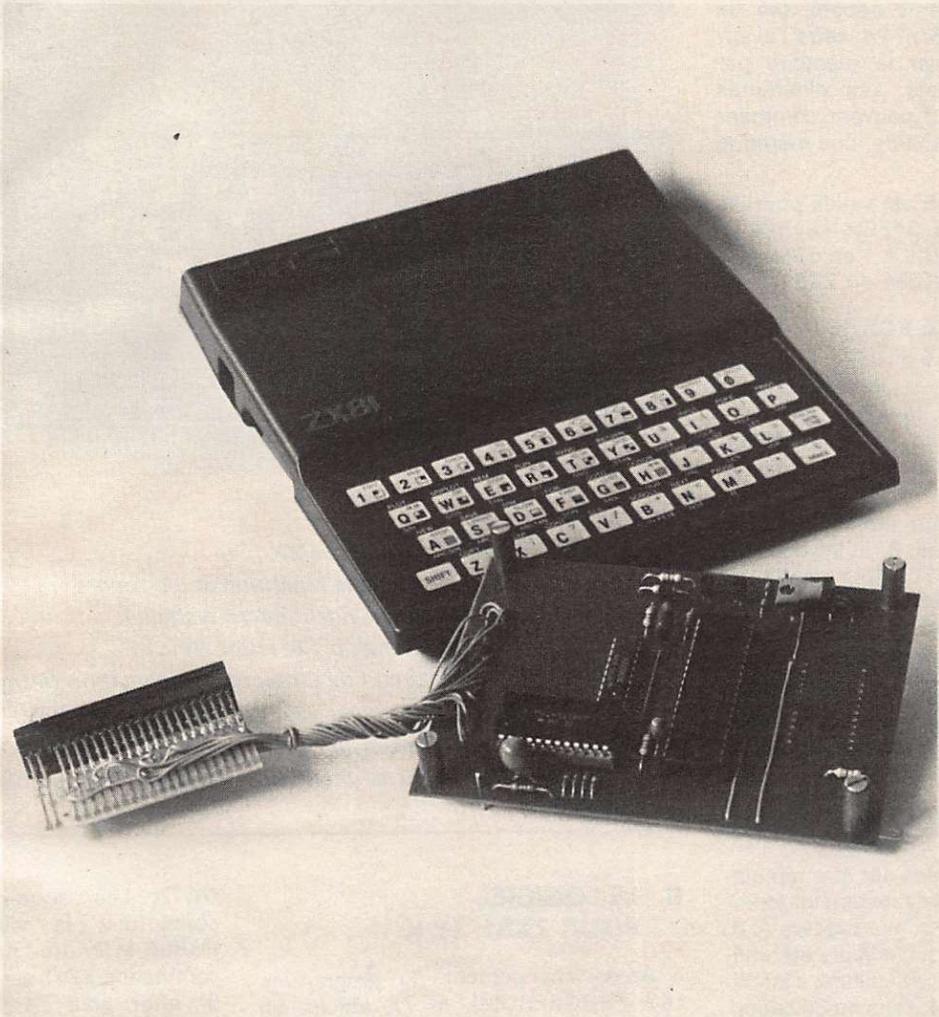


# PROGEPROM



## NOTE DE FONCTIONNEMENT :

① Brancher le connecteur à l'arrière du ZX. Y connecter la 16 K.

② Relier les 3 bornes alimentation + 5 V, + 30 V, 0 V.

③ Charger le programme.

Le menu donne les choix possibles.

Un choix **T** total exécute la manœuvre sur toute la mémoire. **P** impose l'entrée des adresses où l'on désire intervenir :

- les adresses EPROM ou RAM sont par rapport à 0000 H,

- une réponse par **N/L** affecte la valeur :

**07FF** à l'octet final  
**0000** à l'octet de début.

• Les tests et transferts sont instantanés.

• La programmation de la 2716 demande un peu plus de 50mS par octet (≈ 1'45").

## MESURES :

- De nombreuses périodes de fonctionnement prolongé n'ont pas permis de noter d'anomalie.

- Les mesures effectuées au CDA 10M ont porté sur la commutation 5 V/25 V réalisée par le 7805.

Les valeurs correctes ont été relevées.

- L'oscilloscope HAMEG 312 a été utilisé pour contrôler le temps de montée des divers signaux.

Son utilisation principale a été la détermination des paramètres de la double boucle qui donne la temporisation de 50mS lors de la programmation des 2716.

Confronté à l'épineux problème de la programmation des 2716, j'ai écarté le système F8CV (O.C.I. n° 132) qui se suffit à lui-même, mais dont l'utilisation semble astreignante quand on connaît les possibilités du ZX81.

La réalisation d'un programmeur d'EPROM sur plaque pastillée, enfichable à l'arrière du ZX81 ayant donné des résultats satisfaisants, je vous propose la réalisation d'un programmeur d'EPROM sur circuit imprimé relié au ZX par connecteur (pas de faux contact et utilisation facilitée).

Le choix de la 2716 comme support de l'information s'explique par sa disponibilité, pour un prix correct, et une capacité de 2 K suffisante dans la plupart des applications courantes. Cependant des lignes restent disponibles sur le PIA8255 et le circuit est adaptable à d'autres EPROM (2732).

