc'
04 --  Aig cn1A (0:AU0,1:AU1,2:AU2,3:AU3)
05 06  AU0:      ca := cM
06 07          cal := ca + cd
07 08          cPC := cpc' + 1
08 1A          ct1 := 0                                vers AD
0A 0B  AU1:      ca := cM
0B 0C          cal := ca + cd
0C 0D          cPC := cpc' + 1
0D 0E          cV1 := cM
0E 0F          cpc' := cPC
0E 10          cPC := cal
10 11          ca := cM
11 12          cal := ca + cV1
12 13          ca := cal
13 14          cal := ca + cd
14 15          cPC := cpc' + 1
15 1A          ct1 := 0                                vers AD
16
17 1A  AU2:      ct1 := 1
18 1A  AU3:      ct1 := 2
19
1A    AD:      Aig cn2A (AD0,AD1,AD2,AD3,1D4)

```



1B	1C	AD0:	ca := cM	
1C	1D		ca2 := ca + cd	
1D	1E		cpc' := cPC + 1	
1E	1F		cPC := ca2	
1F	20		cV1 := cM	
<hr/>				
20	39		cPC := cpc'	vers AT
21				
22	23	AD1:	ca := cM	
23	24		ca2 := ca + cd	
24	25		cpc' := cPC	
25	26		cPC := cpc' + 1	
26	27		cV1 := cM	
27	28		cpc' := cPC	
28	29		cPC := ca2	
29	2A		ca := cM	
2A	2B		ca2 := ca + cV1	
2B	2C		ca := ca2	
2C	2D		ca2 := ca + cd	
2D	2E		cPC := cpc' + 1	
2E	2F		cpc' := cPC	
2F	30		cPC := ca2	
30	31		cV1 := cM	
31	39		cPC := cpc'	vers AT
32				
33	39	AD2:	cV1 := c $\omega$	vers AT
34	39	AD3:	cV1 := cd	vers AT
35	36	AD4:	cV1 := cM	
36	37		cpc' := cPC + 1	
37	39		cPC := cpc'	vers AT
38				
39	--	AT:	Aig cn3A(AT0,AT1,AT2,AT3,AT4)	
3A	3B	AT0:	ca := cM	
3B	3C		ca3 := ca + cd	
3C	3D		cpc' := cPC + 1	
3D	3E		cPC := ca3	
3E	3F		cV2 := cM	
3F	55		cPC := cpc'	vers AF
<hr/>				
40	41	AT1:	ca := cM	
41	42		ca3 := ca + cd	
42	43		cPC := cpc' + 1	
43	44		cpc'' := cM	
44	45		cpc' := cPC	
45	46		cPC := ca3	
46	47		ca := cM	
47	48		ca3 := ca + cpc''	
48	49		ca := ca3	
49	4A		ca3 := ca + cd	
4A	4B		cPC := ca3	
4B	4C		cV2 := cM	
4C	4D		cPC := cpc' + 1	
4D	55		cpc' := cPC	vers AF
4E				
4F	55	AT2:	cV2 := c $\omega$	vers AF
50	55	AT3:	cV2 := cd	vers AF
51	52	AT4:	cV2 := cM	
52	53		cpc' := cPC + 1	

53	55		cPC := cpc'	vers AF
54				
55	--	AF:	Aig cm1op(PL,M,MUL,DIV)	
56	5A	PL:	ca := cV1 + <sub>K</sub> cV2	vers RES
57	58	M:	ca t= cV1 + <sub>K</sub> ( $\neg$ cV2)	
58	59		cV1 := ca	
59	5A		ca := cV1 + 1	vers RES
5A	--	RES:	Aig ct1A(Sort, Omeg, Hi)	
5B	5C	Sort:	cPC := ca1	
5C	5D		cM := ca	
5D	00		cPC := cpc'	vers Entrée
5E	00	Omeg:	c $\omega$ := ca	vers Entrée
5F	00		cd := ca	vers Entrée

---

Les lignes suivantes du programme commencent à l'étiquette ER et traitent des opérateurs de relation. Les calculs qui concernent les opérands sont identiques à ceux que nous venons de voir, tout au moins pour le cas des deux derniers. Les valeurs des opérands étant alors contenues dans les cellules V1 et V2, l'exploitation de leurs valeurs se fait par les lignes qui suivent:

9E	--	LE:	Aig TopL(LE1, LE2, LE3, LE4, LE5, LE6)	
9F	A0	LE1:	si cV1 = cV2 vers VRAI	
A0	A1	FAUX:	cpc' := cPC + 1	
A1	00		cPC := cpc'	vers Entrée
A2				
A3	A4	LE2:	si cV1 < cV2 vers VRAI	
A4	A0		si cV1 > cV2 vers VRAI	sinon vers FAUX
A5				
A6	A0	LE3:	si cV1 < cV2 vers VRAI	sinon vers FAUX
A7				
A8	AD	LE4:	si cV1 > cV2 vers VRAI	sinon vers FAUX
A9				
AA	A0	LE5:	si cV1 > cV2 vers VRAI	sinon vers FAUX
AB				
AC	AD	LE6:	si cV1 < cV2 vers FAUX	sinon vers VRAI
AD	AE	VRAI:	ca := cM	
AE	AF		cpc'' := cPC	
AF	00		cPC := ca + cpc''	vers Entrée
B0				
B1	B2	DPE:	cpc' := cPC + 1	

---

Je vais maintenant détailler les traitements des opérateurs de procédure, d'insertion et de fin. D'abord les quatre instructions de changement de banc de mémoire pour changer la valeur du contenu des cellules SL1 et SL2 du séquenceur.

