Un programmateur d'EPROM universel le PRM 4

ès que l'on touche un tant soit peu à l'électronique «programmée», le besoin de programmer des EPROMs se fait rapidement sentir. Nous avons développé le PRM 4 dans cet esprit.

Il existe à l'heure actuelle une grande diversité d'EPROMs, le PRM 4 est conçu pour prendre en compte tous les types existants jusqu'aux 27128. Si le côté Hard ne nous a pas posé de problème, l'aspect logiciel, très important pur une réalisation de ce type, nous en pose. Il est en effet inconcevable de mobiliser environ seize pages de votre revue pour publier un listing. De plus ce dernier a été étudié spécifiquement

Par ailleurs, une version IBM PC qui tourne d'ores et déjà vous sera proposée après les adaptations prévues pour Apple, ATMOS et Commodore 64.

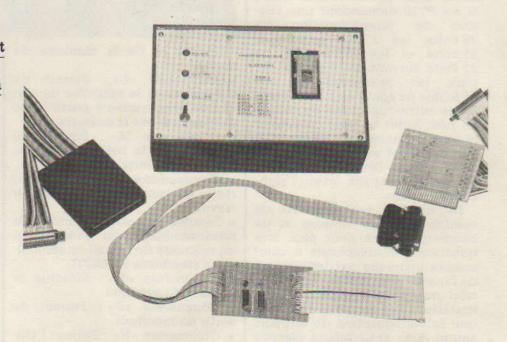
pour les Commodore 4032, 8032 et

8296. Ne vous inquiétez pas, il est

facilement transposable sur tout

micro doté d'un 6502.

Dès le mois prochain, nous vous onnerons plusieurs solutions pour vous procurer le logiciel.
En attendant, étudiez bien le matériel...



Présentation du PRM 4

vec le PRM 4 nous pensons avoir réalisé un programmateur d'EPROMs des plus performants car il utilise toutes les ressources possibles en la matière malgré le recours à des solutions techniques accessibles à l'amateur. Ceci n'a d'ailleurs aucun caractère restrictif mais il faut savoir qu'il n'est pas rare de trouver dans le commerce des appareils de performances équivalentes pour la bagatelle de 10 à 20000 de nos francs! Nous tenons tout de suite à vous rassurer, la réalisation du PRM 4 ne vous demandera pas un tel effort financier, une fraction d'une telle somme suffira bien. Pourtant les

performances sont bien là. Qu'on en juge :

— Programmation des mémoires de 2 à 16 K/octets (2716, 2532, 2732, 2732A, 2764 et 27128).

— Configuration du support par logiciel sous contrôle d'un moniteur performant.

— Emploi d'un algorithme rapide pour la programmation des 2764 et 27128 (5 ms par octet contre 50 pour les autres EPROMs).

— Compatible «hard» avec les ordinateurs équipés de 6800, 6809 et 6502.

— Moniteur très puissant entièrement écrit en langage-machine.

A ce propos, nous vous rappelons que ce moniteur est conçu pour les machines Commodore mais assez aisément transposables pour les micro-ordinateurs dotés de 6502.

Nous vous proposerons après, les cartes d'adaptation hard destinées à Apple 2, Atmos et Commodore 64.

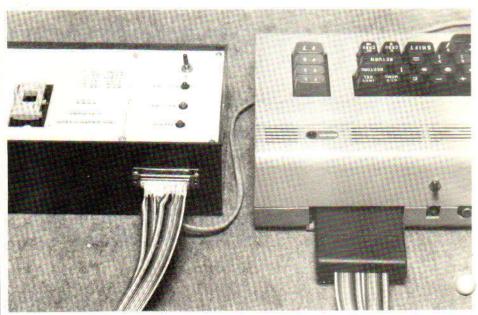
Nous verrons par la suite une version pour IBM PC qui reprend les mêmes cartes, avec une carte d'adaptation spécifique comme pour Oric et Apple et avec un logiciel spécifique (en basic compilé cette fois).

Maintenant que nous vous avons bien alléchés, il est bon de revenir sur quelques points qui nous ont guidés lors de la conception du PRM 4. Tout d'abord, la programmation d'EPROMs aussi diverses complique, on s'en doute bien, la réalisation. Ainsi, si les 2716 demandent une tenprogrammation sion de de 25 volts, les 2732A requièrent 21 volts et les 2764, 12,5 ou 21 volts suivant les modèles. De plus, les brochages de ces EPROMs ne sont pas tous compatibles entre-eux ce qui ne simplifie rien. Enfin, et pour couronner le tout, la tension d'alimentation Vcc des 2764 et 27128 doit être de 6 volts au lieu de 5 en phase de programmation avec l'algorithme rapide.

Devant une telle quantité de combinaisons de tension et de brochages, il ne peut plus être question d'interrupteurs à commande manuelle car les risques de fausses manoeuvres sont bien trop grands. La gestion des commutations des tensions de même que la configuration du support seront donc entièrement placées sous le contrôle du moniteur sans aucune intervention de l'utilisateur.

Le moniteur chargé de gérer tout ce petit monde se doit de disposer de commandes puissantes afin de conférer à l'ensemble une efficacité optimale. Celui que nous avons mis au point dispose des commandes suivantes: • Commande «M»: Permet l'affichage en hexadécimal (DUMP) de la mémoire de travail.

- Commande «V»: Permet de vérifier que l'EPROM est bien effacée.
- Commande «T»: Permet de transférer le contenu de l'EPROM dans la mémoire de travail.
- Commande «P»: Permet de transférer le contenu de la mémoire de travail sur l'EPROM.
- Commande «C»: Permet la comparaison des contenus de



l'EPROM et de la mémoire de travail.

- Commande «L»: Permet de charger dans la mémoire de travail un programme à partir de la disquette ou de la cassette.
- Commande «S»: Permet de sauvegarder la mémoire de travail sur disquette ou cassette.
- Commande «:» : Permet d'altérer le contenu de la mémoire de travail
- Commande «D»: Permet le désassemblage du programme en mémoire de travail (mnémoniques standards du 6502).
- Commande «R»: Effectue le «reset» du système.
- Commande «B»: Permet de sortir du moniteur.
- Commande «H» : Permet l'affichage d'un aide-mémoire.
- Commande «E» : Permet l'effacement de la mémoire de travail.

On voit donc que les commandes sont nombreuses et que vous avez sous la main tous les outils dont on peut réver pour programmer des EPROMs. Nous reviendrons bien entendu sur ce point lors de la description du logiciel.

A ce sujet, le moniteur décrit tourne sur les micro-ordinateurs COMMODORE versions 4032 (40 colonnes), 8032 et 8296 (80 colonnes) et une version a été développée pour le COMMODORE 64. La version dont le programme sera détaillé est celle prévue pour le C.B.M. 8296 et constitue la version la plus «portable» du moniteur ce qui fait que les possesseurs d'ORIC, APPLE et toute machine à base de 6502 n'auront aucun mal à l'adapter à leur

micro. Dans ce but, certaines routines faisant partie de la bibliothèque du C.B.M. 8296 ont été ré-écrites en totalité afin de n'utiliser qu'un minimum de routines-système.

Les présentations étant faites, nous vous proposons d'examiner les caractéristiques des EPROMs que peut programmer le PRM 4 avant d'aborder l'étude des schémas et de la réalisation proprement dite. Pour ce faire nous avons regroupé sur deux tableaux les brochages et les signaux des EPROMs concernées suivant que l'on se trouve en phase de lecture ou d'écriture.

Le tableau de la figur 1 regroupe les brochages et 🐸 signaux des différentes EPROMs en phase de lecture. La «compatibilité» de ces mémoires est, comme on le voit, tout à fait relative puisqu'en dehors des broches Do/D7, Ao/A10 et de la masse (Vss), toutes les autres diffèrent suivant le modèle. Si nous prenons l'exemple d'une 2716, la lecture d'une donnée s'opère en portant OE barre à 0 après avoir sélectionné une adresse et positionné CE barre à l'état 0. Dans tous les cas de figures, la tension de programmation VPP doit être égale à Vcc soit 5 volts. Nous avons fait figurer sur ce tableau les 27256 de 32 K/octets que le moniteur ne peut gérer dans sa version actuelle mais qui ont é+é prévues au niveau hard sur PRM 4 pour une version future puisqu'elles commencent à être disponibles sur le marché.

Le tableau de la figure 2

27256	27128	2764	2732	2532	2716
VPP (5 V)	VPP (5 V)	VPP (5 V)		-	_
A12	A12	A12		_	_
A ₇					
A ₆					
A ₅	A ₅	A ₅	A5	A ₅	A ₅
A ₄					
Аз	A ₃	A ₃	A3	A ₃	A3
A ₂					
A ₁	A ₁	A ₁	Aı	A ₁	A ₁
Ao	Ao	Ao	Ao	Ao	Ao
D ₀					
D ₁					
D ₂					
Vss	Vss	Vss	Vss	Vss	Vss

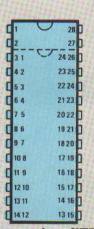
	ш			
C	1	U	28	b
C	2		27	D
0	3 1		24 25	b
0	42		23 25	b
0	53		22 24	b
0	6 4		21 23	b
0	7 5		2022	b
0	8 6		19 21	1
0	9 7		1820	0
0	10 8		17 19	b
0	11 9		16 18	b
C	11 9 12 10		15 17	1
0	13 11		14 16	0
0	14 12		13 15	1
		1	ED	-

2716	2532	2732	2764	27128	27256
-		-	Vcc	Vcc	Vcc
_	_	-	PGM (1)	PGM (1)	A14
Vcc	Vcc	Vcc		A13	A13
As	As	As	A ₈	As	A8 .
A ₉	A ₉	A ₉	A ₉	As	As
VPP (5 V)	VPP (5 V)	A11	An	An	A11
OE T	OEU	OEU	OEU	OET	OE U
A10	A10	A10	A10	A10	A10
CE (0)	An	CE (0)	CE (0)	CE (0)	CE (0)
D 7	D ₇	D ₇	D ₇	D7	D ₇
D ₆	D ₆	D ₆	D ₆	D ₆	D ₆
D ₅	D ₅	D ₅	D ₅	D ₅	D ₅
D ₄	D ₄	D ₄	D ₄	D ₄	D ₄
D ₃	D ₃	D ₃	D ₃	D ₃	D ₃

Lecture des EPROM

Figure 1 - Brochage et signaux des principaux types d'EPROM en phase de lecture.

27256	27128	2764	2732	2532	2716
Vpp	VPP	VPP		_	
A12	A12	A12	_	_	_
A ₇	A ₇	A7	A ₇	A ₇	A ₇
A ₆	A6	A ₆	A ₆	A ₆	A ₆
A ₅	As				
A ₄					
Аз	Аз	Аз	A ₃	Аз	Аз
A ₂					
A ₁					
A ₀	Ao	A ₀	Ao	A ₀	Ao
D ₀	D ₀	D ₀	D ₀	Do	Do
D ₁	D ₁	D_1	D ₁	D ₁	Di
D ₂					
Vss	Vss	Vss	Vss	Vss	Vss



2716	2532	2732	2764	27128	27256
1-1	-	-1	Vcc	Vcc	Vcc
_	_		PGM U	PGM U	A14
Vcc.	Vcc	Vcc		A13	A13
As	A ₈	A ₈	As	As	A ₈
A ₉	A ₉	A ₉	As	A ₉	A ₉
VPP	Vpp	A11	A11	A11	A11
OE (1)	OE U	VPP	ŌE (1)	ŌĒ (1)	ŌĒ (1)
A10	A10	A10	A10	A10	A10
CEIL	A11	CEU	CE (0)	CE (0)	CEU
D ₇	D ₇	D ₇	D ₇	D ₇	D ₇
D ₆	D ₆	D ₆	D ₆	D ₆	D ₆
D ₅	D 5	D ₅	D ₅	D ₅	D ₅
D ₄	D ₄	D ₄	D ₄	D ₄	D ₄
D ₃	D ₃	D₃	D ₃	D ₃	D ₃

Programmation d'EPROM

Figure 2 - Brochage et signaux des principaux types d'EPROM en phase d'écriture.

brochages regroupe les maux des EPROMs mais cette s en phase de programmation. La diversité est encore plus grande qu'en lecture puisqu'ici tout change ou presque suivant les modèles. L'impulsion de programmation est appliquée soit

sur OE barre (2532), soit sur CE barre (2716, 2732, 27256) ou sur PGM barre (2764 et 27128). Pour simplifier les choses, cette impulsion est négative sur tous les modèles sauf la 2716 où elle est positive. Enfin, et pour couronner le tout, la tension de programmation VPP varie de 25 volts pour les «anciennes» mémoires (2716, 2532 et 2732) à 21 ou 12,5 volts pour les plus récentes (2332A, 2764, 27128 et 27256).

Pour programmer une EPROM, il faut appliquer une impulsion d'une durée de 50 ms sur l'entrée

METRIX OX 710 C 2 x 15 MHz

+ port 48 F

- Testeur incorporé pour le dépannage rapide et la vérification des composants (résistances, condensateurs, selfs, semiconducteur).

 Le testeur de composants présente les courbes courant/tension sur les axes à 90°.

 Le mode de sélection alterné choppé est commuté par le choix de la vitesse de la base de temps.

- muté par le orion.

 Ecran de 8 x 10 cm.

 Le tube cathodique possède un réglage de rotation de trace pour compenser l'influence du
 champ magnétique terrestre.

 Bande du continu à 15 MHz (— 3 db).

 Senstionnement en XY.

- Fonctionnement en XY.
 Inversion de la voie B (± YB).
 Fonction addition et soustraction (YA ± YB).

DISTRIBUÉ PAR

ACER COMPOSANTS 42, rue de Chabrol 75010 PARIS. Tél. : (1) 47.70.28.31





REUILLY COMPOSANTS 79, bd Diderot, 75012 PARIS. Tél.: (1) 43.72.70.17 De 9 h à 12 h 30 et de 14 h à 19 h

de programmation après avoir sélectionné une adresse et une donnée, la tension VPP étant à sa valeur nominale. La durée de cette impulsion conduit à des temps de programmaiton qui peuvent aller de 102,4 secondes pour une 2716 à 1638,4 secondes pour une 27256 soit près de 28 minutes! Une telle durée est évidemment énorme et risque de faire perdre patience à plus d'un. Il faut d'ailleurs ajouter à cela le temps que peut mettre le logiciel à gérer le programmateur d'où le recours obligatoire au langagemachine pour ce genre d'application où un Basic interprété fait figure d'escargot. La firme INTEL a donc mis au point un algorithme de programmation rapide pour les EPROMs de la nouvelle génération (2764, 27128 et 27256) dont le gain en vitesse est spectaculaire puisqu'il est de 1 à 10 en moyenne. Nous avons donc utilisé cet algorithme ce qui est, sauf erreur ou omission, une première en ce qui concerne une réalisation amateur.

Contrairement à ce que l'on pourrait croire, le principe en est très simple comme le montre l'ordinogramme de la figure 3. Il consiste à envoyer une impulsion de 1 ms, puis à relire l'EPROM pour vérifier que l'octet a bien été programmé. Si ce n'est pas le cas, on recommence et au bout de 15 essais au maximum, on impulsion de envoie une 4 x 15 x 1 ms soit 60 ms et on vérifie une dernière fois. En cas d'échec, on sort du programme car l'on suppose qu'au bout d'une telle durée une EPROM doit être programmée. Si la programmation s'est effectuée avant les 15 essais fatidiques, on envoie une impulsion égale à 4 fois le nombre d'essais. La durée minimum de programmation d'un octet est donc de 1 ms + (4 x 1 ms) soit 5 ms ce qui explique le gain de 1 à 10 évoqué plus haut. Afin d'assurer une sécurité de programmation aussi bonne avec cet algorithme que par le système classique, il faut porter la tension Vcc à 6 volts ce qui augmente de 1 volt le seuil d'état des lignes de données. Seules les 2764, 27128 et 27256 étant conçues pour supporter une telle tension, l'algorithme n'est applicable, hélas, qu'à ce type d'EPROM.

La revue de détail des principa-

les EPROMs ayant été effectuée, nous pouvons aborder sans plus tarder l'étude des schémas du PRM 4.

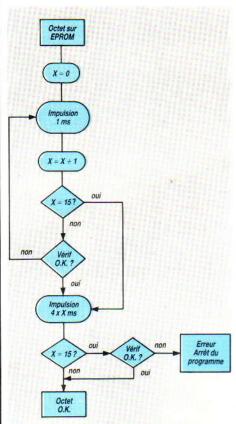


Figure 3 - Ordinogramme de la programmation d'un octet à l'aide de l'algorythme rapide.

Étude des schémas

a conception d'un programmateur d'EPROMs de classe professionnelle tel que le PRM 4 à l'aide de moyens techniques et financiers du domaine de l'amateur n'est pas une tâche spécialement aisée mais nous pensons cependant y être parvenu. Pour des raisons évidentes de mise en page, nous avons scindé en trois parties les schémas du PRM 4 ce qui ne nuit en rien à la compréhension de l'ensemble bien au contraire.

Le programmateur

C'est, en quelque sorte, l'âme du PRM 4 puisque cette partie du montage, visible sur la **figure 4**, est chargée de gérer l'ensemble du montage. Nous pouvons distinguer 4 parties distinctes:

1) L'interface (IC₁, IC₂ et IC₃) chargée de la liaison entre le reste du montage et l'ordinateur.

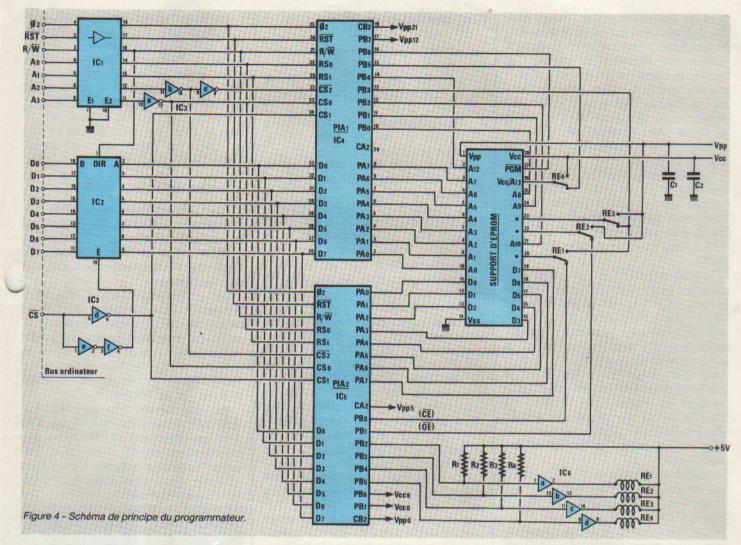
- 2) Les deux PIA (IC4 et IC5) qui pilotent l'ensemble des fonctions.
- 3) Le support d'EPROM.

4) La circuiterie de configuration du support.