Enfin, le prix de revient de l'ensemble se situe aux environs de 150 F, les composants sont disponibles chez de nombreux annonceurs et la réalisation en circuit imprimé simple face garantit la facilité de reproduction par l'amateur.

## I. LE MATÉRIEL

### 1. SÉLECTION D'ADRESSE

Le mode d'adressage du ZX81 laisse peu de possibilités quant au décodage d'adresse. Le système adopté, tiré de MICRO SYSTÈMES n° 24, offre l'avantage de sélectionner la mémoire par page de 2 K octets. Les différentes sorties du décodeur peuvent commander une extension sonore, une mémoire graphique...

Ici, le P.I.O. 8255 est validé pour les adresses 3800H à 3FFFH. Le BC237 force la ligne ROMCS' à 1 en dehors de la sélection de la ROM du ZX81.

### 2. OUTIL D'ENTRÉE-SORTIE

J'ai choisi le 8255 d'INTEL. Plus coûteux que le 6821, il possède 3 ports d'entrée-sortie, soit 24 lignes ouvrant le ZX81 sur l'extérieur. Son utilisation semble également plus simple que celle du 6821.

Par exemple, pour lire la 2716 P<sub>A</sub> et PC sont des sorties, P<sub>B</sub> lit les données en sortie de la 2716. Pour configurer le P.I.O. dans cet état, il suffit d'écrire 82H dans le registre d'état (AO = A1 = 1) en faisant par exemple :

10 POKE 15003, 130 (130 = 82 H)

Le 8255 est alors configuré. Pour envoyer des données sur le port A :

20 POKE 15000,X X apparaît sur A

Pour lire les données par le port B :  
30 PRINT PEEK 15001

La 2716 pourra bien sûr être remplacée par tout système extérieur (commande de moteurs, convertisseurs A/D, D/A...). Le montage a d'ailleurs été utilisé pour tester des afficheurs « intelligents » type DL1414, sans modification, par branchement sur le support de 2716.

Pour accélérer le programme, les communications avec le P.I.O. se font en langage Machine et non par PEEK et POKE.

L'initialisation du 8255 est assurée à la mise sous tension par un R.C. (4,7 µf/6,8 KΩ).

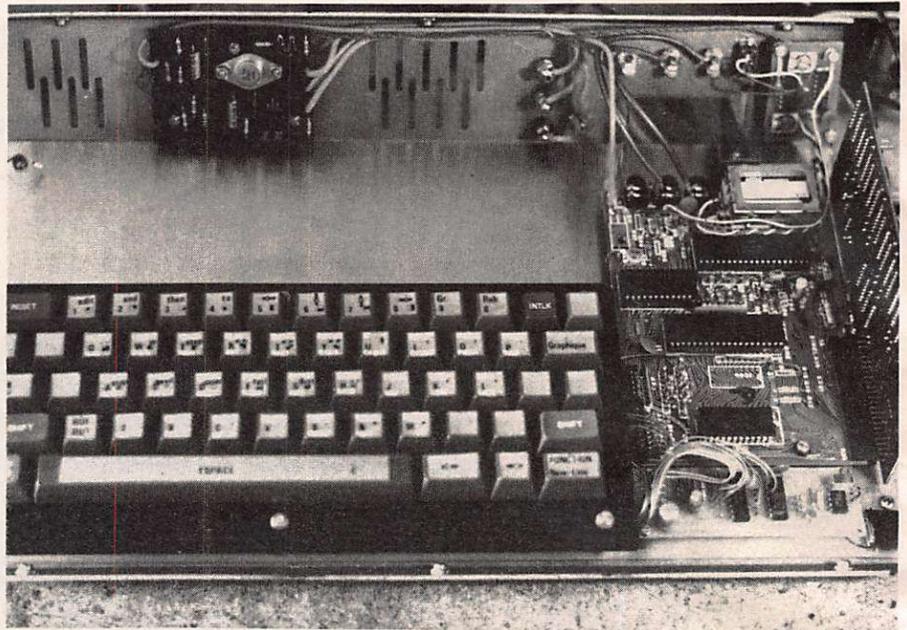
### 3. ALIMENTATION

Une alimentation extérieure « Musclée » fournit à la carte le + 5 V et le + 30 V nécessaire.

Un 7805 dont la broche de « masse » est référencée à 20 V par deux zener de 10 V ramène le +30 V à +25 V en mode PROGRAMMATION.

Le radiateur du 7805 est alors à un potentiel de +20 V par rapport à la masse et devra en être isolé.

Deux BC237 commandés par la sortie Port C6 du 8255 court-circuitent les zeners, ramenant la tension de sortie du 7805 à +5 V, la 2716 est alors en mode LECTURE.