FC	5D	VERS:	vers VERSN
FD	24	INS:	vers INSN
FE	51	FIN:	vers FINN
FF	00	PRO:	vers PRON

---

100	101	PRON:	cpc' := cPC + 1	
101	102		cPC := cpc'	
102	103		ca := cM	
103	104		cpc' := cPC + 1	
104	105		cPC := cpc'	
105	106		cpc'' := cM	
106	107		ca1 := ca + cpc''	
107	108		cpc' := cPC + 1	
108	109		cPC := cpc'	
109	10A		ca2 := cM	
10A	10B		ca := ca1	
10B	10C		ca1 := ca + 3	
10C	10D		ca := ca1	
10D	10E		cpc'' := ca2	
10E	10F		ca3 := ca + cpc''	
10F	111		cpc' := cPC + 1	
110				
111	112	PRO1:	cV1 := ca2	
112	113		cV2 := 0	
113	114		si cV1 = cV2 vers PROF	
114	115		ca := ca1	
115	116		cPC := ca + cd	
116	117		cM := ca3	
117	118		ca1 := ca + 1	
118	119		cPC := cpc'	
119	11A		ca := cM	
11A	11B		cpc'' := ca3	
11B	11C		ca3 := ca + cpc''	
11C	11D		ca := ca2	
11D	11E		ca2 := ca - 1	
11E	112		cpc' := cPC + 1	vers PRO1
11F				
-----				
120	121	PROF:	cPC := cd	
121	122		cM := ca3	
122	123		cPC := cpc'	
123	00		vers Entrée	
124	125	INSN:	cpc'' := cPC	
125	126		cpc' := cPC + 1	
126	127		cPC := cpc'	
127	128		ca := cM	
128	129		cpc := cpc'' + ca	
129	12A		cpc' := cPC + 1	
12A	12B		cPC := cpc'	
12B	12C		cAD := cM	
12C	12D		cpc' := cPC + 1	
12D	12E		cPC := cd	
12E	12F		ca := cM	
12F	130		ca1 := ca + cd	
130	131		ca3 := ca	
131	132		ca := ca1	
132	133		ca2 := ca + 1	
133	134		cPC := ca2	
134	135		cM := ca3	
135	136		ca := ca2	
136	137		ca1 := ca3	
137	138		ca3 := ca + 2	
138	139	INS1:	cV1 := cAD	

139	13A		cV2 := 0	
13A	13B		si cV1 = cV2 vers INSF	
13B	13C		cPC := cpc'	
13C	13D		ca := cM	
13D	13E		cPC := ca + cd	
13E	13F		cV1 := cM	
13F	140		cV2 := ca1	
-----				
140	141		ca := cV1 + ( $\neg$ cV2)	
141	142		cpc'' := ca + 1	
142	143		cPC := ca3	
143	144		cM := cpc''	
144	145		ca := ca3	
145	146		ca3 := ca + 1	
146	147		cPC := cpc' + 1	
147	148		cpc' := cPC	
148	149		ca := cAD	
149	138		cAD := ca - 1	vers INS1
14A	14B	INSF:	ca := ca2	
14B	14C		cPC := ca + 1	
14C	14D		cM := cpc'	
14D	14E		cpc' := ca - 1	
14E	14F		cd := cpc'	
14F	150		cPC := cpc	
150	00		vers Entrée	
151	152	FINN:	ca := cd	
152	153		cPC := ca + 2	
153	154		ca1 := cM	
154	155		cpc' := cd	
155	156		cPC := cpc' + 1	
156	157		cV1 := cM	
157	158		ca2 := ca + ( $\neg$ cV1)	
158	159		ca := ca2	
159	15A		cd := ca + 1	
15A	15B		cPC := ca1	
15B	00		vers Entrée	
15C				
15D	15E	VERSN:	cpc := cPC	
15E	15F		cpc' := cPC + 1	
15F	160		cPC := cpc'	
-----				
160	161		ca := cM	
161	162		cpc'' := cpc	
162	163		cPC := ca + cpc''	
163	00		vers Entrée	

Une seule cellule attaque le bus adresse de la mémoire, c'est le PC. Une légère simplification de ce programme aurait pu être obtenue si une autre cellule, celle qui joue le rôle de I, en fait d, avait pu également l'attaquer. Cela aurait évité les échanges autour de PC pour passer de l'adresse du code programme à l'adresse des données et retour.