Le PRM 4 devant pouvoir être raccordée (au départ) sans acrobaties à tout ordinateur basé sur un microprocesseur de la famille 6800, 6809 ou 6502, nous avons «bufferisé» les signaux chargés de le piloter. Les signaux PHI 2, RST barre, R/W barre et les adresses A₀/A₃ sont donc mis en forme par IC1 qui est un 74LS541. décodage d'adresses est conçu de telle manière que le PIA1 occupe les 4 premiers octets de l'adresse-mémoire où sera implanté le PRM 4 et le PIA2 les 4 suivantes. La sélection de l'un ou l'autre des PIA s'opère par tri sur le bit 2 du bus d'adresses grâce aux inverseurs IC3b, IC3c qui pilotent l'entrée CS2 barre de PIA1 ou de PIA2. De plus, les adresses supérieures à \$XXX7 sont rendues inaccessibles par le tri sur le bit 3 du bus d'adresses grâce à IC3a dont la sortie est reliée à CS0 des deux PIA. La troisième entrée de décodage d'adresse est fournie par le signal CS barre qui gère CS1 des PIA via IC3d. Pour nous résumer, le PRM 4 occupera les 8 adresses consécutives suivant l'adresse mémoire déterminée par l'état de CS barre qui jour le rôle de pilote.

Ainsi, dans le cas du C.B.M 4032, 8032 ou 8296, le PRM 4 ser logé en \$9000/\$9007, l'entrée CS-barre étant reliée à la broche CS-barre de l'ordinateur qui détecte les adresses comprises entre \$9000 et \$9FFF. Pour le COMMO-DORE 64, CS barre sera reliée à un interface spécialisé logeant le PRM 4 en \$DE00/\$DE07 et pour l'ORIC ATMOS, un dispositif similaire le placera en \$0320/\$0327.

La transmission des données est, elle également, mise en forme par IC2 qui est un 74LS245. L'entrée CS barre valide le circuit via IC3e et IC3f afin d'éviter tout conflit sur le bus de données et la direction des échanges est gérée par la ligne R/W barre (0 en écriture, 1 en lecture). Le fait d'avoir mis en forme les signaux aboutissant aux PIA est une sécurité pour les PIA comme pour l'ordinateur mais il ne faut pas croire pour autant que vous pourrez placer le PRM 4 à deux mètres de



l'unité centrale! En effet, la fréquence des signaux qui pilotent appareil est de 1 MHz et à cette fréquence un câble de 50 centimètres constitue la longueur maximum.

Les PIA que nous avons employés sont des modèles bien connus et, de plus, très économi-

ques puisqu'il s'agit des 6821 ou 6520 que nous avons assez longuement décrits dans ces pages (Radio Plans E-L Nº 450). Pratiquement toutes les lignes d'entrées/sorties sont utilisées y compris CA2 et CB2, c'est dire que leur mission est rude!

Le port A du PIA1 (IC4) gère les

Ao/A7 du adresses support d'EPROMs alors que le port B s'occupe des adresses A8/A13, de PGM barre (A₁₄ sur les 27256) et du pilotage de VPP 12. Si CA2 n'est pas utilisée, CB2 gère l'information VPP 21 qui sera évoquée plus loin.

Le port A de PIA2 (IC₅) gère le



bus de données et est, de ce fait, bi-directionnel.Le port B est, lui, largement sollicité puisqu'il pilote les lignes CE barre, OE barre, la commande de la circuiterie de configuration du support et les lignes Vcc Ø et Vcc 6. CA2 commande la ligne Vpp 5 et CB2, la ligne Vpp 0 complétant ainsi la gestion des alimentations Vcc et Vpp du support.

La configuration du support est confiée à 4 relais en boitier DIL munis de contacts repos/travail dont les bobines sont commandées par les lignes PB2/PB5 du PIA2 via le circuit IC6 du type

Cette solution n'est peut-être pas la plus économique car ce genre de relais est assez coûteux mais c'est assurément la plus efficace car elle permet une commutation par logiciel des broches du support sans erreur et en évitant tout mélange de tensions. Il ne faut pas oublier, en effet, que le fait de relier une broche de l'EPROM à VPP conduirait à la destruction de cette dernière et que quelques microsecondes suffisent.

Suivant le type d'EPROM choisi, l'état des relais varie et les tableaux des figures 1 et 2 devraient vous aider à y voir clair. Pour vous aider à vous y retrouver dans la configuration des broches du support suivant le type d'EPROM choisi, nous vous invitons à examiner le tableau suivant où sont indiqués les états des relais et des sorties PB2/PB5 du PIA2. Le collage du relais est symbolisé par le signe «*» ce qui implique que la sortie correspondante du port B du PIA2 se trouve alors à l'état «0».

EPROM	2716	2532	2732	2764	27128
RE1	100	*			
RE2			*		
RE3	*	*			
RE4	*	*	*		

Avant de passer à la suite de l'étude des schémas, il est utile de savoir qu'entre chaque commande, le support est placé hors tension sur toutes ses broches ce qui permet de manipuler les EPROMs sans aucun danger.

L'alimentation VCC

Le PRM 4 a besoin de 3 sources de tensions pour fonctionner :

— 5 volts sous 300 à 400 mA pour l'alimentation des circuits intégrés.

 Une source Vcc programmable délivrant 5 ou 6 volts sous 100 mA pour les EPROMs.

— Une source de tension de programmation délivrant 12,5, 21 ou 25 volts sous 30 mA.

Le schéma de la **figure 5** montre l'alimentation + 5 volts et l'alim Vcc programmable. Le + 5 volts est obtenu de la manière la plus simple qui soit du fait de l'emploi du régulateur intégré IC₇ qui est du type 7805 en boitier TO 220.

L'alimentation programmable Vcc comporte un commutateur statique (T2, T3 et T4) suivi d'un régulateur de tension ajustable (T₁ et IC₈). Le principe de fonctionnement du commutateur statique est on ne peut plus élémentaire et il est facile de comprendre que la saturation de T2 entraine la disparition de la tension entre la base du Darlington T₃/T₄ et la masse ce qui fait que nous retrouvons une tension nulle sur l'émetteur de T4. Par conséquent, nous obtenons environ 12 volts sur «E» de IC⁸ quand Vcc o est à l'état 0 et une tension nulle dans le cas contraire. Un voyant vert (LD1) relié à T2 signale l'absence de tension Vcc indiquant ainsi que l'EPROM peut être manipulée sans danger. Le régulateur intégré IC₈ est du type LM 317T et T₁ shunte AJ₂ lorsque l'entrée V_{CC 6} est à l'état 1. Suivant l'état de V_{CC 6}, nous obtenons donc en sortie de IC₈ une tension de 5 ou 6 volts parfaitement régulée. Un voyant jaune (LD2) signale à l'utilisateur que le PRM 4 est en phase active et qu'il ne faut pas manipuler l'EPROM.

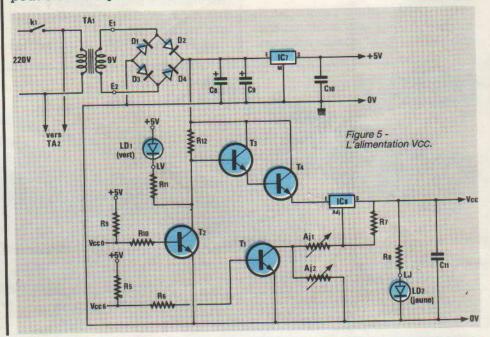
L'alimentation VPP

Cette alimentation, visible sur la **figure 6**, est chargée de fournir la tension de programmation des EPROMs V_{PP} et doit délivrer 0, 5, 12,5, 21 ou 25 volts suivant les cas de figures.

Le commutateur statique reprend exactement le même principe que précédemment et est basé sur T7 et le couple T5/T6. Nous obtenons donc une tension nulle sur «E» de IC9 quand VPP 0 est à l'état «1» et environ 35 volts si VPP 0 est à l'état 0.

Le régulateur de tension IC9 (LM 317T) est piloté par 3 transistors qui mettent en ou hors circuit les ajustables Aj3 à Aj6. Suivant l'état des entrées VPP 5, VPP 12 et VPP 21, nous obtenons en VPP les tensions suivantes :

Vpp0	Vpp5	Vpp12	Vpp21	Vpp
1	X	X	X	0 Volts
0	1	X	X	5 Volts
0	0	1	X	12,5 Volts
0	0	0	1	21 Volts
0	0	0	0	25 Volts



Un petit montage permet de signaler par l'allumage d'un voyant rouge la présence de la tension de programmation. La diode Zener D₉ dont le seuil est de 7,5 volts autorise donc la saturation de T₁₁ pour toute tension V_{PP} supérieure à cette valeur et, partant de là, l'allumage de LD3.

L'étude des schémas est terminée et vous pouvez constater que le PRM 4 n'a rien d'une «usine à gaz» et que nous n'avons pas appliqué le principe cher à certains du «pourquoi faire simple quand on peut faire compliqué?». Cette simplicité ne signifie pas pour autant que nous avons négligé la qualité et la fiabilité du montage et nous pouvons vous assurer que l'appareil n'a pas à rougir, bien au contraire, face aux réalisations du commerce!

La partie théorique de cette description étant achevée, passons à présent aux choses sérieuses en réalisant le PRM 4.

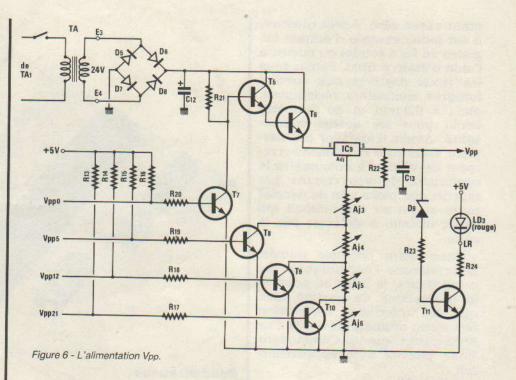
Réalisation

e PRM 4 devant pouvoir être réalisé par tout amateur disposant d'un minimum d'équipement, nous n'avons pas employé de techniques sophistiquées ni de composants chers autant qu'introuvables. Tout le monde devrait donc être satisfait.

Les circuits imprimés

Afin de réduire l'encombrement du PRM 4 de manière à loger tout le montage dans un boitier TEKO P4, nous avons dû employer deux circuits imprimés. Le premier supporte l'ensemble des composants du programmateur et est réalisé en double face car les liaisons sont nombreuses. Le second est plus simple à réaliser puisqu'il s'agit d'un simple face et comprend les composants des alimentations. Quoi qu'il en soit, les tracés ne sont pas trop difficiles à reproduire par les méthodes classiques.

Les tracés du recto et du verso du circuit A sont donnés sur les figures 7 et 8 et nous vous conseillons vivement l'emploi de la méthode photographique vu la finesse des pistes. Le circuit B de la figure 9 peut, quant à lui, parfaitement être réalisé à l'aide des transferts et du feutre, le tracé



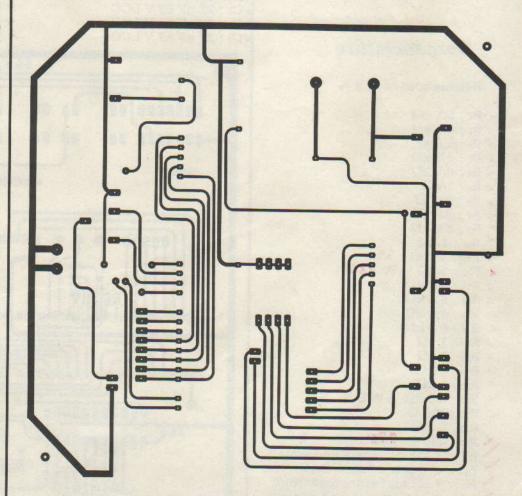


Figure 7 - Le circuit A côté composants.

étant assez aéré. Après gravure, il est indispensable d'étamer les pistes au fer à souder ou mieux, à l'aide d'étain à froid. Percez tous les trous destinés aux circuits intégrés, transistors, résistances, etc... à 0,8 mm et au diamètre voulu pour les autres composants. Avant d'installer ces derniers, nous vous conseillons vivement de vérifier à l'ohmmètre la conformité de vos circuits par rapport aux plans afin de déceler toute erreur ou court-circuit qui serait difficile à détecter par la suite.

Dans notre prochain numéro nous verrons l'implantation des composants, la mise au point et le programme de gestion ainsi que les interfaces nécessaires aux micro-ordinateurs dotés d'un 6502 autres que les Commodore. Vous pouvez déjà approvision-

A bientôt.

P. Wallaert

Nomenclature

Résistances 1/4 W 5 %

« R1: 6.8 kΩ R2: 6,8 kΩ « R₃: 6,8 kΩ R₄: 6,8 kΩ < R₅: 6,8 kΩ

R7: 220 Ω R₈: 330 Ω « R9: 6,8 kΩ

R₁₀: 1 kΩ R₁₁: 330 Ω 330 R₁₂: 1 kΩ ½ W 220

× R₁₃: 6,8 kΩ R₁₄: 6,8 kΩ

R15: 6,8 kΩ R₁₆: 6.8 kΩ

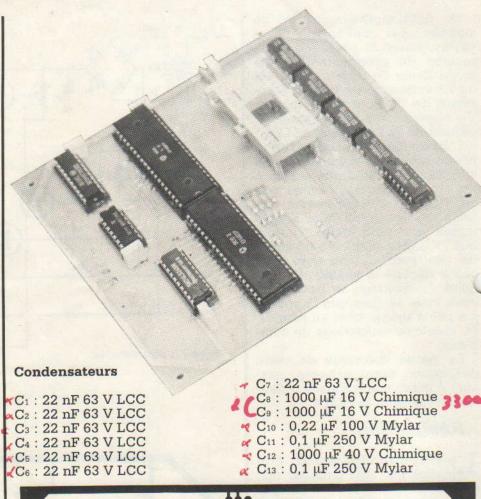
 $R_{17}: 1 k\Omega$ R₁₈: 1 kΩ $R_{19}:1 k\Omega$

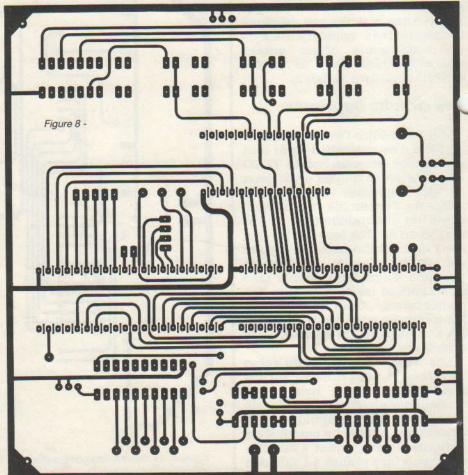
R20: 1 kΩ R₂₁: 3,3 kΩ ½ W 5 %

R22: 220 Ω $R_{23}: 2.2 \text{ k}\Omega$ R24: 330 Ω 230

 $AJ_1: 1 k\Omega$ miniature horizontal AJ_2 : 470 Ω miniature horizontal AJ_3 : 1 k Ω miniature horizontal AJ₄: 2,2 kΩ miniature horizontal

AJ₅: 2,2 kΩ miniature horizontal AJ₆: 1 kΩ miniature horizontal





Programmateur d'EPROM

Semi-conducteurs Circuits intégrés

√ T₄: BD 135, BD 139

X T5: BC 237B

A T6: BD 135, BD 139

T7: BC 237B
T8: BC 237B
T9: BC 237B
T10: BC 237B

D1: 1N 4002 200 V 1A
 D2: 1N 4002 200 V 1A
 D3: 1N 4002 200 V 1A
 D4: 1N 4002 200 V 1A
 D5: 1N 4002200 V 1A

D₆: 1N 4002 200 V 1A D₇: 1N 4002 200 V 1A D₈: 1N 4002 200 V 1A

∠D₉: Zener 7,5 V 0,4 W LD₁: LED verte diam 5 mm

 LD_2 : LED jaune diam 5 mm LD_3 : LED rouge diam 5 mm

IC₁: 74 LS 541 ✓IC₂: 74 LS 245 ✓IC₃: 74 LS 04 ▼IC₄: PIA 6821, 6520

IC₈: 7805 boitier TO 220 IC₈: LM 317 T boitier TO 220 IC₉: LM 317T boitier TO 220

Divers

RE1: relais CELDUC 5 V 1RT boitier DIL 14

RE2: relais CELDUC 5 V 1RT boitier DIL 14

RE3: relais CELDUC5 V 1RT boi-

tier DIL 14 RE₄ : relais CELDUC 5 V 1RT boi-

tier DIL 14 RE₄: relais CELDUC 5 V 1RT boi-

tier DIL 14
K1: inter unipolaire miniature

K₁: inter unipolaire miniature 250 V 1A

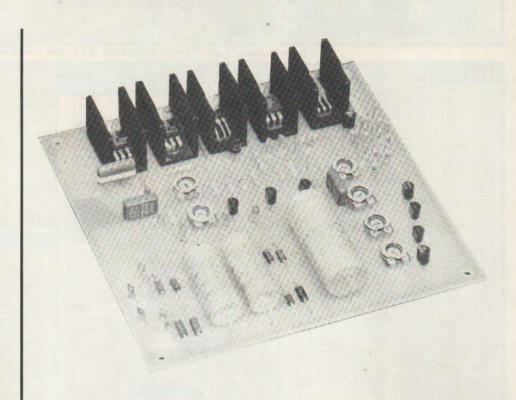
TA₁: transfo 220/9 V 5 VA TA₂: transfo 220/24 V 2 VA

►2 supports pour circuits intégrés 40 broches

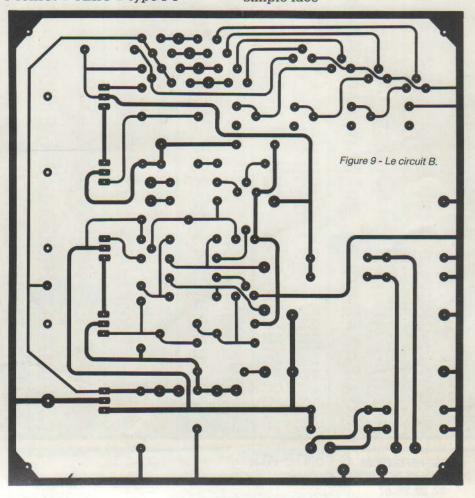
2 supports pour circuits intégrés 20 broches

≯2 supports pour circuits intégrés 14 broches

✓ 1 support à wrapper 28 broches
1 support insertion nulle 28 broches
1 connecteur male 25 broches DB



1 connecteur femelle 25 broches 1 circuit imprimé « A » époxy double face
20 cosses « poignard » 1 circuit imprimé « B » époxy simple face



25

Nomenclature du PRM 4 carte d'interface IBM PC

COMPLÉMENT AU Nº 471

Circuits intégrés

X IC1: 7493 X IC2: 74 L537 IC3: 74 LS 28 X IC4: 74121 X IC5: 74 LS 244 IC6: 74 LS 21 X IC7: 74 LS 04 X IC8: 74 LS 244 IC9: 74 LS 244

₩ IC9: 74 LS 245

Condensateurs

C₁: 0,1 μF MKH C₂: 39 PF

√ • C₃: 0,1 µF MKH ▶ C4: 0,1 μF MKH

C5: 39 pF

I NE

Résistances 1/4 W 5 %

× • R1 : 2.2 kΩ « R₂ à R₁₃ : 3,3 kΩ R14: Voir texte K AJ1: 47 kΩ

Diode

× 1N4148

Divers

1 prise CANON DB 25 pour CI Cavaliers Équerre de fixation pour prise DB 25

Un programmateur d'EPROM universel le PRM 4 (suite)

ous poursuivons dans ce numéro la réalisation du PRM 4 entamée le mois dernier.