ZX 81:

- clavier mécanique PRO. modifié ZX
- en bas à droite: circuit touche répétition et bip sonore
- en haut à droite: extension vidéo (micro-système)  
les diverses prises (alim, BF...)
- en haut et au centre: circuit de protection alimentation (à thyristors). Protège des inversions de polarité et des surtensions ( $V \geq 5,25V$  et  $\geq 12,3V$ )
- l'emplacement libre devant le clavier recevra d'ici peu les extensions CW +RTTY
- les extensions viennent se brancher en bas à droite (en place lors de la photo: extension 16K)

## II. LE LOGICIEL POUR ZX81 16 K

Après changement du programme (K7 PROGEPRM), le ZX affiche un menu :

- |                          |
|--------------------------|
| 1 - Mémoire vierge       |
| 2 - Transfert 2716 → RAM |
| 3 - Modifier la RAM      |
| 4 - Comparer 2716 et RAM |
| 5 - Programmation        |
| 6 - Sauvegarde           |

NOTA : Les sous-programmes L.M. comportent la valeur 7EH. Il est IMPÉRATIF de ne pas utiliser la commande EDIT sur une REM chargée. En effet, pour ZX, 7EH indique qu'un nombre suit (sur 5 octets). Il traite donc

Vérifie que les octets sont à FF  
Amène le contenu de la 2716 en RAM  
Lecture et modification de la RAM  
Vérifie la similitude 2716 → RAM  
Transfert RAM → 2716 + Contrôle  
Sauvegarde Programme et Contenu de la RAM

Ce programme est un mélange de BASIC (entrée des adresses et de certaines données) et de L.M. (tests répétitifs, transferts, programmation 2716).

Les principaux sous-programmes en L.M. sont stockés dans des REM sur lesquelles le tableau I donne toutes les indications y compris la première adresse du programme L.M.

La REM 1 sert de tampon pour le passage de données entre BASIC et L.M. Son contenu est détaillé au tableau II.

Le tableau III donne le détail des sous-programmes L.M.

ce nombre en conséquence, ce qui modifie la REM I...

L'organigramme permettra de suivre le fonctionnement du programme.

\* Tous les affichages adresses et données se font en HEXADÉCIMAL.

\* Toutes les adresses EPROM et RAM sont par rapport à 0000H, le programme corrigeant les valeurs pour travailler en adresses absolues.

Au chargement, le programme affiche donc le menu. En fonction du choix, il peut demander si la manœuvre à exécuter porte sur l'ensemble des mémoires ou sur une partie seulement. Il est donc

possible d'utiliser en plusieurs fois des 2716, de copier des sous-programmes relogeables à l'endroit voulu...

Pour les choix 1, 2, 4, 5 et 6, le programme ignore les fausses manipulations et donne un compte rendu de son travail.

Le choix 3 affiche une « fenêtre » de 3 données avec leur adresse :

- [R] Renvoie au menu.
- [N/L] Incrémente l'adresse de 1 et fait défiler la fenêtre.
- L'appui sur une touche chiffre  $0 \leq N \leq F$  modifie la donnée de l'adresse en cours (repérée par  $\triangleright$   $\triangleleft$ ). Après introduction de deux chiffres, l'adresse est incrémentée automatiquement.
- [M] fait apparaître en bas de l'écran un bloc de 4 chiffres. Les chiffres sont entrés de gauche à droite. Quand l'adresse est correcte, [N/L] déplace la fenêtre de lecture à l'adresse choisie.

L'affichage dans la fenêtre est fait par lecture SYSTÉMATIQUE des données en mémoire. Donc les octets enregistrés sont immédiatement contrôlés. La lecture est possible de 0000<sub>H</sub> à FFFF<sub>H</sub> mais l'écriture est bloquée au-delà de 07FF<sub>H</sub> (ligne 3100 du Programme).

L'utilisation de ce choix 3 peut être accélérée par l'adjonction de FAST. Mais quand il s'agit de rentrer un programme L.M. octet par octet en contrôlant soigneusement la conformité des valeurs enregistrées, la vitesse semble suffisante.

Pour ma part j'utilise la fonction DUMP du FAST LOAD MONITOR qui donne accès directement aux mémoires. Mais tout le monde ne dispose pas du Fast Load Monitor...

### III. RÉALISATION

Le schéma donne toutes les indications pour une réalisation en Wrapping ou sur plaque pastillée.

Un dessin de circuit imprimé est joint, en simple face pour une réalisation aisée. Il comporte 5 straps.