## DEMARRAGE D'UN CALCUL.

Lancer un calcul implique que l'ensemble du calculateur soit dans un état qui inclue la présence d'un code-programme, un jeu de données initiales, et des moyens d'accès à ces informations. Au delà de ce qui caractérise la tâche en cours de démarrage, il faut également que soit complètement défini tout ce qui concerne le mode de gestion du travail. Il faut donc disposer en mémoire, des codes qui définissent la tâche, ainsi que des moyens pour les atteindre, mais également les codes qui représentent la manière dont sera conduit le calcul. Nous touchons là, immédiatement aux problèmes de système.

Dans une première version simplifiée, je rajouterai simplement deux possibilités à la carte du tableau de commande, décrite P14 du B31. Les états 2 et 3 obtenus aux clefs CL1, CL2 et CL3 permettent de charger directement les registres PC et I.

Etat 2, noté avec la clef CL2 à un, les autres à zéro, le bus données visualise le contenu de la cellule PC. Appuyer sur la clé "ECRIT" enregistre en PC la quantité affichée aux 16 clés.

Etat 3, noté avec les deux clefs CL1 et CL2 à un, le bus données visualise le contenu de I, c'est-à-dire de la cellule d. Appuyer sur la clé "ECRIT" envoie la quantité affichée aux 16 clés en d.

Dans tous les cas cette version de base demeure utile pour le dépannage de la circuiterie. Mais dans une version plus élaborée c'est le système qui doit réaliser ces calculs qu'il doit mener à bien en puisant ses paramètres par consultation extérieure.

Le lancement du calcul est en conséquence précédé d'une étape de mise en place de l'information.

- 1) Passer en "manuel".
- 2) Mettre à l'état 0,
- 3) Charger une adresse, d'abord positionnée aux clefs, puis introduite avec "ECRIT".
- 4) Mettre à l'état 1,
- 5) "LECT" sélectionne la cellule-mémoire, "ECRIT" la charge avec ce qui est affiché aux clefs.
- 6) Répète 5) jusqu'à la fin de la liste à introduire.
- 7) Répète 2) jusqu'à la fin de toutes les listes à charger.
- 8) Mettre à l'état 2,
- 9) Charger I.
- 10) Mettre à l'état 3,
- 11) Charger d.
- 12) Vérifier toute l'information chargée.
- 13) Passer en automatique.

A suivre.

## VOUZZAVEDIBISAR

### Les bonnes recettes de Tantine Danielle.

Vous êtes de droite et vous voulez convertir un homme de gauche. Vous allez sans doute essayer de discuter, d'argumenter, d'exhiber des exemples probants, bref de convaincre ou de persuader selon votre tempérament.

Non non non!

Ecoutez Tantine Danielle: il vous faut mettre votre homme au pouvoir. Vous allez être émerveillé, d'un seul coup vous en aurez fait un parfait homme de droite. Mais surtout ne vous inquiétez pas, plus votre homme agira à droite, plus il se prétendra de gauche. Succès garanti.

Vous êtes de gauche, et vous voulez convertir un homme de droite. Là non plus ne tentez pas la conversion orale: aucune efficacité. Même Tantine Danielle n'a pas dans ce cas la recette miracle, mais quelques uns de ses tours de mains vous garantiront pas mal d'effets savoureux. D'abord il faut beaucoup de patience, la recette porte sur plusieurs années. Vous vous munissez d'un homme de droite de très bonne qualité, qui ait de la famille, une situation, de l'entregent, des ressources, bref une bonne surface sociale. Vous le faites baigner à feu doux dans une époque bien trouble. Vous le compromettez dans quelques histoires les moins ragôutantes possibles. N'ayez aucune inquiétude pour ce point ça marche en général très bien, il suffit qu'il y ait quelques profits immédiats à la clef. Il vous suffit alors d'attendre patiemment le prochain changement de régime. Les compromissions, les trafics avec l'ennemi, les tripatouillages divers et juteux, les déplacements de populations à têtes de bouc-émissaire pour masquer le tout vont faire de votre homme un personnage souple et prêt à se convertir à tout ce que vous voudrez. Au pire il deviendra centriste. Une infime minorité deviendra vraiment de gauche (voir alors la première recette), une autre infime minorité laissera sa peau dans le changement de régime.

Mais Tantine Danielle vous assure que pour l'essentiel, vous en avez pour quarante ans de succès moyens. Pendant cette période, votre homme refusera obstinément l'étiquette de droite, puis après, peu à peu, quand les choses se seront tassées, il faudra recommencer.

E. B.

Qu'est-ce qu'un robot ?

C'est un être d'une organisation interne complexe, capable de réaliser des tâches spécifiques d'un assez grand degré de complexité, dont la proportion de matière organique qui le constitue s'accroît sans cesse, disponible à merci, et surtout qui ne touche aucun salaire.

Dernière brève.

On constate un étrange phénomène depuis peu, des chômeurs de haut niveau, d'anciens grands responsables de grandes entreprises qui, pour des raisons diverses ne trouvent plus d'emploi, acceptent d'exercer gratuitement des responsabilités.