Cet article cloture la réalisation
pratique de l'ensemble de base et
présente le logiciel. A ce propos,
nous rappelons que ce dernier ne
sera pas présenté dans la revue
sous forme de listing. Dans un
premier temps, vous devrez en
faire la demande par l'intermédiaire
de la rédaction, puis si les
demandes sont suffisantes, nous le
ferons distribuer.

Enfin, dans les deux prochains numéros, nous décrirons plusieurs interfaces pour adapter le PRM 4 à quelques machines courantes : autre Commodore, Apple 2, ATMOS et même PC/IBM.

En vous aidant des figures 10 et 11, implantez les composants en débutant par les traversées entre pistes qui sont signalées par «.». N'oubliez pas de souder les pastilles au recto et au verso du circuit A et montez le support à wrapper 28 broches de manière à ce que sa face supérieure soit située à 15 mm du circuit, le support à force d'insertion nulle s'enfichant ensuite dans celui-ci. Si vous décidez de vous passer de ce type de support pour des raisons économiques, placez la face supérieure du support à wrapper

à 20 mm du circuit A. Les sorties sur le circuit B se font à l'aide de cosses «poignard» et seules les broches 0 V, + 5 V, VPP et Vcc du circuit A sont à munir de ce type de cosses lesquelles doivent être soudées côté cuivre.

Avant d'installer les circuits dans le coffret, il est conseillé de vérifier les deux circuits à l'ohmmètre afin de détecter toute erreur. Cette opération doit se faire sans les circuits intégrés de A et vous évitera de longues recherches en cas d'insuccès lors des essais.

Montage du PRM 4

Le montage des éléments à l'intérieur du coffret P4 ne pré-

sente aucune difficulté et nous vous conseillons de vous inspirer de la figure 12 et des photos pour mener à bien l'opération. Le circuit A est fixé au couvercle du boîtier et 4 boulons vous permettent de le positionner correctement. Soignez la découpe du support d'EPROMs et assurez-vous qu'il n'y a pas de court-circuit entre les broches et la tôle. Pour la décoration du couvercle, nous vous conseillons l'emploi d'une feuille de carton à dessin collée plutôt que d'exécuter le lettrage directement sur la tôle, l'effet n'en sera que plus flatteur. Une feuille de plastique adhésif transparent sera ensuite collée améliorant ainsi la présentation et protégeant efficacement votre travail.

En vous aidant des schémas, cablez le montage en utilisant du fil souple de 6/10 pour les liaisons

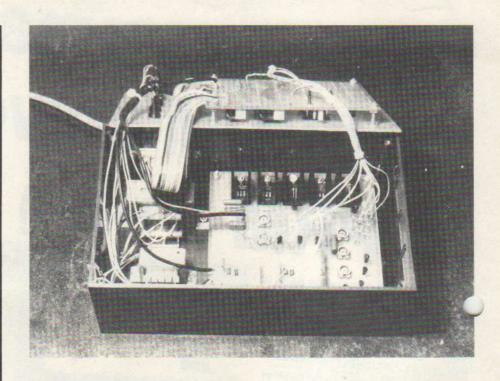
entre les transfos et le circuit A et les liaisons 0 V, + 5 V, Vcc ET VPP entre les deux circuits. A l'aide de cable plat de 0,4, reliez le circuit A à la prise 25 points en suivant les indications de la figure 13 et en reliant par 8 fils les masses de la prise au 0 V du circuit A. A l'aide de fil de même section, cablez le reste du montage en laissant suffisamment de longueur pour faciliter la maintenance. Après cette dernière opération s'achève la réalisation du PRM 4 et nous pouvons à présent procéder aux essais.

Mise en service

Les essais du PRM 4 vont se dérouler en deux temps :

- 1) Les tests sans l'aide du moniteur.
- 2) Les tests avec l'aide du logiciel.

La première phase de tests va donc se faire sans que le PRM 4 ne soit relié à l'ordinateur, le logiciel n'ayant pas encore été étudié. Otez tout d'abord tous les



circuits intégrés du circuit A et | vérifiez que vous obtenez 5 volts à 2% près entre +5V et la masse. Si tout va bien et, sauf

cas, la diode LD1 doit s'allumer. Reliez Vcc o à la masse et constatez l'extinction de LD1, l'allumage de LD2 et l'apparition d'une tenerreur de cablage, ce doit être le sion en Vcc. Réglez AJ1 pour obte-

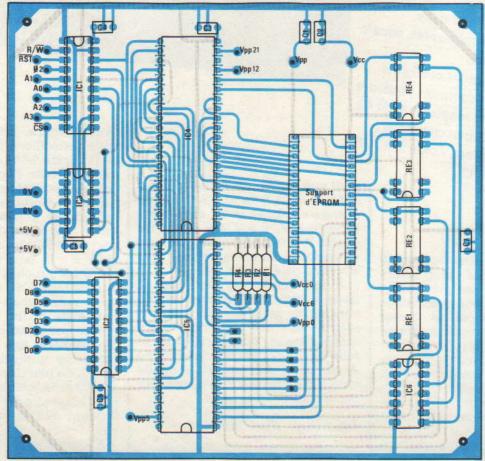


Figure 10 - Implantation des composants sur le circuit A. Les points indiquent les traversées entre pistes.

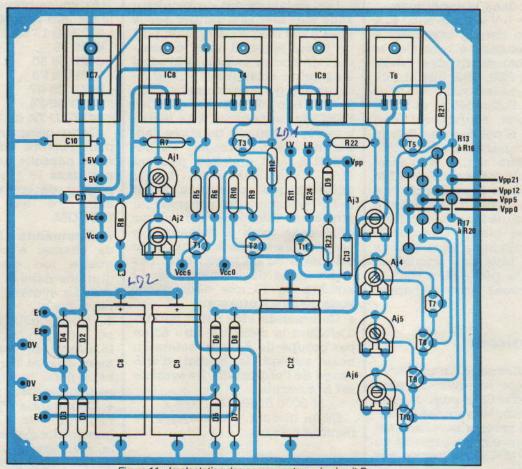


Figure 11 - Implantation des composants sur le circuit B.

nir une tension égale à 5 volts puis, en reliant Vcc 6 et Vcc 0 à la masse, réglez AJ2 pour obtenir 6 volts. En cas d'échec à ce niveau, vérifiez votre cablage car, étant nnée la simplicité du montage, seul ou un composant défectueux peut être à l'origine de vos déboires. Le test de l'alimentation VPP est aussi simple et lors de la mise en marche LD3 doit être éteinte et la tension en «E» de IC9 et en VPP, nulle. Si ce n'est pas le cas, vérifiez que la tension entre collecteur de T7 et la masse est nulle et échangez T7 dans le cas contraire. Ce premier test passé, reliez V_{PP 0} à la masse et réglez AJ3 pour obtenir 5 volts en VPP. Reliez VPP o et VPP 5 à la masse et agissez sur AJ4 pour que la tension en VPP atteigne 12,5 volts. De plus, LD3 doit être allumée alors qu'elle était éteinte lors du test précédent. Réglez AJ5 pour obtenir 21 volts en VPP en reliant VPP 0, VPP 5 et VPP 12 à la masse et AJ6 avec VPP0, VPP 5, VPP 12 et VPP 21 à la masse pour lire 25 volts. Cette série de tests ayant été passée avec succès montre le bon fonctionnement des trois alimentations. Les réglages des différentes tensions n'ont pas besoin d'être peufinés car nous les reprendrons lors des essais avec le moniteur.

Installer à présent IC₁, IC₂, IC₃ et IC₆ sur le circuit A et vérifiez que vous obtenez bien les mêmes

signaux sur les broches des deux PIA que sur les broches de la prise 25 points. Pour ce faire, reliez les broches PHI2, RST barre, R/W barre et A₀/A₃ à la masse ou au + 5 V par l'intermédiaire d'une résistance de 470 Ω. Pour le test de D₀/D₇, reliez R/W barre

à la masse dans le sens prise -> PIA et au + 5 V dans le sans PIA → prise. Un dernier test reste à faire qui consiste à contrôler le bon fonctionnement des relais. Reliez donc PB2, PB3, PB4 et PB5 de IC5 à la masse ou au + 5 V et vérifiez à l'ohmmètre que les commutations s'effectuent correctement. N'oubliez pas que le relais colle quand l'une des broches PB2/PB5 est à la masse. Dans le cas, par exemple, de RE4, la broche 26 du support doit être reliée à Vcc quand il est collé et à PB5 de IC4 dans le cas contraire.

Ce dernier test prouve que le PRM 4 fonctionne à 95 % et il ne vous reste plus qu'à étudier le logiciel qui le pilote pour les 5 % restants et vous servir de l'appareil

Le moniteur

e logiciel chargé de piloter le PRM 4 est entièrement écrit en assembleur 6502 pour des raisons évidentes de rapidité d'exécution et la version proposée est celle qui nous a servi à mettre au point le PRM 4. Elle tourne sur les ordinateurs COMMODORE 8032 et 8296 qui disposent d'un affichage de 25 lignes de 80 caractères et de 32 ou 128 K/ octets de RAM. Une version modifiée (affichage) tourne sur les COMMODORE 4032 (affichage 40 colonnes) et une autre encore sur le COMMODORE 64 qu'il est inutile de présenter. Nous reviendrons plus tard sur cette dernière version car elle nécessite une interface particulière de même que sur la version ORIC ATMOS qui feront l'objet d'un article particulier par la sui-

Nous avons donc écrit le moniteur en assembleur et les performances sont excellentes puisque nous obtenons, par exemple, le transfert en mémoire des 16384 octets d'une 27128 en 3,5 secondes ce qui donne 210 microsecondes par octet. Essayez donc de faire ça en BASIC! Notre remarque n'a cependant rien de péjoratif mais BASIC est fait pour certains types de programmes et l'assembleur pour d'autres.

Cependant, nous aurons l'occasion de décrire prochainement une version IBM PC en basic compilé aussi rapide.

Dès le lancement du moniteur vous êtes invités à vérifier que le PRM 4 est bien sous tension puis un menu apparait vous permettant de choisir le type d'EPROM à exploiter ainsi que la valeur de la tension de programmation VPP. Il est bon de savoir que les EPROMs ne peuvent être installées sur le support qu'après l'apparition du message '»COM-MANDE?' et qu'elles peuvent être manipulées sans danger entre chaque commande. Il est donc possible de programmer plusieurs mémoires du même type sans avoir à recharger le programme-source à partir de la disquette ou de la cassette. Les commandes disponibles sont au nombre de 13 que nous allons vous décrire à présent :

• Commande «M»: Elle permet d'afficher la mémoire sur l'écran par groupe de 8 ou 16 octets en hexa suivant le format d'affichage de l'ordinateur. La syntaxe est la suivante:

M 0000 0050 «RETURN»
Et on obtient, par exemple, le résultat suivant :

Voir tableau III.

• Commande «:»: Elle permet d'entrer directement des octets en mémoire de travail. Deux solutions sont possibles:

1) Votre ordinateur dispose d'un éditeur plein écran (cas de tous les COMMODORE) et il vous suffit d'amener le curseur sur l'octet à écrire ou modifier (ligne écrite par commande «M») et d'appuyer sur «RETURN» pour entrer la ligne modifiée.

2) Vous n'en disposez pas (quel dommage!) et il vous faut alors entrer la ligne complète avec le format obtenu par la commande "M".

• Commande «D»: Permet le desassemblage du programme en mémoire. Le desassemblage ne comprend que les codes du 6502 mais il serait tout à fait possible de le modifier pour qu'il s'adapte au 6800 ou au 6809. La syntaxe est la suivante:

D 0423 0430 «RETURN»

Et vous obtenez, par exemple, le résultat ci-après :

: 0423 20 13 0A	JSR \$0A13
: 0426 38	SEC
: 0427 E9 30	SBC # \$30
: 0429 F0 F8	BEQ <\$0423>
: 042B C9 07	CMP # \$07
: 042D B0 F4	BCS < \$0423>
: 042F 8D 7A 02	STA \$027A

Cette commande occupe pas mal d'octets en mémoire mais nous pensons qu'elle ne dépare pas dans le tableau puisqu'elle permet de contrôler très facilement ce qui est logé sur l'EPROM.

• Commande «L»: Elle permet de charger à partir de la disquette (ou cassette) un programme dans la mémoire de tr vail. La syntaxe en est fort simple:

L «NOM.PROGRAM» «RETURN» où «nom.program» est, bien sûr, le nom du programme préalablement chargé sur la disquette. Il est à noter que le nom du fichier ne peut exéder 16 caractères.

• Commande «S»: C'est le complément de la précédente puisqu'elle permet de transférer la mémoire de travail sur disquette (ou cassette). Dans l'exemple qui suit, on transfère sur le drive N° 1 le programme situé entre \$0000 et \$07FF sous le nom: «nom.prog». Comme pour la commande «L», le nom du fichier ne peut dépasser 16 caractères.

S «1: NPM.PROG» 0000 07 <RETURN>

• Commande «E»: Permet de vider la zone mémoire spécifiée en la remplissant avec la valeur \$FF ce qui correspond à l'état des EPROMs quand elles sont effacées et évite la programmation d'octets inutiles. Il est à noter que la mémoire de travail est automatiquement «vidée» de la sorte lors de l'initialisation du moniteur. Si nous voulons effacer la zone \$0000/\$0800, il suffit de faire:

E 0000 08000 < RETURN>

• Commande «V»: Dans l'exemple donné, on vérifie que l'EPROM est bel et bien effacée

	ADR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
3	0000	A0	98	45	F1	EA	79	EE	76	A9	OD	8D	85	03	78	A9	OC
	0010	A2	00	A0	00	9D	80	03	8D	04	80	CO	23	54	9A	25	02
:	0020		00														

entre \$0000 et \$025D. Ce qui revient à contrôler que tous les octets de cette zone sont bien égaux à \$FF. Toutes les adresses non effacées sont affichées sur l'écran.

V 0000 025D < RETURN>

• Commande «T»: Cette commande permet de transférer le contenu de l'EPROM entre les bornes indiquées dans la mémoire de travail. Dans l'exemple qui suit, nous transférerons la zone \$0000/\$0569:

T 0000 0569 < RETURN>

• Commande «P»: Elle effectue l'opération inverse de la précénte et programme l'EPROM en utilisant ou non l'algorithme rapide suivant le type de mémoire. Si nous voulons programmer la zone \$0000/\$1FFF, nous utiliserons la commande suivante:

P 0000 1FFF < RETURN>

L'opération se déroule en trois phases qui apparaissent à l'écran :

1) Vérification de la virginité de la zone à programmer et arrêt du programme en cas d'erreur.

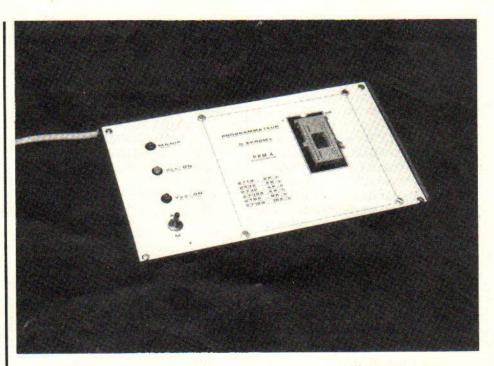
2) Programmation sur l'EPROM et test immédiat de conformité pour les 2764 et 27128.

3) Vérification de la conformité entre le programme-source et l'EPROM.

• Commande «C»: Elle permet c' vérifier la correspondance corre-te entre la mémoire de travail et l'EPROM. Toutes les adresses où une erreur est détectée sont affichés. Pour comparer les adresses \$0000/\$03C2 nous ferons:

C 0000 03C2 < RETURN>

- Commande «R»: L'appui sur «R» suivi de «RETURN» entraine la réinitialisation du programme permettant de choisir un autre type d'EPROM. La mémoire de travail est totalement mise à \$FF après cette commande. Un nouvel appui sur «RETURN» est demandé pour confirmer cette commande, comme la suivante d'ailleurs, afin d'éviter toute erreur.
- Commande «B: L'appui sur «B» suivi de «RETURN» permet de sortir du moniteur et de retomber sous le conrôle de basic dans le cas du CBM. Un ordre «RUN» relance le moniteur.



• Commande «H»: Un appui sur «H» suivi de «RETURN» permet d'obtenir l'affichage d'un aidemémoire à l'écran qui résume le rôle des commandes.

Cette longue description des commandes disponibles vous aura, nous l'espérons, convaincus des intéressantes possibilités du PRM 4 et de son moniteur. Avant de conclure, il nous faut lever le voile sur le fonctionnement exact de certaines routines-système du C.B.M 8296 ou 8032 et des particularités de cet ordinateur afin que vous puissiez l'adapter sans problème au votre.

Si vous étudiez le listing-source du moniteur, vous vous apercevrez que les PIA sont implantés en \$9000/\$9007, que la mémoire de travail débute en \$4000 et que le moniteur lui-même commence en \$0401.

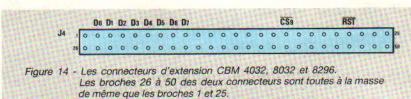
La zone \$9000/\$9FFF est libre sur les C.B.M 4032, 8032 et 8296 si ce n'est qu'elle peut recevoir une PROM complémentaire (cas de VISICALC, par exemple). Si une telle PROM réside dans votre C.B.M, il vous faudra l'oter pour

utiliser le PRM 4 ou, mieux encore, la neutraliser en reliant par le biais d'un inverseur la broche 20 de cette mémoire soit à CS9 barre (voir figure 14) soit au + 5 V en série avec une résistance de 1 KΩ. De cette manière, vous pourrez continuer à utiliser vos logiciels sans avoir à manipuler constamment la PROM en question.

La mémoire de travail doit contenir au moins 16 K/octets puisque nous voulons gérer les 27128. Nous avons donc placé cette zone en sommet de RAM soit entre \$4000 et \$7FFF.

Sur les C.B.M, le BASIC débute en \$0401 et nous avons recréé une ligne de BASIC (c'est la seule!) qui fait office de «boot» de lancement du moniteur ce qui permet de lancer ce dernier par un simple «RUN».

Pour assurer une portabilité maximum, nous n'avons employé qu'un minimum de routines-système et en avons même ré-écrites un bon nombre qui font partie de la bibliothèque du système (ADOUT, HEXASC, CODEHEX,



etc). Le rôle et l'effet des 8 routines employées sont les suivantes :

READY: Un «JMP» vers cette routine redonne le contrôle à BASIC. On obtient exactement le même effet qu'un «END» en BASIC.