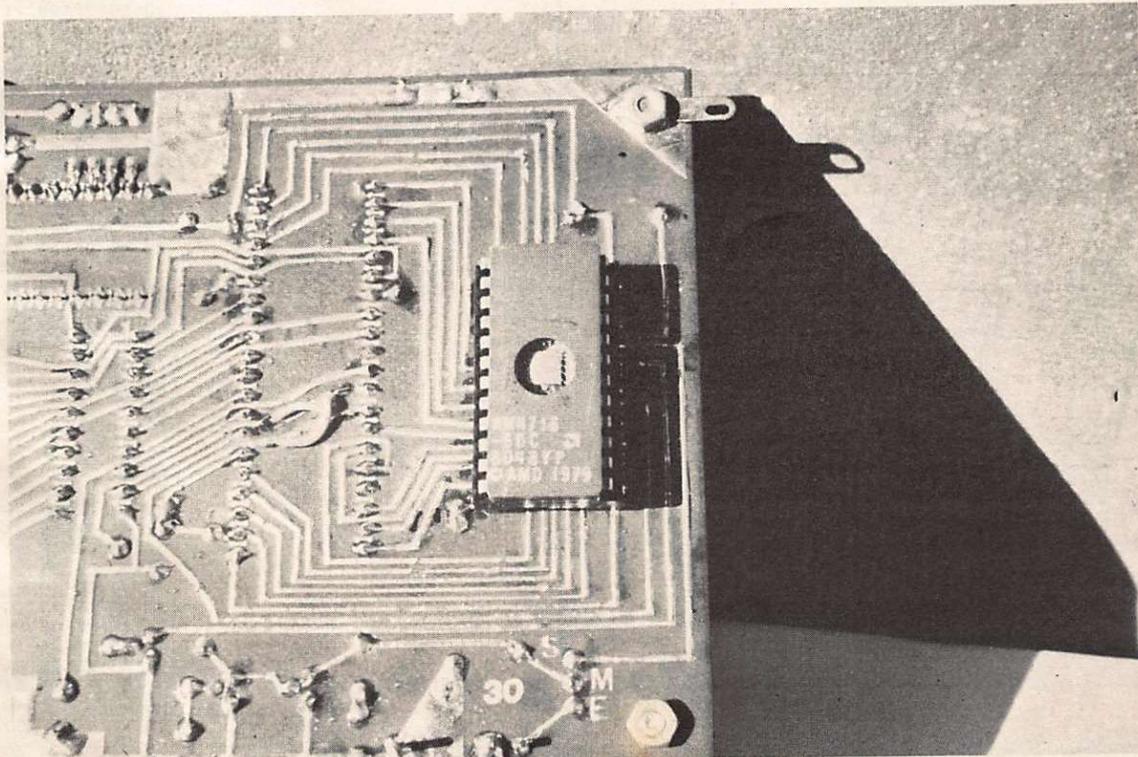
Le support de 2716 est monté *côté cuivre*. Ceci assure un dégagement suffisant pour les manipulations. On peut prévoir de protéger le circuit imprimé par une plaque isolante d'où émergera le support de 2716. Un carré de mousse conductrice collé sur la plaque isolante facilitera le rangement des 2716.

La plaque est reliée au ZX par un connecteur. Cette solution s'avère à la longue plus fiable qu'un circuit monté directement à l'arrière du ZX. En général dans ce cas, l'échange de la 2716 oblige à rentrer à nouveau le programme...

Ce programmeur d'EPROM, pour un prix modique permettra au possesseur d'un ZX81 l'utilisation d'EPROM 2716 pour stocker des routines en langage machine. Son prix peu élevé et sa facilité de réalisation le mettent à la portée de tous.

### NOMENCLATURE

C.I.	7805 (5 V/1,5 A)
	MC14515BCP
	P8255A
	SN74LS245
R	2K2 1/4 W
4X	4K7 1/4 W
	6K8 1/4 W
2X	10K 1/4 W
Divers	C.I. 9 x 12 cm
	Connecteur M/F ZX81
	4 PIEDS
	Visserie
	Supports C.I.
To/DIODES	4X BC237B
	4X 1N4148
	2X Zeners 10 V/0,5 W
C	10nf/250 V
	4,7 $\mu$ f/35 V Tant.
	22 $\mu$ f/35 V Tant.



PROGEPROM VU COTE CUIVRE: la modification du circuit imprimé à gauche de la 2716 qui existe sur cet exemplaire a été rectifiée. Le mylar joint compte de cette modification.

TABLEAU I

N° REM	1 <sup>re</sup> adresse PGM		NBRE D'OC-TETS	TITRE	COMMENTAIRES
	HEXA	DEC			
1	4082	16514	15		Contient diverses adresses, Flags...
2	409C	16540	55	PPAL 1	Teste les octets de la 2716 si # FF Positionne « FLAG » à FF
3	40DF	16607	41	CONF PIO LECT	Configure le PIO, met sur les ports de sortie les valeurs qui font passer la 2716 en lecture.
4	410B	16651	38	CALC LONG	Calcule le nombre d'octets restant à traiter range le résultat en « LONG »
5	4133	16691	19	COMP	Compare les valeurs contenues dans les octets VALLUE et VALCOMP de la REM 1.
6	414E	16712	47	VAL RAM	M et « VALLUE » à l'octet d'adresse ADDRDEST puis incrémente ADDRDEST.
7	4183	16771	49	LIT RAM	Lit l'octet d'adresse ADDRDRAM, range la valeur en « VALCOMP »
8	4189	16825	33	TEMPO	EN FAST Temporisation de 50mS (double boucle) Pour impulsion de programmation.
9	41E1	16865	55	LIT PIO	Lit l'octet adressé sur la 2716 via le port B. Range la valeur en « VALLUE ».
10	421D	16925	54	PPAL 2	Transfert 2716 en RAM.
11	4259	16985	53	PPAL 4	Compare 2716 et RAM.
12	4294	17044	119	PPAL 5	Programme la 2716.
13	430F	17167	2055	RAM	Réserve de 2K.

Tableau des 13 REM indiquant le nombre total d'octets (titre compris) de chaque REM et l'adresse du PREMIER octet UTILE (Début du Programme L.M.). L'indice des « PPALn » est le même que celui du menu.

Ex. : PPAL 1 correspond au choix 1 « 2716 VIERGE » et effectue le test des octets ( FF ), pour vérifier l'effacement complet de l'EPROM.

TABLEAU II

Contenu des octets de la REM 1

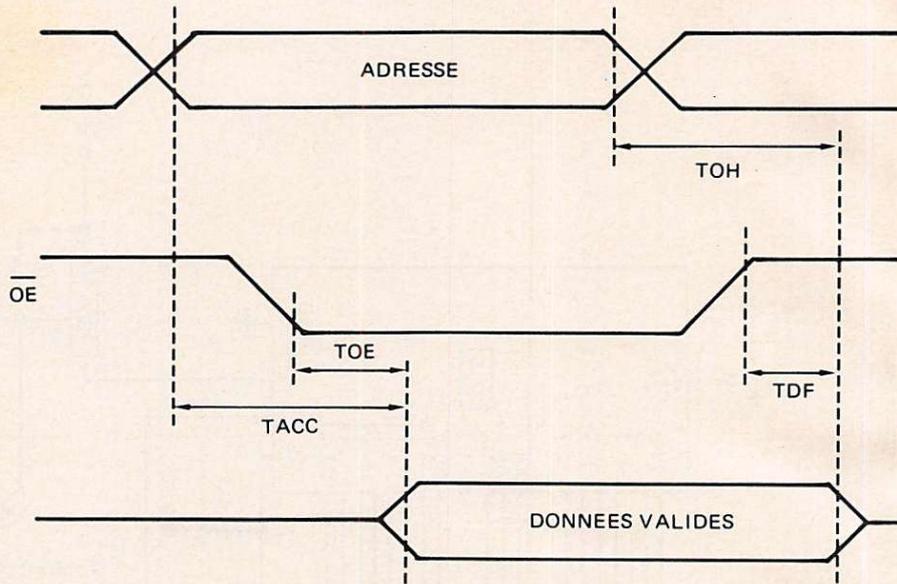
ADR. HEXA	ADR. DEC	ÉTIQUETTE	COMMENTAIRES
4082	16514		Poids faible
		ADR DEB	Adresse EPROM à traiter
4083	16515		Poids fort
4084	16516		P.f.
		ADR FIN	Adresse fin traitement EPROM
4085	16517		P.f.
4086	16518		P.f.
		ADR DEST	Adresse traitement RAM
4087	16519		P.f.
4088	16520		P.f.
		LONG	= ADR DEB - ADR FIN + 1
4089	16521		P.f.
408A	16522	FLAG	Passage d'informations L.M. → BASIC
408B	16523	VAL LUE	VALEUR LUE en EPROM
408C	16524	VAL COMP	VALEUR LUE en RAM

TABLEAU III

CONTENU DES REM : à partir de l'adresse suivant immédiatement le « REM »

REM 1 = 15 octets	4082	FF 07 FF 07 FF 07 02 00 FF 10 00 80 80 80 80
REM 2 = 55 octets	4097	35 35 26 31 1D 3E 00 32 8A 40 CD DF 40 CD E1 41 3E FF 32 8C 40 CD 33 41 28 06 3E FF 32 8A 40 C9 CD 0B 41 C8 18 E4 80 80 82 B4 AB 86 80 AB B7 A6 B3 AB B4 AE B8 80 80
REM 3 = 41 octets	40D4	28 34 33 2B 35 2E 34 31 2A 28 39 3E 82 32 03 38 21 83 40 ED 6F 3E 01 ED 67 C9 80 80 80 88 AA B5 B7 B4 B2 8B 80 80 80 80 80
REM 4 = 38 octets	4103	28 26 31 28 31 34 33 2C 2A 84 40 ED 5B 82 40 A7 ED 52 23 ED 63 88 40 C9 80 80 80 80 80 80 17 9C 9D 98 A4 9F 17 80
REM 5 = 19 octets	412F	28 34 32 35 3A 8B 40 21 8C 40 BE C9 80 80 80 80 80 80 80
REM 6 = 47 octets	4148	3B 26 31 37 26 32 2A 86 40 11 0F 43 3A 8B 40 19 77 2A 86 40 23 ED 63 86 40 C9 A1 A6 B1 B1 AA AA 96 AD B4 AB AD AE B2 AE B3 AD 96 A6 B5 A4 A3
REM 7 = 49 octets	417D	31 2E 39 37 26 32 2A 86 40 11 0F 43 19 7E 32 8C 40 2A 86 40 23 ED 63 86 40 C9 80 80 80 A2 A5 A3 9C 9C 80 AC AE BB B4 B7 B8 80 80 80 80 80 80 80 80
REM 8 = 33 octets	41B4	39 2A 32 35 34 00 00 00 06 2C 3E FF A7 3D 20 FD 3C 10 F7 C9 80 80 AB 9D AC BE B9 98 80 80 80 80 80
REM 9 = 55 octets	41DB	31 2E 39 35 2E 34 21 00 38 3A 82 40 77 3A 83 40 23 23 77 2B 7E 32 8B 40 21 83 40 ED 6F 3E 00 ED 67 2A 82 40 23 ED 63 82 40 C9 80 80 80 80 80 80 80 80 80 80 80 80 80
REM 10 = 54 octets	4218	35 35 26 31 1E 21 82 40 36 00 23 36 00 23 36 07 23 36 FF 23 36 00 23 36 00 01 00 08 CD DF 40 CD E1 41 CD 4E 41 0B 78 B1 C8 C3 34 42 80 80 80 80 80 80 80 80 80
REM 11 = 53 octets	4254	35 35 26 31 20 3E 00 32 8A 40 CD DF 40 CD E1 41 CD 83 41 CD 33 41 28 06 3E FF 32 8A 40 C9 CD 0B 41 21 88 40 7E 23 B6 C8 C3 5E 42 80 80 80 80 80 80 80 80 80 80
REM 12 = 119 octets	428F	35 35 26 31 21 3E 00 32 8A 40 3E 80 32 03 38 CD 83 41 21 82 40 7E 32 00 38 23 ED 6F 3E 04 ED 67 7E 32 02 38 3A 8C 40 32 01 38 ED 6F 3E 06 ED 67 7E 32 02 38 CD B9 41 ED 6F 3E 04 ED 67 7E 32 02 38 CD DF 40 CD E1 41 CD 33 41 28 06 3E FF 32 8A 40 C9 CD 0B 41 21 88 40 7E 23 B6 C8 C3 99 42 00 00 00 00 00 96 AB B4 B5 BE B7 AE AC AD B9 80 9C 9D 98 A4 9F 80 80 80
REM 13 = 2 055 octets	430C	37 26 32 X X X X X X X X X X X X X ↓ 2K RAM
	480C	X X X 80 80 80 80