INPUT: Un «JSR» vers cette routine simule le «INPUT» du BASIC. Le curseur clignote, l'entrée de la chaine de caractères s'opère en mode plein écran et ladite chaine est transférée en \$0200 et suite après «RETURN» et se termine par un octet \$00.

ECRINT: Cette routine permet d'afficher sur l'écran le nombre entier dont la partie haute est dans l'accumulateur et la partie basse dans X. La séquence d'appel est donc:

LDA # MSB LDX # LSB JSR ECRINT

ECRIT: Permet l'affichage du caractère ASCII dont le code est dans l'accumulateur. Les mouvements de curseur sont gérés et les registres X et Y sont préservés.

TESTOP: Cette routine teste l'appui sur la touche «STOP». En cas d'appui on retrouve \$00 dans l'accumulateur et la valeur précédente de l'accumulateur dans le cas contraire.

LOAD: Permet de charger un programme en mémoire à partir de la disquette ou de la cassette. Les adresses suivantes doivent être garnies avant le «JSR LOAD»:

STATUS (S96) doit être mis à \$00. NOPER (\$D4) avec le N° de périphérique (8 : disquette, 1 : cassette).

NBCAR (\$D1) avec la longueur du nom du fichier.

ADNOM (\$DA/\$DB) doit pointer vers le nom du fichier (MSB en \$DB, LSB en \$DA).

SAVE: Cette routine effectue la sauvegarde d'une partie de la mémoire sur disquette ou cassette et la séquence d'appel est la même que pour «LOAD». L'adresse de début du programme doit être en \$FB/\$FC et l'adresse de fin en \$C9/\$CA.

GET: Porte bien son nom et correspond au «GET» du BASIC. Si aucune touche n'est enfoncée, on retrouve \$00 dans l'accumulateur et le code ASCII de la touche enfoncée dans le cas contraire.

Comme vous pouvez le constater, tout celà n'a rien de bien compliqué et vous ne devriez pas avoir trop de problèmes pour adapter le moniteur à un ORIC ou un APPLE 2.

Nous pouvons vous fournir les logiciels suivants qui tournent sur COMMODORE 4032, 8032, 8296 et COMMODORE 64:

— Moniteur du PRM 4 (GES-PROM).

Editeur/assembleur.Moniteur langage-machine.

Lesquels constituent le «package» nécessaire à une exploitation correcte du PRM 4. Si vous êtes intéressés; envoyez une demande à notre attention à la revue qui transmettra.

Ceci est une solution provisoire. Nous étudions une solution que nous vous proposerons le mois prochain afin de faire distribuer le logiciel. De toute façon, il est hors de question de le publier dans la revue eu égard à l'espace occupé.

Il ne nous reste plus qu'à tester le programmateur sous le contrôle du moniteur et, pour ce faire, relier le PRM 4 à l'ordinateur.

Procurez-vous deux connecteurs 2 fois 25 points au pas de 2,54 mm qui seront reliés aux ports J4 et J9 du C.B.M. En vous aidant de la figure 14, reliez le PRM 4 à l'ordinateur par deux cables en nappe de 12 conducteurs (longueur maxi: 50 cm) sans oublier les 8 fils de masse (4 sur J₄ et 4 sur J₉). Sur le 8296, il faut établir le strap JU7 qui se trouve entre le 6502 et les ROMs afin d'autoriser la lecture des registres des deux PIA. Sur le 4032 et le 8032 le bus de données du PRM 4 doit être relié, non pas au port J4 comme pour le 8296, mais au support de PROMs sit en \$9000 grâce au petit montage de la figure 15 qui permet, en plus, de continuer à vous servir de l'éventuelle ROM implantée à cette adresse.

Mettez le PRM 4 sous tension puis le C.B.M et constatez que ce dernier fonctionne et que la LED verte du PRM 4 est bien allumée.

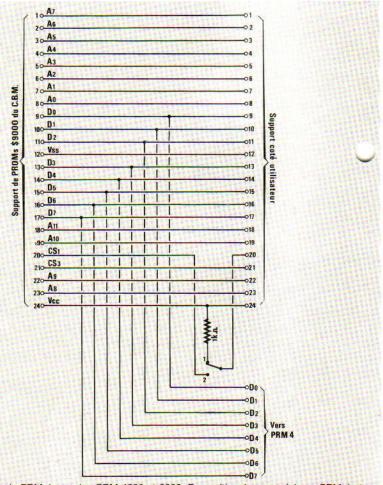


Figure 15 - Liaison du PRM 4 pour les CBM 4032 et 8032. En position 1, on accède au PRM 4 et en position 2, à la ROM habituellement branchée sur le support. Sur le support d'origine, on peut utiliser des EPROM 2532 (voir notice de l'appareil).

Un programmateur d'EPROM universel le PRM 4



Chargez et lancez le moniteur | par «RUN» et choisissez une 27128 avec une tension VPP de 12.5 volts. Lancez la commande «C 0000 3FFF»: la LED jaune doit s'allumer pendant quelques secondes prouvant le bon fonctionnement du PRM 4. Contrôlez les tensions Vcc et VPP qui doivent être de 5 volts pendant l'exécution de la commande et reprenez éventuellement les réglages de AJ₁ et de AJ₃. Lancez une commande «P 0000 1000», la diode jaune doit s'allumer puis la diode rouge et, pendant la programmation, Vcc doit atteindre 6 volts et VPP, 12,5 volts. Retouchez AJ₂ et AJ4 le cas échéant. Par la commande «R», choisissez une 27128 avec une tension VPP de 21 volts et recommencez le test précédent en agissant sur AJs si le besoin s'en fait sentir. Choisissez

maintenant une 2716 et recommencez le test par une commande «P 0000 0200» et intervenez sur AJ6 pour régler la tension VPP à 25 volts.

Ce dernier test démontre le bon fonctionnement du PRM 4 et il ne vous reste plus qu'à programmer quelques EPROMs pour vous faire la main.

Nous cloturons ici ce long article (il faut dire que le sujet est vaste!) et vous donnons rendezvous dans un prochain numéro de «Radio Plans» pour l'interfacage du COMMODORE 64, de l'ORIC ATMOS et, par la suite, de l'APPLE 2 et de l'IBM PC. Comme de coutume, nous restons à la disposition des lecteurs dans l'embarras et vous souhaitons plein succès dans la réalisation du PRM 4.

Ph. WALLAERT

1, allée des Berges 94370 Sucy-en-Brie Tél.: 45.90.56.11

REALTECHNIC

POUR TOUS PROBLEMES DE COMMANDE A DISTANCE

ECOMMANDE RADIO DE GRANDE TELECOMMANDE RADIO DE GHANDE FIABILITE, monostable ou bistable, toutes applications professionnelles ou privées, par déclenchement de flash, d'appareil ou caméra ouverture/fermeture d'a. s divers, garage, propriété, transmis-sion d'alarme, activation et neutralisation de systèmes d'alarmes....

POUR APPLICATIONS PHOTO

POUR APPLICATIONS PHOTO

ENSEMBLE TR 1/1: Dimensions emetteur/récepteur 131 x 60 x 29, boitier
plastique noir. Mini-antenne téléscopique.
Codé à la fabrication, 19 683 codes différents. Sur l'émetteur: poussoir de test.
prise pour raccordement au matériel de
commande (prise synchro, barrière photo-électrique, et tous contacts travail). Sur le
récepteur, relais 2A et prise pour raccordement du matériel à commander (flash. moteur
de boitler...). Matériel compact et léger, idéale pour la chasse ou le piège photographique. Portée environ 150 m. Prix sans piles : 990 F l'ensemble.

POUR APPLICATION PROFESSIONNELLE OU PRIVEES

Une gamme d'émetteurs récepteurs de 1 à 15 canaux (30 sur demande), 50 mW à 4 W. Temps, de réponse : 50 ms. Prix de 805 à 2.775 F avec antenne, sans piles.

Exemple: voir photo.

Exemple: voir photo.

Emetteur récepteur 8 canaux. 4 W, relais 8A, livré avec antenne. Dimensions de l'émetteur : 190 x 75 x 40 métal noir et allu, clavier de commande type téléphone digital. Consommation nulle en veille.

Récepteur, dimensions : 190 x 138 x 68, boitier plastique gris clair, trèsd esthétique. 243 codes PCM différents. Prix de l'ensemble 2 395 F, sans piles.

TRANSMETTEUR TELEPHONIQUE, complément indispensable à votre système d'alarme 1 numéro, alimentation 12 V par la centrale, ou extérieure. Raccordement par bonnier à vis. Signale l'alarme par émission d'un signal sonore caractéristique puis raccroche. Renouvelle l'appel toutes les dix minutes, jusqu' au décroché du numéro qu'il a composé. Acquit par rappel du lieu où se trouve le transmetteur, laissez sonner dix fois, puis raccrocher. Ne mobilise pas la ligne. Programmation facile du numéro à avertir. Prix sans alimentation : 1 200 F.
Pour tous ces matériels, PORT 35 F.

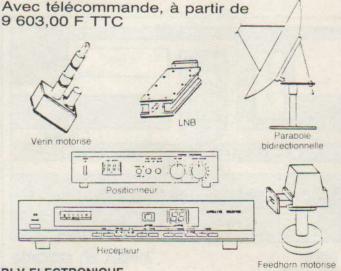
VENTE PAR CORRESPONDANCE DOCUMENTATION ET TARIFS SUR DEMANDE EN JOIGNANT UNE ENVELOPPE TIMBREE A

REPLIECHNIC 1, allée des Berges 94370 Sucy-en-Brie

Pour tout renseignement téléphoner de 9 h à 19 h au 16 (1) 45.90.56.11

du lundi au samedi

ENFIN DES SYSTEMES DE RECEPTION DE CHAINES PAR SATELLITES A DES PRIX ABORDABLES



BLV ELECTRONIQUE

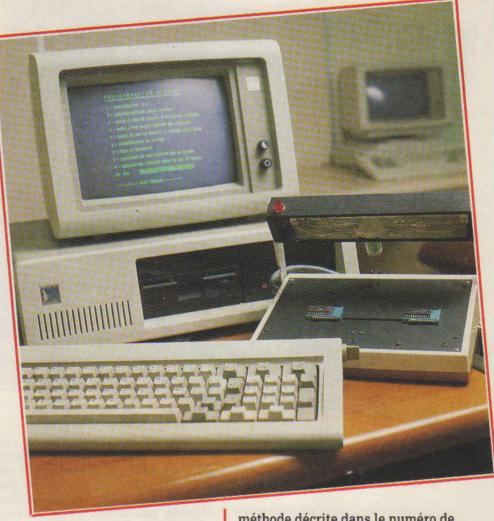
15. av. du Gl-Malleret-Joinville 94110 ARCUEIL

Tél.: (1) 46.63.60.88. Tlx: 631.615 F

DEMANDE DE DOCUMENTATION joindre 15 F en timbres

Nom	 	 			 0		All								*				 			. ,	
Adresse	 	 	1900		 						 								 	7070			

ous avons choisi le PRM 4 pour sa simplicité, son faible coût et surtout à cause d'une certaine universalité que nous ne retrouvons malheureusement pas toujours avec les cartes compatibles PC proposées dans le commerce et ce, à des prix nettement plus élevés. De plus ce montage a déjà été réalisé à un certain nombre d'exemplaires et il est parfaitement éprouvé. Par contre, il existe une certaine ncompatibilité apparente entre cet appareil et l'IBM-PC (1) à cause des PIA choisis: les 6520 ou 6821. En effet, ces circuits fonctionnent à 2 MHz maximum dans leurs versions les plus rapides. Quant aux versions que l'on trouve couramment chez les revendeurs (classe A), elles ne dépassent pas le megahertz, ce qui est nettement insuffisant pour le PC dont la vitesse d'horloge est de 4,77 MHz. Evidemment, il eût été intellectuellement plus satisfaisant d'utiliser le circuit que l'on rencontre dans les PC: le 8255 "Intel, dont la programmation est de surcroît plus simple que celle des 6821 ou 6520. Nous n'avons cependant pas suivi cette démarche pour plusieurs raisons: tout d'abord la vitesse d'horloge a peu d'importance lorsqu'on la compare aux millisecondes, voire aux centièmes de secondes nécessaires à la programmation d'un octet en Eprom. D'autre part, il nous a paru intéressant pour beaucoup de nos lecteurs, d'étudier un interface permettant de connecter à un PC, des périphériques conçus pour des ordinateurs plus lents et ceci, sans passer par une liaison asynchrone plus complexe à mettre en œuvre. C'est dans cet esprit que nous



avons voulu une carte à la fois la plus universelle et la plus simple possible, ce qui explique la présence des cavaliers de configuration et d'une zone laissée intentionnellement vierge, utilisable en wrapping selon la

méthode décrite dans le numéro de décembre (procédé Circuigraph).

Mais avant de passer à l'étude détaillée de cette carte, quelques considérations sont à prendre en compte sur la mise en œuvre du programmateur.

Le PRM 4 et l'IBM-PC

(1) IBM-PC est une marque déposée de International Business Machines Corp.



Le langage de programmation

e deuxième objectif que nous nous sommes fixé est de vous proposer des logiciels que vous pourrez éventuellement aisément modifier et dont vous deviendrez finalement les maîtres d'œuvre. Nous avons donc éliminé d'emblée l'assembleur bien qu'il reste le plus efficace en ce qui concerne les temps d'éxécution — parce qu'il ne répond pas tout à fait à l'idée que nous nous faisons de la simplicité et de la commodité de modification.

Le langage C aurait été certainement le meilleur compromis, mais il reste encore méconnu. Restaient finalement le Pascal et le Basic, nous avons dû faire un dernier choix difficile et opter pour le langage le plus répandu : notre bon vieux cher Basic!...

Nous entendons déjà les rumeurs et protestations qui s'élèvent en ce qui concerne ses piètres performances! A cela nous ne répondrons qu'une chose: si vous pensez que 1 minute 47 secondes est un temps inacceptable pour programmer une 2764, c'est que vous en avez cer-

tainement des centaines à programmer chaque jour, et effectivement notre réalisation n'a pas la prétention de répondre aux contraintes d'un usage professionnel, par contre nous pensons que pouvoir programmer 30 Eprom 2764 à l'heure reste encore une cadence raisonnable pour les amateurs que nous sommes.

Basic donc, mais pas n'importe comment, en effet les programmes que nous vous proposons sont conçus pour être COMPILES et ne peuvent ABSOLUMENT PAS FONCTIONNER CORREC-TEMENT avec des interprèteurs tels que Basica ou Gwbasic. INSISTONS, Si essayez de programmer les Eprom avec notre logiciel en utilisant un interprèteur : VOUS LES DETRUIREZ PUREMENT ET SIM-PLEMENT, car les temps de programmation ne seront absolument plus respectés.

Pour ceux d'entre vous qui ne connaissent pas exactement la différence entre un interprèteur et un compilateur, une explication s'impose : compiler un programme, y compris en assembleur, consiste à en traduire définitivement les instructions en langage machine avant de l'éxé-

cuter, alors que les mêmes instructions utilisées avec un interprèteur seront traduites pendant le déroulement du programme à chaque fois qu'elles sont éxécutées, même si l'on y passe plusieurs fois comme dans le cas d'une boucle par exemple. Il est bien évident que dans ces conditions, les performances de l'interpréteur sont en général moins bonnes (1 à 10 en moyenne). Par contre la mise au point avec un interprèteur est plus facile parce que l'on peut mieux suivre pas à pas le déroulement du programme et en modifier les instructions sans avoir à repasser par les phases de compilation et de linkedit.

Un autre avantage du basi compilé est de permettre, sans précautions particulières, d'utiliser le concept de la programmation structurée. Ce concept consiste à découper le programme en bloc d'instructions constituant chacun une fonction élémentaire et dont le fonctionnement interne reste isolé de celui des autres fonctions. En d'autres termes, on structure le programme en fonctions logiques, en s'assurant que chacune de ces fonctions ne possède qu'un seul point d'entrée et de sortie, c'est-à-dire que les « goto » se réfèrent toujours à une instruction appartenant à ce même bloc logique sinon on utilisera l'instruction « gosub », c'està-dire l'appel d'une autre fon tion. Cette technique permet avant tout de faciliter la mise au point et la modification du programme, chaque fonction pouvant être isolée et testée de façon indépendante, un peu comme les composants d'un montage électronique.

Les logiciels

Nous vous proposons les sources en basic structuré de deux logiciels. Le premier est un logiciel de mise au point assistée par ordinateur qui vous permet d'une part, de contrôler le bon fonctionnement du programmateur et qui d'autre part vous propose un diagnostic des pannes éventuelles et les réglages à effectuer. Il vous offre même une petite récompense musicale en cas de succès. Le second

Suite page 96.

Suite de la page 72.

concerne l'utilisation du PRM 4 lui-même et comporte les fonctions suivantes :

- définition de la configuration PC
- caractéristiques des Eprom (ou ROM, RAM, etc...)
- copie d'un fichier disquette en RAM de travail
- copie de la RAM de travail (ou autre partie de la RAM adressable) sur disquette
- copie du contenu d'une Eprom en RAM de travail
- programmation d'une Eprom (accélérée ou non)
- test de virginité
- affichage de la RAM de travail et modification
- comparaison du contenu d'une Eprom et de la RAM de travail.

Pour ces deux programmes, nous avons « aéré » et simplifié au maximum l'écriture, nous n'avons utilisé d'astuces de programmation qu'en cas d'absolue nécessité et ajouté des commentaires pour presque toutes les instructions. Nous espérons que toutes ces précautions vous encouragerons, même si vous êtes programmeur débutant, à plonger dans les programmes et à les modifier à votre convenance.

Nous insistons sur la possibilité, voire la nécessité de modifier les programmes pour plusieurs raisons:

- les adresses de ports utilisées sont celles de la carte prototype IBM c'est-à-dire 300 à 31 F, ces adresses peuvent être modifiées et reportées jusqu'en 33 F, cette modification s'opère par des cavaliers sur la carte d'interface, mais faut-il encore adapter le programme (attention toutefois au contrôleur du disque dur en 320-32 F).
- les logiciels qui vous sont fournis ont été développés pour satisfaire le plus grand nombre, ils ne gèrent donc qu'un seul module PRM 4 et sont conçus pour fonctionner aussi bien sur une carte d'affichage monochrome que sur une carte graphique-couleur, il n'y a donc pas d'utilisation de la couleur. Certains d'entre vous voudront peutêtre modifier l'un ou l'autre de ces points.
- Nous avons prévu l'utilisation d'un certain nombre d'Eprom depuis la 2716 jusqu'à la 27256,

mais les possibilités du PRM 4 dépassent largement cet éventail, c'est ainsi que nous avons rajouté dans la panoplie, pour nos besoins propres, la ROM 4764 ainsi que la RAM 6264 pour effectuer les tests des programmes. Cette liste n'est absolument pas limitative et on pourrait, par exemple, envisager la programmation de la 27512 en modifiant légèrement le PRM 4 (en utilisant le contrôle périphérique CA 2 de IC4 pour l'adresse A 15, pin 1 de la 27512) ou même d'Eprom monotension.