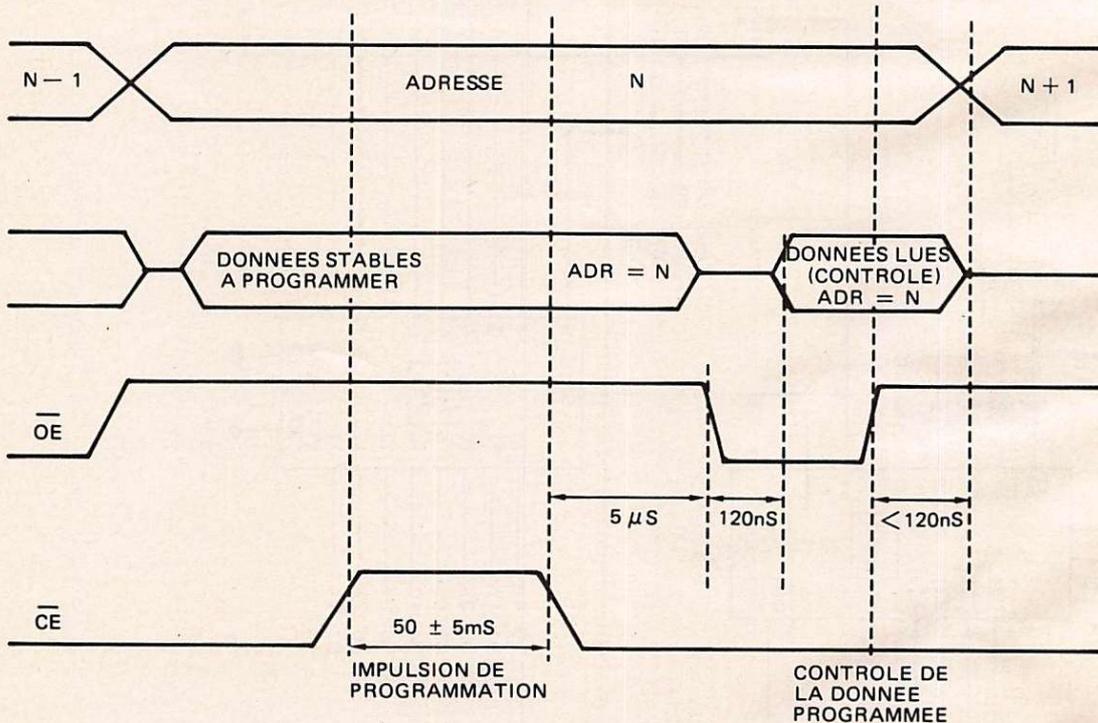




2716 MODE LECTURE ( $\overline{CE} = 0$ )

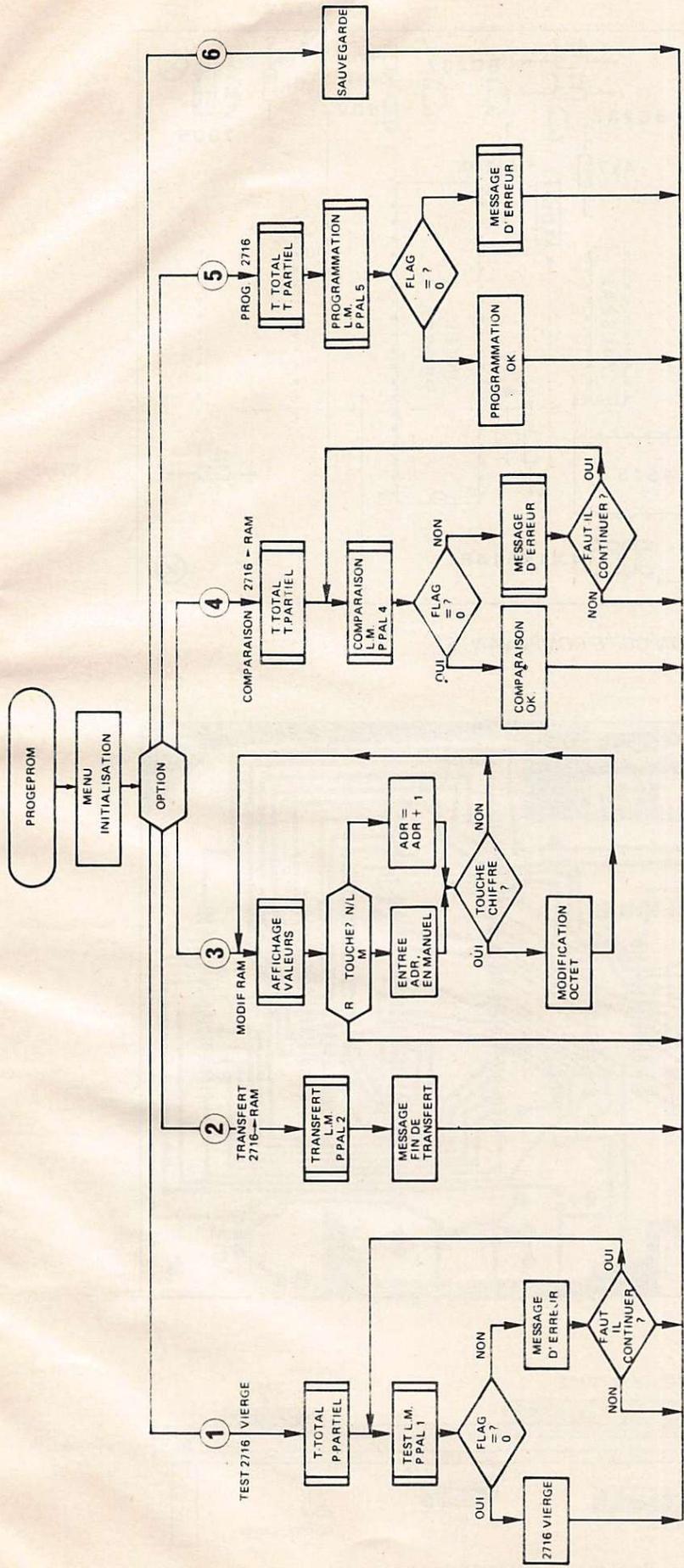
TOE: retard d'apparition des données après OE bas  $< 120\text{nS}$   
 TACC: temps d'accès  $< 450\text{nS}$   
 TOH: retard de disparition des données  
 TDF: retard au passage en Hte impédance  $0 < \text{TDF} < 100\text{nS}$

$V_{cc} = 5V \pm 5\%$        $V_{pp} = V_{cc} \pm 0,6V$        $62 < I_c < 100\text{mA}$   
 MODE Stand By:  $\overline{CE} = 1$        $21 < I_c < 35\text{mA}$