- Nous nous sommes aperçus en faisant de nombreux tests sur des Eprom d'origines diverses et parfois même douteuses, que pour une référence donnée, il pouvait arriver dans certains cas que les caractéristiques soient différentes. Ainsi un lot de 2764 acheté à un prix défiant toute concurrence a refusé obstinément de se laisser lire en présence de la tension de programmation, ce qui exclut la programmation accélérée. De même, pour une même référence nous avons rencontré des tensions de programmation différentes suivant les constructeurs : 21 V ou 12,5 V pour des 27256. Dans le même ordre d'idée, toutes les Eprom 2764 à 27512 ne supportent pas ipso facto 6 V en tension d'alimentation. Ces disparités peuvent donc nécessiter d'adapter les algorithmes en fonction de vos besoins.
- En ce qui concerne l'effacement, comme vous avez pu l'apercevoir sur la photo, notre réalisation est pourvue d'un tube U.V., la première chose qui vient à l'esprit c'est naturellement d'asservir cet effacement par un test de virginité, ce qui est d'une simplicité biblique avec le PRM 4. Eh bien, dites vous que ce serait trop simple; en effet, le fait de mettre sous tension et de lire une Eprom pendant son exposition aux U.V. amène parfois des phénomènes inattendus et contraires au but recherché: on reprogramme certains octets! nous vous proposons cependant une méthode (nous devrions dire une recette) dans le paragraphe concernant l'effacement, mais nous n'avons pas jugé utile de la programmer, certains d'entre vous souhaiteront peut-être le faire.

- la programmation d'Eprom nécessite une zone de mémoire tampon dans laquelle nous serons éventuellement amenés à travailler avant de charger l'Eprom. Or, il se trouve que sur un PC, nous ne sommes pas entièrement maîtres de l'utilisation de la mémoire qui est en réalité gérée par le DOS. Il nous reste alors trois solutions :
- une table en Basic, cette solution a le mérite de la simplicité, par contre le domaine de basic est limité à un segment, soit 64 k et un poste de table utilise en Basic 2 octets, nous ne pourrions donc pas envisager la programmation des 27256 puisque le logiciel occupe une partie i segment. De plus, les temps d'accès seraient sensiblement plus longs qu'avec des « peek » et des « poke ».
- de la mémoire additionnelle placée dans l'espace adressable réservé aux extensions, par exemple dans les segments D0000 et E0000, si la configuration le permet. Cette deuxième solution serait la plus élégante car nous serions certains que le DOS ne viendrait pas nous prendre de la mémoire à cet endroit. En outre, il existe des cartes multifonctions nous offrant cette possibilité.

Malheureusement cette solution n'est pas aussi simple qu'on pourrait le croire et ne représente pas la panacée.

En effet, en plus du prix non négligeable de la carte multifonctions, cette portion de mémoire n'est pas initialisée par la routine BIOS au démarrage du système, et il peut arriver dans certains cas que le système y découvre malgré tout des erreurs de parité avant même de solliciter l'autoexec.bat.

Dans ce cas, la seule solution envisageable est d'inhiber le contrôle de parité pour cette partie de mémoire (c'est possible sur certaines cartes multifonctions).

— Il existe une troisième solution utilisable sur des machines généreusement pourvues en mémoire, qui consiste à adresser une portion de mémoire que le DOS a peu de chance d'utiliser. En fait le DOS utilise la mémoire « par les deux bouts », en début de mémoire nous retrouvons la partie résidente du DOS ainsi que nos divers programmes résidents

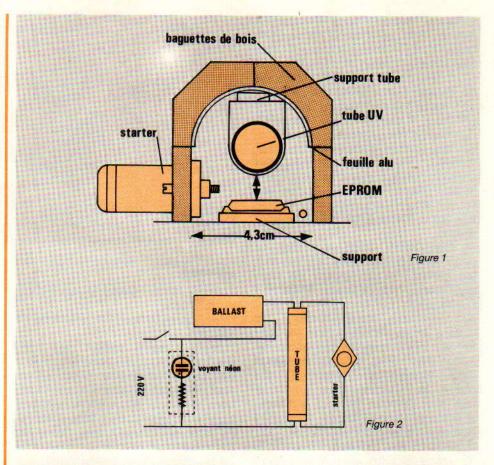
comme keybfr. En fin de mémoire, DOS dispose d'une portion de mémoire réutilisable où sera chargé, entre autre, le « command; com », à vous de situer votre mémoire de travail entre les deux. Pour fixer les choses, sur une machine disposant de 640 K, vous ne serez pas importunés, même par DEBUG entre les adresses hexadécimales 40000 et 80000.

Cependant vérifiez-le, car tout dépend des programmes résidents que vous utilisez.

La solution retenue dans notre logiciel est la troisième, le début de la zone de travail étant paramétrable. Par contre, dans le cas machine un peu juste en taille mémoire, il faudrait envisager l'une des deux premières solutions. La première entraîne une légère modification des programmes alors que la seconde peut être mise en œuvre sans modifications des programmes basic mais avec les restrictions évoquées ci-dessus. Là encore, ce sera à chacun de choisir en fonction de la configuration dont il dispose et d'adapter les programmes à ses besoins.

L'effacement

Sur notre maquette le dispositif d'effacement est une boîte rabattable constituée par des baquettes de bois collées côte à côte et d t le fond semi-cylindrique recouvert de papier d'aluminium sert de réflecteur (figures 1 et 2). Nous y avons placé un tube U.V. de 6 watts et d'un longueur horstout de 22 cm (15 cm utiles). On trouve facilement dans le commerce ce type de tube dont la principale vertu est d'être bactéricide mais qui efface également nos Eprom. Les fabricants préconisent en réalité un rayonnement d'une longueur d'onde de 2537 angströms et une insolation totale de 15 W sec/cm2, ce qui équivaut à une intensité de 12,5 mW/cm² pendant 20 minutes, mais rassurez-vous il y a une certaine marge. Ce sera à chacun d'étudier la meilleure solution en fonction du matériel dont il dispose et de ses talents de bricoleur, en gardant toutefois présent à l'esprit que les ultraviolets sont DANGEREUX pour les yeux. Par conséquent, il est nécessaire que le dispositif en position d'efface-



ment soit étanche aux U.V. et il ne serait pas superflu d'ajouter un micro-contact pour empêcher tout allumage accidentel en position relevée, surtout si vous avez de jeunes enfants (chers bambins!). Sur notre maquette, nous nous sommes contentés d'un témoin de mise sous tension mais cela nous paraît d'une sécurité suffisante. Un autre point important réside dans la distance entre le tube et les Eprom ; là encore. les constructeurs ont exprimé leur avis en stipulant une distance de 2,5 cm, cependant nous pensons qu'elle s'applique à des tubes professionnels de plus forte puissance que celui qui est utilisé dans notre montage ; nous avons donc réduit cette distance à 1 cm, et ce, malgré l'avis pessimiste de certains, qui nous avaient prédit qu'après un tel traitement les Eproms pourraient être testées à la fourchette (quand la fourchette s'enfonce facilement dans l'Eprom, c'est qu'elle est vierge!). Nous avons donc effectué une série de tests qui consistaient à exposer aux U.V. une 2764 pendant une heure et à la reprogrammer ensuite, nous avons pu exécuter plusieurs fois de suite cette opération sans

problèmes apparents. En fait, la méthode que nous avons mise au point de façon empirique, consiste à exposer l'Eprom hors tension pendant 2 minutes, puis à effectuer un test de virginité : si il est positif nous la laissons 5 minutes de plus, sinon nous l'exposons pendant un quart d'heure supplémentaire. Pratiquement toutes les Eproms que nous avons soumises à ce traitement ont retrouvé leur virginité d'antan, à l'exception toutefois d'une 2716 de récupération dont quelques dizaines d'octets sont restéés réfractaires. De toutes façons, sachez que le nombre de cycles d'écriture/effacement n'est pas infini et qu'une overdose d'U.V. n'arrange rien.

La carte d'interface

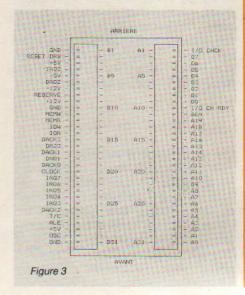
ette carte d'interface assure les trois fonctions suivantes :

— ISOLER le périphérique du bus pour des raisons de sécurité et d'intégrité des signaux. Un court-circuit accidentel surtout au niveau du câble de liaison et des connexions est vite arrivé, ce serait tout de même dommage de faire sauter les circuits de con-

trôle du bus du PC, il vaut mieux que ce soit ceux de la carte d'interface. D'autre part, pour des raisons de confort et de commodité, il est préférable que le câble de liaison entre le PRM 4 et le PC soit un câble rond et qu'il mesure environ 1 mètre, ceci entraîne des capacités parasites importantes qui provoqueraient, non seulement, une déformation des signaux présents sur le bus, mais aussi, une charge trop importante. Le rôle des circuits IC5, IC8 et IC9 est d'éviter tous ces désagréments.

 DECODER l'adresse choisie pour le périphérique. Sur un PC, les positions des ports ne font pas partie de l'espace mémoire adressable comme c'est le cas sur certains ordinateurs; un signal nous permet donc de discerner les adresses mémoire de celles des ports, il s'agit du signal AEN broche A11 (figure 3). En outre, la sélection des ports se fait sur 16 bits, ce qui autorise théoriquement l'accès à 65536 positions d'entrée/sortie ; en fait, la plupart des cartes ne décodent en général que les 10 premiers bits soit 1024 positions, ce qui reste encore largement suffisant. Ainsi, les adresses des ports qui pourraient nous intéresser sont les suivantes :

- les manettes de jeux en 200-20F
- l'interface série secondaire en 2F8-2FF
- notre propre carte en 300-30F (modifiable)
- le disque fixe en 320-32F
 une des imprimantes en 378-37F



Adresse	Positions de	es cavaliers	Remarques
	J1	J2	
300 - 30 F		0 0 0	Adresses de la carte prototype IBM (1)
310 - 31 F	000	000	Adresse de la carte prototype IBM (1)
320 - 32 F	000	000	Adresses utilisées par le disque dur
330 - 33 F	000		Figure 4

- l'écran monochrome en 3B0-3BF
- l'écran graphique/couleur en 3D0-3DF
- l'interface série primaire en 3F8-377

Notre carte utilise les bits d'adresse A₄ à A₉ pour sélectionner le périphérique, cette sélection est configurable par les cavaliers J₁ et J₂ suivant le tableau figure 4; quant aux bits A₀ à A₃, ils sont transférés tels quels au périphérique, ce qui nous procure 16 positions, soit éventuellement deux modules PRM 4 grâce à une petite modification apportée au deuxième module.

— RALENTIR le PC lorsque le périphérique est sélectionné, et seulement dans ce cas-là. Cette fonction est assurée grâce à une possibilité très intéressante du PC qui permet de bloquer la présence des signaux sur le bus pendant un temps pouvant aller jusqu'à 2 microsecondes.

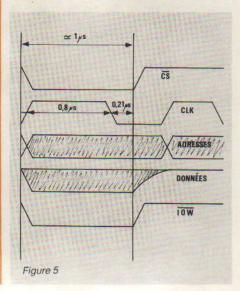
Attention toutefois, c'est un maximum, au-delà vous risquez quelques ennuis, en particulier avec le rafraîchissement de la RAM dynamique.

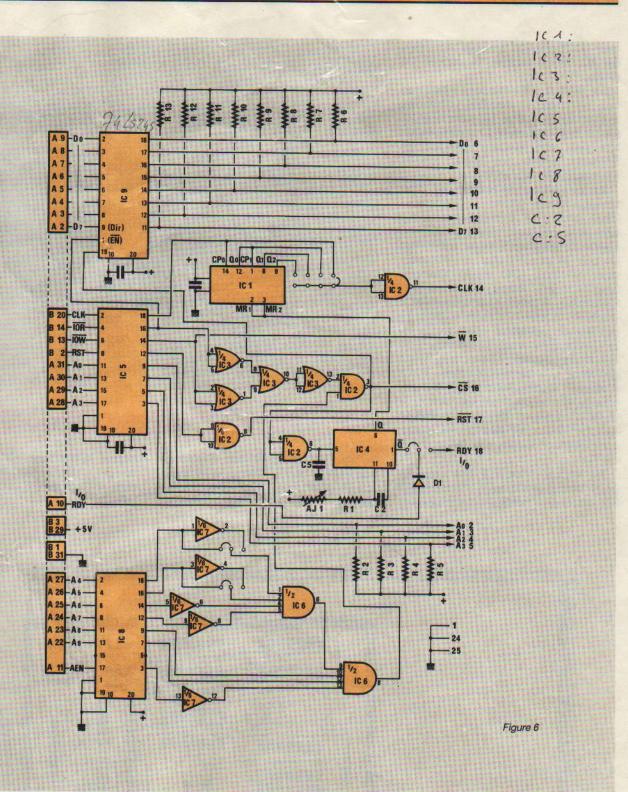
Ce delai est encore bien suffisant pour un périphérique fonctionnant à 1 MHz, toutefois les 6520 ou 6821 ne se contentent pas de celà, il leur faut plusieurs cycles d'horloge consécutifs pour effectuer leurs transferts internes. Nous leur fournissons donc un signal d'horloge à 1,2 MHz que nous resynchronisons à chaque appel du périphérique (voir diagramme des temps figure 5). Nous n'avons pas voulu limiter la carte d'interface à la seule utilisatin du PRM 4, par conséquent, nous l'avons dotée de deux cavaliers de configuration supplémentaires. Le premier permet de choisir la fréquence d'horloge qui peut aller de 600 kHz à 4,77 MHz. Le second permet de sélection l'origine du signal provoquant le retard : soit le périphérique quand il est capable de dire qu'il est prêt (dans ce cas, ce signal est récupéré sur la prise DB25), soit le monostable présent sur la carte elle-même dont on règlera la largeur d'impulsion par AJ1 en fonction des besoins. Dans le cas du PRM 4 on choisira cette deuxième solution. Ensuite le signal sera dirigé vers la position A₁ du bus (I/O Channel Ready).

Principe de la carte d'interface

Les bits d'adresse A₄ à A₉ ainsi que le signal AEN sont prélevés sur le bus par l'intermédiaire 12 tampon IC₈.

Si nous choisissons par exemple, comme adresse pour notre carte: 300 hexa, la configuration de bits à sélectionner sera alors:





 $A_9 = 1$; $A_8 = 1$; $A_7 = 0$; $A_6 = 0$; $A_5 = 0$; $A_4 = 0$; AEN = 0.

Nous serons donc amenés à inverser A₇, A₆, A₅, A₄ et AEN avant de les soumettre à la double porte AND/4 entrées, constituée par IC₆. A l'issue de IC₆, nous récupérerons un 1 logique lorsque toutes les conditions seront satisfaites. A noter que l'on peut choisir d'inverser ou non A₄ et A₅ grâce aux cavaliers

J₁ et J₂, ce qui permet de changer d'adresse.

On récupère également par le tampon IC5, les signaux de lecture et d'écriture IOR et IOW. Ces signaux sont combinés en logique OU et retardés grâce à IC3, ils servent également à commander les directions des tampons 3 états: IOR celle de IC9 et IOW celle du circuit IC2 du programmateur. Nous avons choisi

ce principe pour que les tampons soient toujours parfaitement synchronisés malgré la présence du câble de liaison. Par contre, il est indispensable de n'appliquer le signal CS que lorsque les directions sont bien établies sur les deux tampons. Pour éviter des conflits à la transition. C'est la raison du retard introduit par les portes NOR de IC3 et de la combinaison de ce signal avec celui

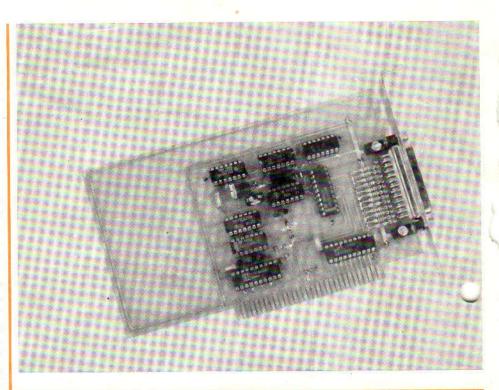
issu de IC₆ grâce à la porte NAND de IC₂ pour former le signal CS définitif.

Le signal CS est alors appliqué sur les commandes ENABLE des tampons 3 états et sert également à déclencher le monostable IC4 par son entrée trigger de Schmitt après un léger retard

introduit par IC3 et C5.

La raison de ce retard est l'obtention d'un front montant sur CLK alors que le programmateur est déjà sélectionné par CS. La sortie O sera appliquée à travers D₁ sur la broche A₁₀ du connecteur de bus (I/O READY) pour freiner le PC. Avec certaines cartes mère, il peut être nécessaire de mettre une résistance de rappel R14 en parallèle avec D1, pour le savoir, mesurez l'impédance de A_{10} et ramenez celle-ci à 4,7 k Ω grâce à R₁₄ si nécessaire. Pour d'autres applications, on pourra également récupérer un signal I/ O RDY externe disponible sur la broche 18 de la prise DB 25, cette option est donnée par la position de J4.

La sortie Q de IC4 est utilisée pour la remise à zéro du diviseur IC1 qui nous permet de disposer



de quatre fréquences d'horloge possibles 4,77 MHz, 2,38 MHz, 1,2 MHz et 0,6 MHz sélectionnables par J₃. Pour terminer, ce signal d'horloge est inversé par IC₂ pour disposer d'un front montant à l'appel du programmateur par CS.

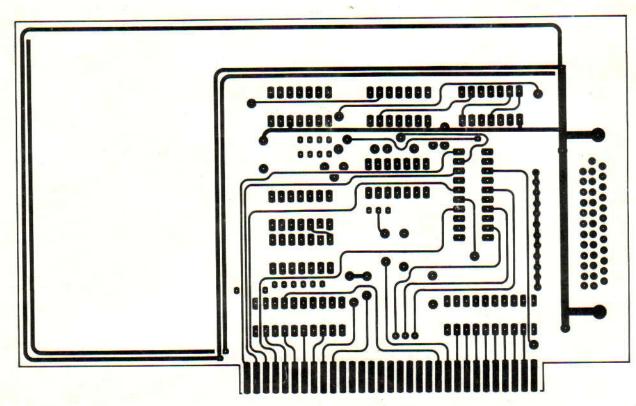
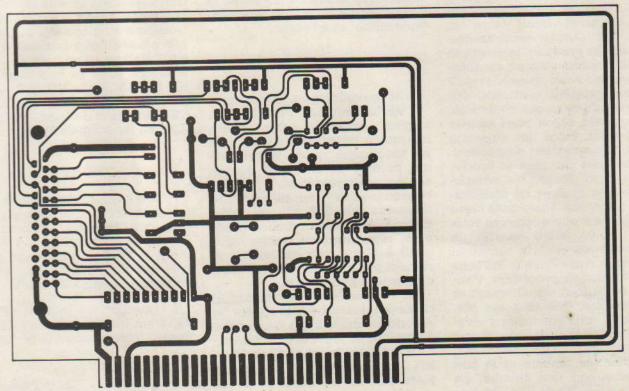
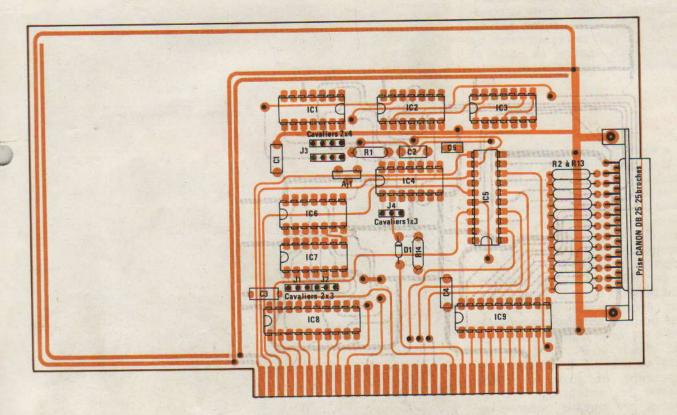


Figure 7







Réalisation

a première tâche à effectuer après la gravure de la carte sera son découpage à la scie, surtout ne partez pas tête baissée, car de la précision de ce découpage va dépendre la position du connecteur dans le slot. 1 millimètre d'écart et l'on risque le court-circuit avec l'endommagement du PC à la clef. Beaucoup

de soin donc, pour que le connecteur n'ait pas de jeu et que les contacts du slot correspondent EXACTEMENT aux pistes gravées sur la carte. Vérifiez avec soin et dites vous qu'il est plus

simple de refaire une autre carte que de dépanner la carte mère du PC. Vérifiez également l'absence de court-circuit entre les pistes. Cette tâche étant accomplie le mieux possible, le montage des composants sera un exercice de routine, nous pensons qu'il est plus prudent d'utiliser des supports; dans ce cas, choisissez des supports qu'on peut souder sur les deux faces, les supports tulipes le permettent en général. N'oubliez pas non plus de souder sur les deux faces, les composants et les queues de résistances de traversée. Une fois la carte terminée, une autre épreuve vous attend : il va falloir résister de toutes vos forces à la tentation d'insérer la carte dans le PC sans autres formes de procédure. Vérifiez et revérifiez, visuellement et à l'ohmmètre, que tout est correct. Nous vous conseillons vivement de connecter la carte à une alimentation 5 V indépendante, de vérifier les tension et le bon fonctionnement des circuits, en particulier IC5, IC8 et IC9. Une fois tous ces tests effectués méticuleusement, si vous avez la conscience tranquille: soyez fatalistes, ce qui doit arriver arrivera de toute façon, insérer la carte et mettez le PC sous tension.

Mise au point et réglage

e seul réglage à affectuer sur la carte d'interface concerne l'ajustable Ajı avec laquelle on freine le PC. Ce réglage est assez précis et dépend de la fréquence d'horloge choisie. Il est nécessaire de le faire à l'oscilloscope en vous aidant du diagramme des temps de la figure 5 (pour exécuter ce réglage, mettez évidemment la carte dans le dernier slot pour avoir accès aux pistes). Vous pourrez ensuite brancher le PRM 4 en toute tranquillité après avoir éteint le PC. L'étape suivante consistera à exécuter le programme de mise au point assistée par ordinateur, celui-ci effectue un certain nombre de tests dont le premier consiste à écrire dans les PIA et à en relire le contenu ; d'ailleurs ce contenu est affiché sur les deux dernières lignes de l'écran, si ce premier test est positif, vous pourrez conclure que la carte d'interface est

opérationnelle, les tests suivants concernent le PRM 4 lui-même et il vous suffira de suivre les consignes affichées à l'écran.

Modifications éventuelles du PRM 4

es modifications concernent exclusivement l'ajout d'un module. Sur maquette, ce module additionnel ne fonctionne qu'en lecture, ce qui permet de n'utiliser qu'une alimentation. consommation supplémentaire nécessite un 5 V un peu plus musclé que nous obtenons grâce à un régulateur en boîtier TO3, exilé à l'extérieur du boîtier et refroidi par un radiateur adéquat, en remplacement de IC7.

La deuxième modification concerne les adresses; pour le second module, il est nécessaire d'éliminer la présence de l'inverseur entre les broches 12 et 13 de IC3. Donc coupez par exemple la piste qui relie la broche 12 de IC3

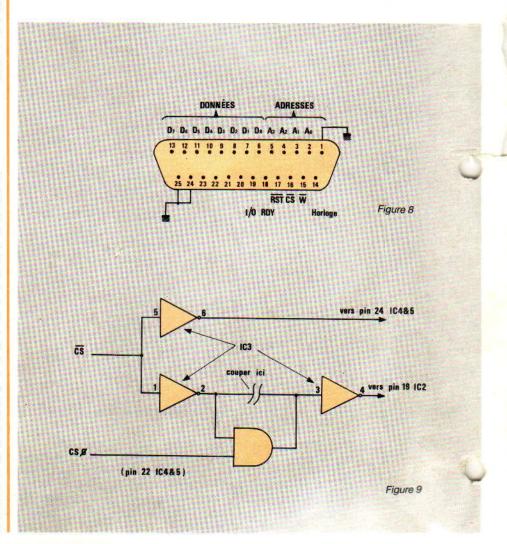
à la broche 22 de IC₄ et reliez directement la broche 22 de IC₄ à la broche 11 de IC₁. Quant à la troisième modification, elle est un peu plus complexe car elle nécessite l'ajout d'un poste AND (74LS08) pour la commande de IC₂ (ENABLE pin 19) qui ne doit plus dépendre que du seul signal CS mais aussi de la valeur du bit A3/CS0 (voir figure 9).

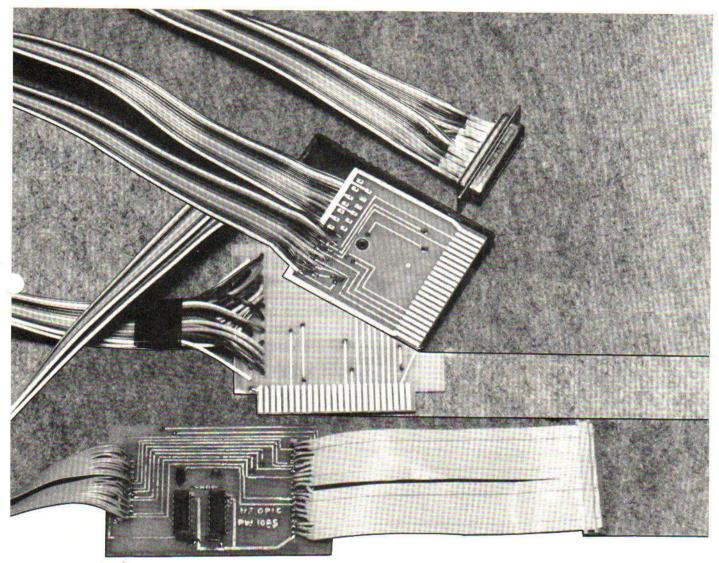
Conclusion

ous espérons satisfaire avec cette carte d'interface parallèle, les lecteurs désireux de connecter à leur PC, non seulement le PRM 4, mais aussi d' tres périphériques déjà décres dans cette revue ou à venir.

L'obtention de logiciel se fera dans un premier temps en en faisant la demande à la revue.

J.-C. Pougnard





Trois interfaces pour le programmateur d'EPROM universel PRM 4

RP-EL Nº 470

omme promis lors de la description du PRM 4, voici les interfaces qui vont vous permettre de piloter notre programmateur d'EPROMS universel grâce à trois ordinateurs bien connus puisqu'il s'agit du COMMODORE 64, de l'ORIC ATMOS et du célébrissime APPLE 2.

Dans notre prochain numéro, nous étudierons l'interface IBM PC.

DESCRIPTION DES INTERFACES

orsque nous avons conçu le PRM 4, nous nous étions fixé comme but de le rendre compatible avec toutes les machines équipées d'un 6502. La partie électronique de l'appareil a donc été prévue dans ce but, lequel fut atteint sans trop de problèmes. Le logiciel qui commande l'appareil peut tourner sur les ordinateurs équipés de 6502 que sont l'ORIC ATMOS et l'APPLE 2. Restait un problème majeur qui consiste à relier le programmateur à l'ordinateur sans casse ni solutions trop sophistiquées.

Pour le COMMODORE 64 et pour l'APPLE 2, la liaison avec le PRM 4 n'a posé aucun problème puisqu'elle se résume à un simple connecteur, ces deux ordinateurs possédant d'origine au moins une ligne permettant un accès très facile à un périphérique. Pour l'ORIC ATMOS, un petit circuit de décodage d'adresse est nécessaire et nous aurons donc affaire à un véritable interface ce qui n'est pas le cas des deux ordinateurs précédents.

Rappelons pour mémoire que le PRM 4 occupe 8 adresses consécutives sélectionnées par A0/A3 et que l'accès aux deux PIA s'opère par la descente du signal CS barre. Celà étant précisé, nous vous invitons à suivre la description des trois solutions proposées et nous débuterons par celle consacrée au COMMO-DORE 64.

Version COMMODORE 64

a liaison de cet ordinateur au PRM 4 s'effectue par le biais

du port d'extension mémoire qui sert par ailleurs à l'exploitation des cartouches de jeux. La totalité des signaux du microprocesseur 6510 (version améliorée du 6502 propre au C64) est présente sur ce port ainsi que diverses lignes de décodage qui vont nous être bien utiles pour ce qui nous occupe. Le port d'extension se présente sous la forme d'un connecteur « nez de carte » comportant 2 rangées de 22 broches au pas de 2,54 mm dont le brochage est indiqué sur la figure 1. La dénomination de chacune des broches est la suivante :

Broche	Broche	Broche	Broche	
1: GND	12 : BA	A: GND	N:CA9	
2:+5V	13 : DMA	B: ROMH	P: CA8	m
3:+5V	14 : CD7	C: RESET	R:CA7	
4:IRQ	15 : CD6	D: NMI	S:CA6	113
5 : CR/W	16: CD5	E: S02	I: CA5	
6 : Dot Clock	17 : CD4	F: CA15	U: CA4	
7: I/01	18 : CD3	H: CA14	V: CA3	
8: GAME	19 : CD2	J: CA13	W:CA2	
9: EXROM	20 : CD1	K: CA12	X:CA1	
10 : I/02	21 : CD0	L: CA11	Y: CA0	
11 : ROML	22 : GND	M: CA10	Z: GND	

Deux signaux sont particulièrement intéressants pour ce qui nous occupe. Il s'agit de I/01 et de I/02 qui passent à l'état 0 pour les adresses comprises entre \$ DE00 et SDEFF (I/01) et entre SDF00 et SDFF (I/02). I/01 est utilisé pour la gestion de la carte Z80 optionnelle et I/02 pour l'interface IEEE également optionnel. La liaison entre le PRM 4 et le C64 s'effectue donc suivant le schéma de la figure 2 et nous voyons que I/01 est relié à CS barre. Le PRM 4 occupe de ce fait les adresses \$DE00/\$DE07 et il est évident qu'il est difficile de faire plus simple!

La partie réalisation se résume

à celle du circuit imprimé double face qui sert de connecteur et dont le tracé est donné à l'échelle 1/1 sur les figure 3 et 4. Ce circuit a été dessiné de manière à pouvoir être logé dans un boîtier de cartouche de jeu ce qui donne un aspect professionnel à l'ensemble. Le câblage est des plus simple puisqu'il suffit d'effectuer les traversées entre pistes en suivant les indications de la figure 5 et de relier le circuit à la prise DB25 du PRM 4 à l'aide de câble plat de 24 conducteurs.

Au niveau logiciel, certaines routines propres au C64 ont été employées et les principa modifications apportées au pro-

Figure 1 : Brochage du port d'extension du Commodore 64 vu de l'arrière de l'ordinateur.

COM	IMODORE 64	PRM 4
(7)	I/O1 O-	
(c)RE	SET O	→ AST (18)
(5) C	R/W o	→ R/W (6)
(E)	SØ2 0-	→ Ø2 (17)
(v)	CA3 O	——— A3 (14)
(w)	CA2 O	→ A2 (16)
(x)	CA1 O	○ A1 (4)
(Y)	CAO	→ A ₀ (3)
(14)	CD7 O	□ D7 (12)
(15)	CD6 O	——○D8 (24)
(16)	CDs O	OD5 (23)
(17)	CD4 0-	O4 (10)
(18)	CD3 O	OD3 (9)
(19)	CD2 O	OD2 (21)
(20)	CD1 O	——◆ D1 (20)
(21)	CDa O	○ D 0 (7)
22AZ)	GNDO	OV (2581)

Figure 2 : Brochages comparés du Commodore 64 et du PRM 4. Les numéros et lettres entre parenthèses correspondent à ceux du port d'extension du C 64 et de la prise du PRM 4. gramme de base touchent la présentation sur l'écran et la génération du « bip » d'erreur. Les routines employées sont les suivantes :

CADRE/SD020: Couleur du cadre (5 = cyan).

FOND/\$D021: Couleur du fond (15 = gris moyen).

READY/SFE66: Retour sous contrôle de BASIC.

INPUT/\$A560: Simule la commande INPUT du BASIC.

ECRINT/SBDCD: Ecrit l'entier dont le MSB est dans X et le LSB dans l'accumulateur.

TESTOP/SF6ED: Teste l'appui sur « STOP ».

SETLFS/SFFBA: Etablit les dresses primaires et secondaires pour l'accès à un fichier.

SETNAM/SFFBD: Prépare le nom d'un fichier.

ECRIT/SFFD2: Ecrit le caractère ASCII dont le code est dans

LOAD/SFFD5: Chargement d'un programme à partir de la cassette ou de la disquette.

SAVE/SFFD8: Sauvegarde sur cassette ou disquette du programme en mémoire.

GET/SFFE4: Lecture du clavier, le code du caractère va dans A.

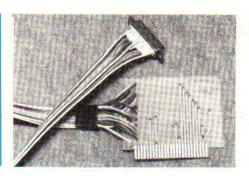
Enfin, et pour conclure, la ligne de BASIC lançant le programme devient : 10 SYS (2063) et le programme est assemblé à partir de l'adresse \$0801. De même que pour les versions C.B.M décrites

ans l'article consacré au PRM 4, nous pouvons fournir le programme d'exploitation du PRM 4 version C64 sur cassette ou disquette et pour ce faire, il vous suffit d'envoyer votre demande à la revue qui transmettra.

Passons à présent à un autre machine utilisant le 6502 et presque aussi répandue que le Commodore 64 : l'Oric ATMOS.

Version ORIC ATMOS

ORIC, comme le COMMODO-RE 64, dispose d'un connecteur d'extension sur les broches duquel sont sortis les signaux du



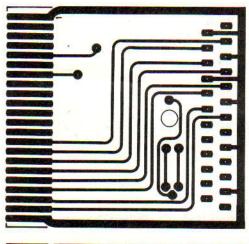


Figure 3 : Circuit imprimé du connecteur côté composants.

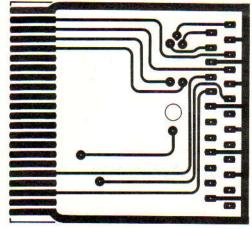


Figure 4 : Circuit imprimé côté cuivre.

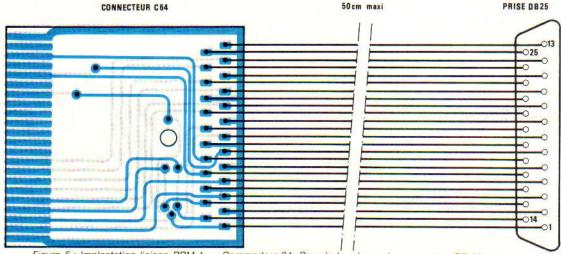


Figure 5 : Implantation liaison PRM 4 → Commodore 64. Pour le brochage du connecteur DB 25 se reporter à la figure 13 de l'article sur le PRM 4.

57

6502 plus quelques informations de contrôle fort utiles pour ce qui nous occupe. Le port d'extension se présente sous la forme d'un connecteur mâle de 2 fois 17 broches au pas de 2,54 mm dont le brochage est indiqué sur la figure 6. L'appellation des diverses broches est la suivante :

Broche	Broche
1: MAP	2 : ROMDIS
3: Ф2	4 : RESET
5 : I/O	6: I/O ctrl
7 : R/W	8 : IRQ
9: D2	10:D0
11 : A3	12:D1
13 : A0	14:D6
15 : A1	16:D3
17 : A2	18 : D4
19: D5	20 : A4
21 : A5	22 : D7
23: A7	26: A14
27 : A8	28 : A13
29: A9	30 : A12
31 : A10	32 : A11
33:+5V	34 : GND

Sur l'ORIC, la page 3 de la mémoire, c'est-à-dire les adresses \$0300/\$03FF, est réservée aux périphériques et le signal I/O passe à l'état 0 lorsque la dite page est sélectionnée. Tout serait donc pour le mieux si le VIA 6522 interne à l'ordinateur n'était situé en \$0300. Fort heureusement les concepteurs de la machine ont prévu une désactivation dudit VIA par le moyen de l'entrée I/O ctrl qui déconnecte le 6522 lorsqu'elle est à l'état 0. Nous avons donc implanté le PRM 4 en \$0320/\$0327 car les adresses \$0310/\$031F sont utilisées par le lecteur de disquettes qui, par conséquent, pourra être branché en même temps que le PRM 4. Le schéma de la figure 7 montre l'extrême simplicité de l'interface qui se borne au décodage de l'adresse \$0320. Le montage utilise deux portes NOR (IC_{1a}et IC_{1b}) ainsi qu'une porte NAND (IC_{2a}) et il est facile de vérifier que le signal CS barre n'est à l'état 0 que si A4, A6, A7 et I/O sont à 0 ET si A5 est à 1 ce qui correspond aux adresses \$0320/ \$032F. La ligne I/O ctrl est, quant à elle, gérée par IC1c et T1 qui est monté en collecteur ouvert de manière à éviter tout conflit avec d'autres périphériques.

La réalisation ne pose aucune difficulté et les composants sont très courants :

- IC1: 74LS02
- Ic2 74LS10
- T1: BC 237B, BC107
- C1: 100 pF 63 V céramique
 C2, C3: 15 nF 63 V céramique
- 1 connecteur à sertir 2 x 17 points
 - 1 prise DB25 mâle
- 20 cm de fil en nappe 34 conducteurs
- 50 cm fil en nappe 12 conducteurs.