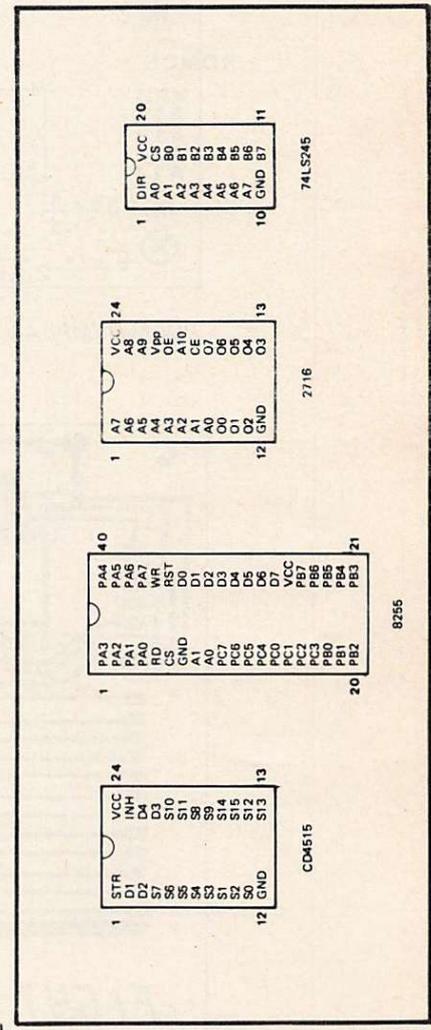


2716 MODE PROGRAMMATION ( $V_{pp} = 25V$ )

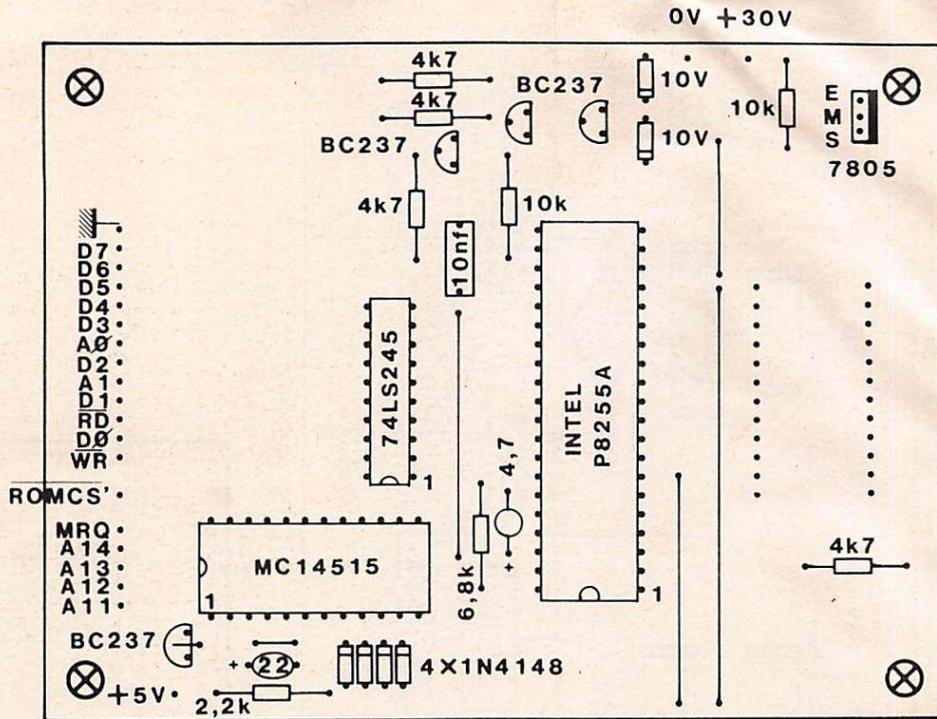
$V_{pp} = 25V \rightarrow I_{vpp} \leq 30\text{mA}$        $V_{pp} = 5V \rightarrow I_{vpp} < 6\text{mA}$   