Le circuit imprimé est un double face dont le tracé est donné sur les **figures 8 et 9** et sa réalisation ne doit pas poser de gros problèmes. Le câblage s'effectue en suivant les indications de la figure 10 sans oublier les traversées entre pistes ni les soudures côté composants. Nous vous conseillons l'emploi d'un connecteur à sertir pour relier le circuit à l'ORIC et d'installer le montage dans un petit coffret car un accident est très vite arrivé!

Avant de brancher l'interface à l'ordinateur et au PRM 4 il est tout à fait souhaitable de contrôler le montage à l'ohmmètre afin de traquer efficacement tout erreur ou court-circuit.

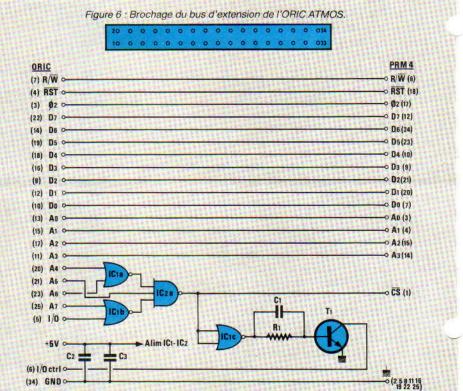
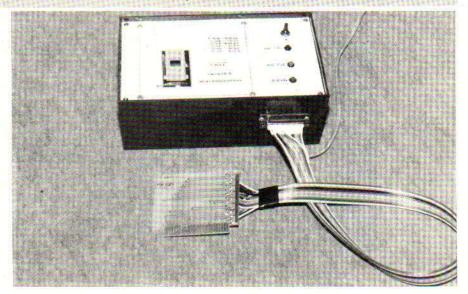


Figure 7 : Schéma de principe de l'interface ORIC/PRM 4. Les chiffres entre parenthèses correspondent à ceux du bus d'extension de l'ORIC et à la prise du PRM 4.



Vous pouvez à présent essayer le PRM 4 sous le contrôle de l'ORIC et nous vous invitons à recopier le court programme BASIC ci-après qui va vous permettre de vérifier le bon fonctionnement de l'ensemble : 100 REM TEST PRM 4 VERSION ORIC ATMOS

110 OG = 800 : REM IORA PIA1 EN \$0320

120 POKE OG + 1,0 : POKE OG+3,0 : REM RESET PIA1

30 POKE OG+5,0: POKE

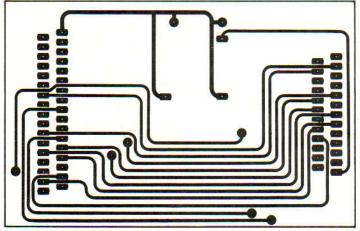


Figure 8 : Circuit imprimé côté composants de l'interface ORIC.

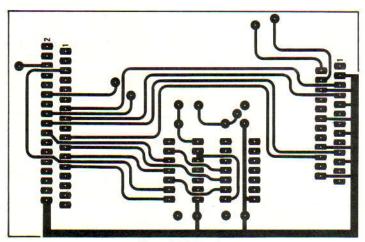
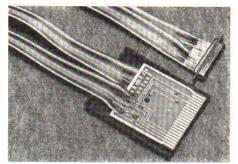


Figure 9 : Côte cuivre.



OG+7,0: REM RESET PIA2 140 POKE OG,255: POKE OG+2,255: REM PIA1 EN SOR-TIE

150 POKE OG+4,0: POKE OG+6,255: REM IORA2 = ENTRÉE, IORB2 =SORTIE

160 POKE OG+5,52: REM CRA2 = %00110100

170 POKE OG+7,52 : REM CRB2 = %00110100

180 POKE OG+6,128: REM ANNULE VCC

190 GOSUB 230 : REM TEMPORI-SATION

200 POKE OG+6,0: REM ÉTABLIT VCC

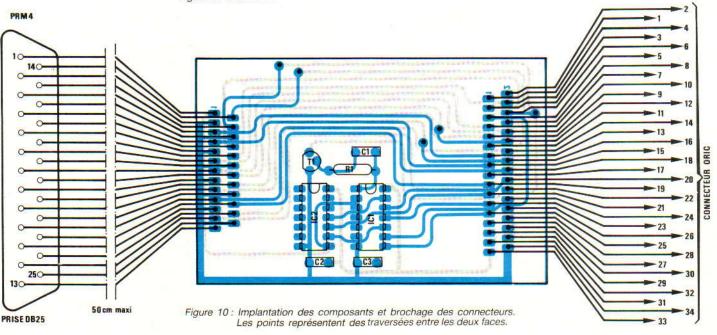
210 POKE OG+7,60 : REM ÉTA-

BLIT VPP 220 GOSUB 230: GOTO 160:

RECOMMENCE 230 FOR I=1 TO 200: NEXT I: RETURN

Dès le lancement du programme, les diodes LED du PRM 4 doivent clignoter alternativement prouvant ainsi le bon fonctionnement de l'ensemble du montage.

Il ne vous reste plus à présent qu'à adapter le programme original afin qu'il fonctionne sur



l'ORIC ce qui ne devrait pas poser trop de problèmes à ceux, et ils sont nombreux, qui connaissent bien cet ordinateur. A ce sujet, si vous parvenez au bout de vos peines, ne manquez pas de nous contacter car les amateurs risquent d'être nombreux!

Il ne nous reste plus qu'à examiner le cas de l'APPLE 2 avant de conclure cette longue étude.

Version APPLE 2

tant donné le nombre très élevé d'ordinateurs APPLE 2 qui existent en France, cette dernière version risque de faire des heureux! L'interfaçage du PRM 4 sur APPLE 2 est aussi simple à réaliser que dans le cas des C.B.M et du COMMODORE 64 et il faut bien avouer qu'il s'agit là de machines semi-professionnelles où tout est prévu d'origine pour faire face à ce genre de problèmes.

L'APPLE 2 dispose de 7 connecteurs d'extension ou « slots » permettant de gérer très simplement toutes sortes de périphériques. Chaque « slot » se présente sous la forme d'un connecteur « nez de carte » comportant 2 fois 25 broches au pas de 2,54 mm et la figure 11 en donne le brochage. La dénomination des diverses connexions est la suivante :

Broche	Broche	Broche
1:I/O SELECT 2:A0 3:A1 4:A2 5:A3 6:A4 7:A5 8:A6 9:A7 10:A8 11:A9 12:A10 13:A11 14:A12 15:A14 17:A15	18: R/W 19: N.C 20: I/O STROBE 21: RDY 22: DMA 23: INT OUT 24: DMA OUT 25: +5V 26: GND 27: DMA IN 28: INT IN 29: NMI 30: IRQ 31: RES 32: IN H 33: -12V 34: -5V	35: N.C 36: 7M 37: Q3 38: PHI 1 39: USER 1 40: PHI 0 41: DEVICE SELECT 42: D7 43: D6 44: D5 45: D4 46: D3 47: D2 48: D1 49: D0 50: + 12V

Il est clair que sur cet ordinateur, nous disposons d'une profusion de signaux mais celui qui nous intéresse pour piloter le PRM 4 est « DEVICE SELECT ». Cette ligne est, en effet, spécialisée pour la commande de périphériques et passe à l'état 0 lorsqu'on accède à l'adresse concernée. Sur le slot 1, « DEVICE SELECT » correspond aux adresses comprises entre \$C090 et \$ C09F, pour le slot 2 ce sera \$ COAO/\$COAF, \$COBO/\$COBF pour le slot 3, etc. Pour notre part, nous avons employé le slot 4 et le PRM 4 occupera donc les adresses \$C0C0/\$C0C7 ce dont il faudra tenir compte lors de l'assemblage du programme.

Compte-tenu de ce qui vient d'être dit, nous aboutissons à un montage qui se résume, comme pour le C64, à un simple connecteur respectant le « SCHÉMA » de la figure 12. Le connecteur ressemble aux cartes APPLE et s'installe directement à l'intérieur de l'ordinateur. Sa réalisation est on ne peut plus simple car il vous suffit de confectionner le circuit double face des figures 13 et 14 et de relier ce dernier au PRM 4 en suivant les indications de la figure 15.

Pour tester l'ensemble, chargez le programme de test décrit plus haut pour l'ORIC en modifiant la ligne 110 en fonction du slot où vous désirez placer l'interface. Si



Figure 11 : Brochage de l'un des slots de l'APPLE 2.

SLOT APPLE 2	PRM 4
Device	
(41) Select O	
(31) RESO-	→ RST (18)
(18) R/W 0	→ R/W (6)
(40) Ø0 O	O Ø2 (17)
(5) A3 ○	O A3 (14)
(4) A2 O	O A2 (15)
(3) A1 O	O A1 (4)
(2) A0 O	O A0 (3)
(42) D7 O-	O D7 (12)
(43) D6 O	→ D6 (24)
(44) D5 O	O D 5 (23)
(45) D4 O	O D4 (10)
(46) D3 O	O D3 (9)
(47) D2 O	O D2 (21)
(48) D1 O	O D1 (20)
(49) D0 0	→ D0 (7)
(26) GND O	O OV (2.58 ti

Figure 12 : Brochages comparés des slots APPLE et du PRM 4. Les numéros entre parenthèses correspondent à la prise du PRM 4 et au slot de l'APPLE.

A propos des logiciels

Le programme complet pour C64 sur disquette est disponible auprès de la société: Votre bureau.

Centre informatique 744 Route Nationale 20 Saran

45400 Fleury-les-Aubrais Tél.: 38.73.30.97

Les logiciels pour les autres machines Commodore (4032, 8032, 8296) sont disponibles auprès de l'auteur. (Faire la demande à la rédaction.)

Enfin, nous n'avons pas étudié le logiciel pour ATMOS et Apple II. vous utilisez le slot 4, la ligne 110 devient :

110 OG=49344 : REM IORA PIA1 EN \$C0C0

Une version améliorée pouvant être imaginée sous la forme :

105 INPUT «SLOT CHOISI: »;NS 110 OG=12 * 4096 +

8 * 16 + NS * 16

De toute manière, le résultat doit être le même que pour l'interface ORIC et les diodes LED du PRM 4 doivent clignoter alternativement dès le lancement du programme.

Au niveau de l'adaptation du programme pilotant le PRM 4, voici quelques routines du moniteur de l'APPLE 2 qui peuvent vous être utiles :

COUT1 /SFDF0: Affiche sur l'écran le caractère ASCII dont le code est dans A (idem ECRIT sur C.B.M.).

BELL /SFF3A: Génère un signal sonore (code ASCII 7 sur C.B.M.). GETLN /SFD6A: Simule l'INPUT du basic, la chaîne lue au clavier va en \$0200 et suite et sa longueur est dans X.

KEYIN /\$FD1B: Lit le clavier ; le caractère ASCII se retrouve dans l'accumulateur (idem GET sur C.B.M.).

Nous voici arrivés au terme de cette étude du PRM 4 et nous espérons que vous serez nombreux à en entreprendre la réalisation. La longueur de cette série d'articles consacrés à ce programmateur d'EPROM vous semblera peut-être excessive mais le sujet en valait la peine puisque vous pouvez à présent disposer d'un appareil très performant et franchir ainsi de nouvelles étapes dans l'exploration du monde passionnant de la micro-informatique. Comme de coutume, nous restons à la disposition des lecteurs dans l'embarras que ce soit du point de vue théorique ou pratique.

Ph. Wallaert

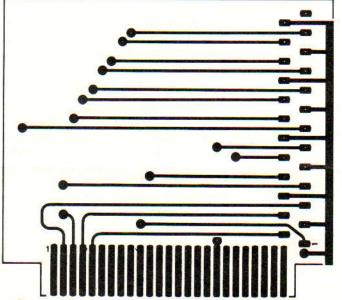


Figure 13 : Circuit imprimé côté composants de l'interface APPLE 2.

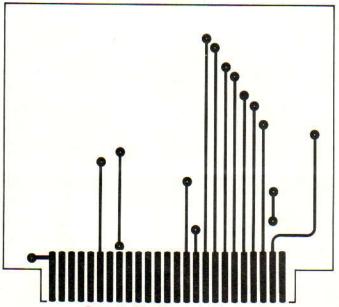


Figure 14 : Circuit imprimé côté cuivre

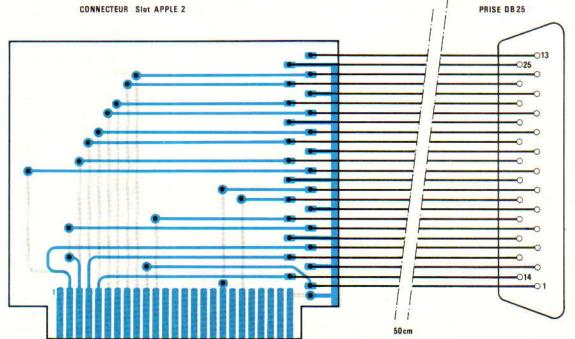


Figure 15 : Implantation et liaison APPLE 2 → PRM 4. Les points figurent les traversées entre faces.

omme nous l'avions promis dans le numéro 478, une nouvelle version du logiciel de commande du PRM 4 version PC et COMMODORE 64 est désormais disponible. Par rapport à la version PC qu'avait créée M. Pougnard, les modifications sont importantes et portent principalement sur la commande de l'effaceur d'EPROMS décrit dans le numéro 478 ainsi que sur l'interface utilisateur. De plus, les commandes reprennent les possibilités de la version COMMODORE qui étaient plus étendues.

En ce qui concerne la version COMMODORE 64, les modifications portent surtout sur de nouvelles possibilités de programmation en ce qui concerne les 2764, 27128 et 27256 ainsi que sur la suppression de deux (petits) « bugs » qui nous avaient été signalés. De plus, la nouvelle version permet la programmation des 27256.

La version IBM PC

e logiciel comporte deux programmes distincts: le programme principal et le programme de configuration.



Le programme principal (GES-PROM. EXE) dispose de toutes les commandes qui vont vous permettre d'exploiter le PRM 4 au maximum de ses possibilités. L'ergonomie, pour employer un terme à la mode, a été particulièrement soignée et l'utilisation du programme est très intuitive. L'action sur l'une ou l'autre des commandes fait appel au curseur lors d'un choix et aux touches de fonctions dans les autres cas. La saisie au clavier n'intervient que lors de l'entrée des adresses ou

lors de l'utilisation de l'éditeur qui va vous permettre de modifier les octets de la mémoire de travail.

Le programme de configuration (INSGESPR. EXE) est chargé de paramètrer le programme principal en tenant compte des caractéristiques du PC. En effet, suivant les unités centrales, l'indice de vitesse d'exécution peut varier de 1 à plus de 10 par rapport à l'IBM PC de base (8088 et horloge à 4,77 MHz). Ce programme va donc créer un fichier

(GESCONF. SEQ) qui sera lu par GESPROM. EXE et contiendra les paramètres suivants :

— Indice pour la création des impulsions de 1 ms nécessaires à la programmation rapide des 2764, 27128 et 27256.

— Indice pour la création des impulsions de 50 ms utilisées lors de la programmation avec un algorithme normal.

— Adresse de base de la mémoire de travail qui est liée à la taille de la RAM.

Type d'affichage (CGA, EGA ou monochrome)

De plus, ce programme permet le réglage de la durée des impulsions sans avoir à effectuer de manipulations, ce qui est très pratique.

GESPROM gère les EPROMS 2716, 2532, 2732, 2732A, 2764, 27128 et 27256 et permet de choisir le type d'algorithme de programmation pour les trois dernières, ainsi que la tension Vpp et la valeur Vcc. L'algorithme employé pour la programmation rapide est très efficace, puisqu'il permet de diminuer le temps d'écriture d'une EPROM dans un rapport de 1 à 4.

Le logiciel est utilisable sur tout IBM-PC ou compatible XT ou AT équipé d'au moins 384 K de RAM et d'une carte d'affichage monochrome ou couleur. Dans la version de base, le PRM 4 occupe les adresses (hexa) 300 à 307 et il faut donc positionner les straps J1 et J2 de la carte d'interface en conséquence (voir N° 471 page 98).

GESPROM et INSGESPR sont écrits en QUICK BASIC MICRO-SOFT et compilés, ce qui garantit une vitesse de traitement tout à fait convenable. Pour fixer les idées, le temps de chargement en mémoire des 32 K d'une 27256 ne prend que 11 secondes sur notre PC (VICTOR VCPII, indice NORTON de 2,4) ce qui est tout à fait honorable.

Nous avions utilisé l'assembleur pour plusieurs fonctions, mais y avons renoncé pour deux raisons. Tout d'abord, la portabilité est moindre ce qui risque d'occasionner un « plantage » sur des ordinateurs peu compatibles. Par ailleurs, le logiciel allait trop vite et le fonctionnement n'était assuré que pour une vitesse d'horloge de 4,77 MHz. Nous

avons donc sagement renoncé à l'assembleur qui risquait de rendre notre logiciel moins portable.

Compte-tenu des caractéristiques de la carte et du logiciel, il vous faut respecter les règles suivantes:

— N'utilisez pas de logiciels résidents (SIDEKICK, PCTOOLS, etc) en même temps que GESPROM, les constantes de temps risquant d'être allongées dans des proportions non négligeables et l'adresse de base de la mémoire de travail risque de ne plus convenir.

— Choisissez la vitesse d'horloge compatible avec la carte d'interface et adaptez le strap J3 à la vitesse choisie.

— Utilisez de préférence un cable blindé multiconducteurs entre la carte et le PRM 4, certains lecteurs ayant éprouvé quelques problèmes à ce niveau.

Pour obtenir le logiciel, la procédure est exactement la même que pour la version COMMO-DORE 64 et « RADIO-PLANS » se charge de la diffusion. La commande du logiciel (référence GESPROM. PC) est à faire parvenir à la revue accompagnée de votre règlement de 50 F par chéque à l'ordre de « RADIO-PLANS ». Pour l'étranger, le tarif est fixé à ce jour à 65,00 F.

Le logiciel est délivré sous la forme d'une disquette 360 K contenant les fichiers suivants :

— GESPROM. BAS: Source en QUICK BASIC.

— INSGESPR. BAS: Source en QUICK BASIC

— GESPROM. EXE: Programme exécutable.

— INSGESPR. EXE : Programme exécutable.

— GESCONF. SEQ: Fichier de configuration de GESPROM.

— TEST1. GES: Fichier de test contenant la mémoire de travail.

GESPROM n'est pas un logiciel du domaine public et sa copie pour d'autres usages que personnels est donc illicite. Par contre, l'auteur s'engage à effectuer les mises à jour en cas de « bug » et vous faire bénéficier de la sorte d'une maintenance technique.

Les sources sont fournies avec le logiciel, ce qui vous permet de l'adapter à vos besoins si vous possédez le compilateur QUICK BASIC et si vous vous sentez le courage de programmer.

Utilisation de INSGESPR

La configuration du logiciel s'effectue grâce à INSGESPR qu'il suffit de lancer après avoir mis le PRM 4 sous tension et effectué les tests de bon fonctionnement. Ce programme permet de déterminer les paramètres suivants:

 Indice pour la création des impulsions de 1 ms.

— Indice pour la création des impulsions de 50 ms.

— Adresse de base de la mémoire de travail.

Type d'affichage.

Comme nous l'indiquions plus haut, ce même programme vous permet de régler les impulsions et de revenir sur les paramètres, ce qui est très pratique. Dés le lancement, vous obtenez l'affichage de l'écran de la figure 1 et il vous suffit d'entrer les valeurs demandées.

La valeur de l'indice pour les impulsions de 1 et 50 ms dépend, bien entendu, de la vitesse de votre unité centrale. Les valeurs indiquées sur la figure 1 conduisent à des temps de 0,96 ms et 46,45 ms sur le VICTOR VPCII que nous avons utilisé pour mettre le programme au point. Sur PC de base (8088) 4,77 MHz), les valeurs indiquées devront être diminuées de près de moitié car l'indice de rapidité de ces machines est plus faible.

L'adresse de base de la mémoire de travail doit être choisie dans une zone « tranquille ». En principe, les adresses au-delà de 4000 hexa peuvent être considérées comme telles et le logiciel ne devrait pas, en principe, employer ces adresses pour y ranger les variables.

Le logiciel permet l'affichage en couleur ou en monochrome. Si vous disposez d'un écran couleur, sélectionnez « 1 » et « 0 » dans le cas contraire.

Pour régler les impulsions de 1 ms, reliez le PRM 4 et branchez votre oscillo sur la broche 22 du support d'EPROM.

Dès que tous les paramètres ont été entrés, la ligne « Réglage... » clignote et vous observez des impulsions négatives sur l'écran de l'oscillo. Si la durée des impulsions est trop importante, appuyez sur « Page Haut » pour remonter au niveau de la saisie de l'indice 1 ms et diminuez-en la

valeur. Validez vos choix jusqu'à revenir sur le réglage et ainsi de suite jusqu'à l'obtention d'une durée variant de 0,95 à 1,05 ms.

Le réglage de la durée des impulsions de 50 ms s'opère exactement de la même manière et le réglage peut être considéré

comme correct pour des durées variant de 45 à 55 ms.

Dès que tous les paramètres ont été saisis, le programme crée le fichier « GESCONF. SEQ » qui sera lu par le programme GES-PROM. EXE.

GESPROM version P.C CONFIGURATION DU LOGICIEL

Indice pour impulsions de 1 m/s :

Indice pour impulsions de 50 m/s:

Adresse mémoire de travail (hexa): 4000

Ecran couleur (1) ou monochrome (0) : 1

Réglage des impulsions de 1 m/s ((Enter) : suite)

Réglage des impulsions de 50 m/s ((Enter) : suite)

Figure 1 - L'écran de travail d'INSGESPR.

- SUPPORT HORS TENSION -

COPYRIGHT P. WALLAERT ET RADIO-PLANS 29/10/1987

CHOIX EPROM CHARGE MEMOIRE

LECTURE EPROM AFFICHAGE MEMOIRE EFFACE MEMOIRE

VERIFIE VIRGINITE COMMANDE LAMPE U.V SAUVE MEMOIRE PROGRAMMATION COMPARAISON EPROM DEPLACE MEMOTRE DIRECTORY

Effacement de la mémoire de travail en cours. Patientez ! Figure 2 - L'écran d'accueil de GESPROM.

Utilisation de GESPROM

GESPROM va vous permettre de piloter le PRM 4 et nous avons prévu un ensemble de commande assez complet qui reprend les possibilités de la version COM-MODORE 64 avec une présentation nettement plus soignée.

Dès le lancement, vous obtenez l'écran de la figure 2 et, en cas de mauvais branchement du PRM 4, le message « LIAISON DÉFECTUEUSE » apparaît en haut de l'écran. Pendant l'affichage du copyright, GESPROM place la valeur hexa « FF » dans les 32 k de la mémoire de travail ce qui revient à l'effacer.

L'écran comporte une zone de travail où apparaissent les données et trois lignes où figurent les commandes. Sur la ligne du haut, un message vous signale la présence ou l'absence de tension sur le support et le type d'EPROM choisi et indiqué sur la ligne du bas. Le choix de l'une ou de l'autre des commandes s'effectue très simplement en déplaçant le « pavé » superposé à la commande à l'aide des touches du curseur et en validant votre choix par « enter ».

La procèdure de sortie de GES-PROM vers le DOS s'effectue en appuyant sur « Esc ». Une demande de confirmation apparaît alors sur la ligne du bas et il suffit d'appuyer sur « F1 » pour confirmer l'ordre ou sur toute autre touche pour l'annuler.

Avant toute utilisation de GES-PROM, il vous faut obligatoirement choisir le type d'EPROM à programmer. Dès la validation de la commande « CHOIX EPROM », vous obtenez l'écran de la figure 3. Si vous désirez une 27128, amenez le « pavé » par les touches « Cur Bas » et « Cur Haut » et validez par « enter ». Pour choisir la tension Vpp, le type d'algorithme et la tension Vcc, procédez comme précédemment mais avec « Cur Droit » et « Cur Gauche ». Si vous revenez dans cet écran, le « pavé » se placera sur vos choix précédents.

Enfin, un appui sur « Esc » vous permet de revenir de suite à l'écran principal.

Dés le lancement de la commande « VERIFIE VIRGINITE », GESPROM lit le contenu de l'EPROM et vous indique le nombre d'octets non effacés. Le temps de lecture est assez rapide et un message en haut de l'écran vous indique que le support est sous tension.

La commande « COMMANDE LAMPE U.V. » fait appel au circuit décrit dans le Nº 478 et l'effacement de l'EPROM devient automatique. Le procédé est très simple: GESPROM teste la virginité de l'EPROM et lance l'allumage de la lampe pendant 1 minute si le test est négatif. Un nouveau test suit l'extinction de la lampe et ainsi de suite jusqu'à l'effacement de l'EPROM. Cette commande est très pratique et prolonge grandement la vie de vos chères (dans tous les sens du terme!) EPROM.

La commande « SAUVE MEMOI-RE » va vous permettre de sauver la mémoire de travail sur disquette ou disque dur. Trois données doivent être saisies :

 L'adresse hexa du début de la zone à sauvegarder.

- L'adresse hexa de la fin de la zone à sauvegarder.

- Le nom du fichier de sauvegarde (8 caractères maxi).

Les fichiers sont sauvegardés avec l'extension « GES » ce qui vous permet de les identifier facilement. L'appui sur « Page Haut » vous fait retourner au niveau de la saisie précédente et « Esc » vous renvoie directement au niveau du menu principal.

« CHARGE commande MEMOIRE » vous permet de récupérer les programmes sauvegardés par la commande précédente et la seule donnée à entrer est le nom du programme. Il est important de savoir que les programmes sont sauvegardés en utilisant la commande « BSAVE » qui affecte au fichier l'adresse de la mémoire de travail ce qui implique qu'un programme sauve-gardé avec une adresse de base ne pourra être récupéré aux mêmes adresses si vous modifiez cette adresse de base. Si le cas devait se produire, il vous faudra lire le fichier à l'aide de DEBUG et modifier les octets 1 (LSB) et 2

```
SUPPORT HORS TENSION -
       2 K/octets Vpp=25 Volts
2716
       4 K/octets Vpp=25 Volts
2532
       4 K/octets Vpp=25 Volts
2732
 2732a 4 K/octets Vpp=12,5 ou 21 Volts
       8 K/octets Vpp=12,5 ou 21 Volts
 2764
 27128 16 K/octets Vpp=12,5 ou 21 Volts
 27256 32 K/octets Vpp=12,5 ou 21 Volts
< Vpp = 12,5 Volts > < Vpp = 21 Volts >

    ALGORITHME RAPIDE > < ALGORITHME NORMAL >

    < Vcc = 6 Volts > < Vcc = 5 Volts >
    VERIFIE VIRGINITE COMMANDE LAMPE U.V SAUVE MEMOIRE
```

COMPARAISON EPROM PROGRAMMATION LECTURE EPROM CHARGE MEMOIRE DIRECTORY DEPLACE MEMOIRE AFFICHAGE MEMOIRE EFFACE MEMOIRE CHOIX DU TYPE D'EPROM A LIRE OU PROGRAMMER

Figure 3 - L'écran de choix du type d'EPROM.

CHOIX EPROM

```
C:\)debug a:test4.ges
- d
1938:0110 89 A9 0A 9D A8 10 BD 12-10 F0 1E C9 24 B0 1A A5 ..........$...
Figure 4 - Examen d'un fichier à l'aide de DEBUG. Ici le programme est logé en 4080
```

exa.							SUP	POR	T H	ORS	TE	NSI	ON -	_	-	-	Control Vontroles
ADRESSE	344							4	CON	TEN	U						CODE ASCII
NDNESCH	-00-	-01-	-02-	-03-	-04-	05-	06-	07-	08-	09-	OA-	0B-	OC-	0D-	OE-	OF-T	TOTAL IS TUDEL OLD
0600	49	53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST 48 TYPEAe+0
0610	42	4C	4F	43	CB	61	83	46	84	62		46				- 1	BLOC TaaFabaFçAaS
0620	49	53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEae O
0630	42	4C	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOCTABFABAFÇAAS
0640	49	53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST-da TYPEae+0
0650	42	4C	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOCTaaFäbaFçAaS
0660	49	53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST Lda TYPEàe O
0670	42		45	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOCTaaFabaFçAaS
0680	40	53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEae 0
0690	1000	40	41	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOCTaaFäbaFçAaS
06A0	1,0	53	5/	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEae 0
	1	4(CE								87	41	83	53	BLOCTaaFäbaFçAaS
0680	100				3 64			54			45	85	65	CE	4F	20	IST da TYPEae+0
0600	1	2 40			10000	200						46				53	BLOCTaaFabaFçAaS
06D0	1		100					1000	59			85		CE	4F	20	IST da TYPEae+0
06E0	4				יט כ											53	BLOCTaaFabaFçAaS
06F0	4	2 4	C 4	F 4:	3 (1	0 01	. 0.	, 40	, 0-								1 1

(Page Haut): page précédente (Page Bas): page suivante (F2): Modification adresse de base (F1): Modification de la mémoire, (Esc): Retour au menu principal

AFFICHAGE DU CONTENU DE LA MEMOIRE DE TRAVAIL 27128 16K/o Vpp=21 Volts Vcc=6 Volts Alg rapide

Figure 5 - L'écran d'affichage de la mémoire de travail.

ADRESSE			210			201	FFU		HOR:	-	LIVO.	LON				CODE ASCII
	-00-01	-02	-03-	-04-	-05-	-06-	-07-	-08-	-09-	-OA-	-0B-	OC-	-OD-	OE-	-0F-	1002 10011
0600	49 53															IST da TYPEàe+0
0610	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC Ta a Faba Fc A a S
0620	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEae+0
0630	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC-aaFabaFcAaS
0640	49[53	1354	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEàe+0
0650	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC_aaFabaFcAas
0660	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST då TYPEåe+0
0670	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC-aaFabaFcAaS
0680	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEàelO
0690	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC_aaFabaFcAaS
06A0	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEàelO
06B0	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC-aaFabaFcAaS
06C0	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEaeto
06D0	42 40	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOC-aaFabaFcAaS
06E0	49 53	54	D3	64	83	20	54	59	50	45	85	65	CE	4F	20	IST da TYPEaelO
06F0	42 4C	4F	43	CB	61	83	46	84	62	83	46	87	41	83	53	BLOCTaaFabaFçAaS

(Curseur Bas): Ligne suivante (Page Haut): Octet précédent (Curseur Haut): Ligne précédente

(Page Bas): Octet suivant

(Esc): Sortie de l'éditeur

Figure 6 - L'écran d'affichage de l'éditeur.

(MSB) qui contiennent l'adresse de base de la mémoire de travail (voir **figure 4**). Si vous n'avez jamais pratiqué **DEBUG**, c'est l'occasion ou jamais!

La commande « LECTURE EPROM » permet d'effectuer le transfert du contenu de l'EPROM vers la mémoire de travail. Les seules données à saisir sont les adresses (en hexa) de la zone à recopier. Par rapport à la précédente version, il est donc possible de ne charger en mémoire qu'une partie du contenu de l'EPROM et non la totalité. Comme vous pourrez le constater, le temps de chargement est très rapide malgré le retour à un BASIC compilé.

La commande « PROGRAM-MATION » va vous permettre de programmer vos EPROMS avec ou sans le recours à l'algorithme rapide. Comme pour la lecture, les seules données à saisir sont les adresses (en hexa) de la zone à programmer. La programmation de l'EPROM s'effectue en trois phases :

 Vérification de la virginité de la zone concernée.

Programmation de l'EPROM.
Vérification de la copie.

Si l'algorithme rapide est utilisé, la phase de vérification s'effectue après la programmation de chaque octet et une erreur entraîne l'arrêt immédiat de la programmation. Comme nous l'avons signalé plus haut, GES-PROM pilote le PRM 4 en programmation avec trois valeurs possibles pour Vpp et permet de choisir Vcc à 5 ou 6 Volts avec l'algorithme rapide car toutes les EPROMS ne supportent pas 6 Volts. Tous les cas de figures sont donc envisagés et vous ne devriez pas, en principe, rencontrer d'EPROM non programmable par le PRM 4.

L'utilisation de la commande « COMPARAISON EPROM » est simple puisqu'il suffit d'entrer les adresses hexa de début et de fin de la zone à contrôler. GESPROM vous indique le nombre d'octets non conformes et vous en indique fugitivement les adresses.

GESPROM est doté d'un moniteur qui va vous permettre de visualiser la mémoire de travail et d'éventuellement la modifier. Le moniteur est accessible par la commande "AFFICHAGE MEMOIRE" et vous obtenez dans un premier temps l'écran de la figure 5 où sont affichés les adresses, la valeur hexa des octets et le code ASCII correspondant. Les commandes à ce niveau sont:

— « Page Bas » : Affichage des 256 octets suivants.

— « Page Haut » : Affichage des 256 octets précédents.

— « F1 » : Accès à l'éditeur qui va vous permettre de modifier le contenu de la mémoire.

— « F2 »: Modification de l'adresse de base des 256 octets affichés.

— « Esc » : Retour au menu principal.

Pour accéder à l'éditeur, il vous suffit d'appuyer sur « F1 » et vous obtenez l'écran de la figure 6. Par défaut la saisie s'opère sur le premier octet de la page ce qui se traduit à l'écran par les signes « [» et «] » entourant celui-ci. Les commandes de l'éditeur sont les suivantes :

— « Cur Bas » : Accès à l'octet de la ligne suivante.

— « Cur Haut » : Accès à l'octet de la ligne précédente.

— « Page Haut » : Accès à l'octet précédent.

— « Page Bas : Accès à l'octet suivant.

— « Esc » : Retour à l'écran précédent.

Un appui sur « Enter » valide la modification de l'octet et l'affichage de son code ASCII.

La commande « EFFACE MEMOIRE » vous permet de placer la valeur hexa « FF » dans une partie de la mémoire de travail et les seules données à saisir sont les adresses (en hexa) de la zone à effacer.

La commande « **DEPLACE MEMOIRE** » est très pratique et vous permet de déplacer une partie de la mémoire de travail d'un endroit à un autre. Trois données doivent être saisies :

 L'adresse de début de la zone à déplacer.

 L'adresse de la fin de la zone à déplacer.

 L'adresse de début de la zone de destination.

Ainsi, si les valeurs des trois adresses sont respectivement « 0000 », « 1400 » et « 2000 », la zone de la mémoire comprise entre les adresses hexa 0000 et 1400 sera déplacée à partir de 2000.

La dernière commande est « **DI-RECTORY** » qui entraîne l'affichage à l'écran des fichiers « GES » figurant sur la disquette ou le disque dur.

Version **COMMODORE 64**

a nouvelle version du logitiel version COMMODORE 64 porte le Nº 3.2 et reprend la même présentation que la précédente et nous n'allons donc pas nous étendre sur le fonctionnement du programme qui a déjà été décrit.

Il est désormais possible de programmer les 27256 en deux passes, la mémoire du 64 ne permettant pas, hélas, de loger les 32 k d'une 27256 en un seul morceau. Le logiciel vous demande donc de préciser l'offset lors de la programmation, la lecture ou la de ce type comparaison d'EPROM. Cet offset est donc de 0000 pour les adresses basses de l'EPROM (0000 à 3FFF hexa). Il ne faut pas perdre de vue que la mémoire de travail occupe toujours les adresses 0000 à 3FFF hexa, ce qui implique que le transfert complet du contenu d'une 27256 sur la disquette s'effectuera en respectant l'ordre des commandes suivantes:

- 1) T 0000 3FFF 2) OFFSET 0000
- 3) S« 0 : PROG. BAS » 0000 3FFF
- T 0000 3FFF 5) OFFSET 4000
- 6) S« 0: PROG. HAUT » 0000 3FFF

« PROG. BAS » contiendra la partie 0000/3FFF de la 27256 et « PROG. HAUT », la partie 4000/ 7FFF. Cela peut vous sembler un peu confus mais quelques manipulations vous montreront que c'est, en fait, très simple.

Les possibilités de programmation ont été accrues et il est désormais possible d'agir sur la tension Vpp, la type d'algorithme et la tension Vcc avec algorithme rapide ce qui n'était pas le cas avec la précédente version. En effet, certaines 2764, 27128 et 27256 ne supportent pas 6 volts en programmation rapide ce qui ce traduit par une destruction immédiate du composant.

Nous avons profité de la mise au point de cette nouvelle version pour éliminer deux « bugs » que nous avaient signalés plusieurs lecteurs.

Tout d'abord, il fallait indiquer 1 octet de plus lors de la sauvegarde sur disquette ce qui conduisait à la perte du dernier octet d'un programme. De plus, le code de relecture des 2732 occassionnait des problèmes avec certains types d'EPROMS. Ces deux erreurs sont maintenant corrigées et GESPROM est désormais garanti « sans bugs ».

L'obtention de la nouvelle version s'effectuera de la même manière que précédemment au prix de 120,00 F pour les nouveaux demandeurs et de 50,00 F pour ceux qui s'étaient déjà procuré le logiciel. Pour ces derniers, la mise à jour s'effectuera par mise à jour de la disquette originale qu'ils voudront donc bien transmettre à la revue avec leur réglement.

Nous espérons que cette nouvelle version du logiciel de commande du PRM 4 vous satisfera et, comme de coutume, nous restons à votre disposition pour toute suggestion concernant le sujet.

Ph. WALLAERT

PRM 4 version IBM/PC. Les figures suivantes vous redonnent les tracés et l'implantation de la carte double face décrite dans le nº 471. Le tableau ci-contre rappelle la position des cavaliers pour la configuration choisie.

Adresse	Positions de	Remarques			
	J1	J2			
300 - 30 F	0 0 0	0 0 0	Adresses de la carte prototype IBM (1)		
310 - 31 F	0 0 0	000	Adresse de la carte prototype IBM (1)		
320 - 32 F	0 0 0	0 0 0	Adresses utilisées par le disque dur		
330 - 33 F	0 0 0	0 0 0	Figure 4		