 $V_{cc} = 5V$        $I_{cc} \leq 100\text{mA}$



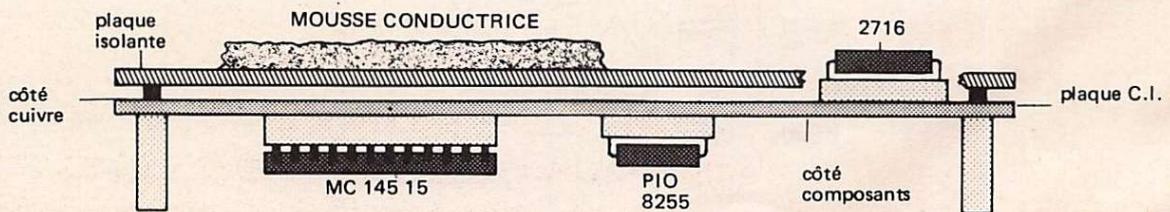
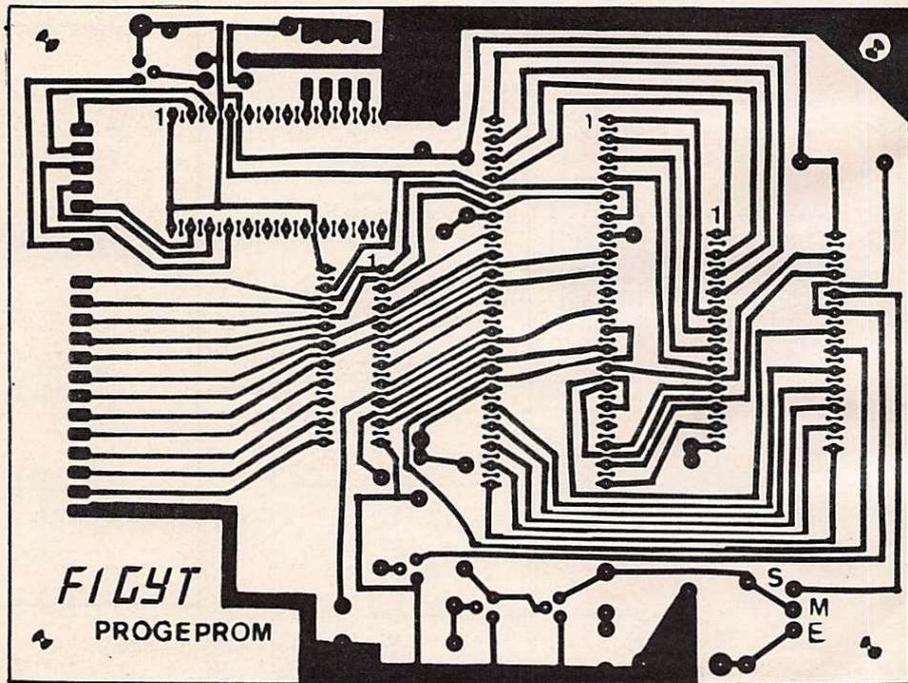
PROGEPROM: ORGANIGRAMME



BROCHAGE DES C.I. UTILISES



PLAN D'IMPLANTATION COTE COMPOSANTS



REALISATION MATERIELLE