

extensions

MSX

une
cartouche
universelle

(2ème partie)

Conçue intelligemment, une platine de fabrication maison, prévue pour recevoir une cartouche du commerce, ouvre des perspectives bien plus variées qu'il n'y paraît au premier abord. Le second article de cette série consacrée à l'établissement de contacts entre les ordinateurs MSX et le monde extérieur illustre quelques-unes des possibilités offertes par une telle platine.

Dans le premier article consacré à l'interconnexion d'un ordinateur MSX au monde extérieur, (pages 34 et suivantes), nous laissions envisager qu'une cartouche du commerce permettait bien d'autres applications que de simples jeux. Poursuivant nos recherches, il nous faut bien avouer avoir été nous-mêmes surpris par les perspectives qu'ouvrirait l'implantation d'une cartouche spéciale dans le connecteur prévu à cet effet, ou sa connexion au port d'E/S. Les quelques lignes suivantes décrivant les possibilités de ce circuit imprimé ne devraient pas manquer de vous mettre l'eau à la bouche.

- 1. Connexion aisée d'extensions matérielles de toutes sources, telles que le bus d'E/S universel (décrit en mai 1985).
- 2. Associé à un morceau de câble plat à 50 brins, il peut être connecté à l'extension de bus d'E/S du Spectravidéo, pour transformer ce dernier en connecteur pour cartouche.
- 3. L'adjonction d'un connecteur encartable de 2 x 25 broches femelle le transforme en connecteur d'extension en équerre.
- 4. Il peut être utilisé comme organe de couplage pour un module synthétiseur de Yamaha (module vendu séparément par certains revendeurs).
- 5. Ce circuit imprimé peut bien évidemment être utilisé en cartouche MSX standard, le type d'EPROM implanté étant fonction de la taille du programme (16...256 Kbits).
- 6. Cette platine constitue en outre un auxiliaire précieux lors de la mise de programmes en EPROM...

Nous ne pourrions malheureusement pas entrer dans le détail de toutes les possibilités offertes par cette dernière propriété. Nos lecteurs ne manqueront pas d'en découvrir d'autres.

Voici de quoi vous occuper pendant de longs mois et à quel prix? celui du petit circuit imprimé double face que nous allons décrire.

La cartouche MSX

Après le récapitulatif non exhaustif de quelques-unes des fonctions de notre platine "cartouche MSX", qui n'assure en fait que la liaison et l'adaptation mécanique, il est bon de s'intéresser aux autres aspects de cette adjonction. Pour mieux comprendre les différents éléments entrant en jeu lors de l'utilisation d'une cartouche, il nous faut entrer dans le concept matériel de la norme MSX et voir comment démarre l'ordinateur, quelle est la différence entre une cartouche de ROM et une extension de RAM,

quelle est la cartographie mémoire de ce type d'ordinateur etc...

Commençons par le début:

■ Le BASIC MSX commence par voir quelle est la taille de la mémoire disponible entre les adresses 8000 et BFFF. Il en fait de même en ce qui concerne l'espace mémoire compris entre C000 et FFFF; le BASIC essaie de découvrir un bloc mémoire unique commun aux deux espaces et active ensuite le bloc mémoire le plus important disponible. L'ordinateur sachant où il en est quant à l'organisation de l'espace mémoire, peut ensuite se consacrer à la vérification de la présence ou non d'une cartouche.

■ L'ordinateur s'intéresse au contenu des connecteurs présents dans l'espace mémoire compris entre les adresses 4000 et BFFF, chaque connecteur possédant son espace mémoire propre de 16 K subdivisé en 4 pages. Le BASIC MSX recherche en début de chaque page un code d'identification spécifique à chaque connecteur, une sorte de

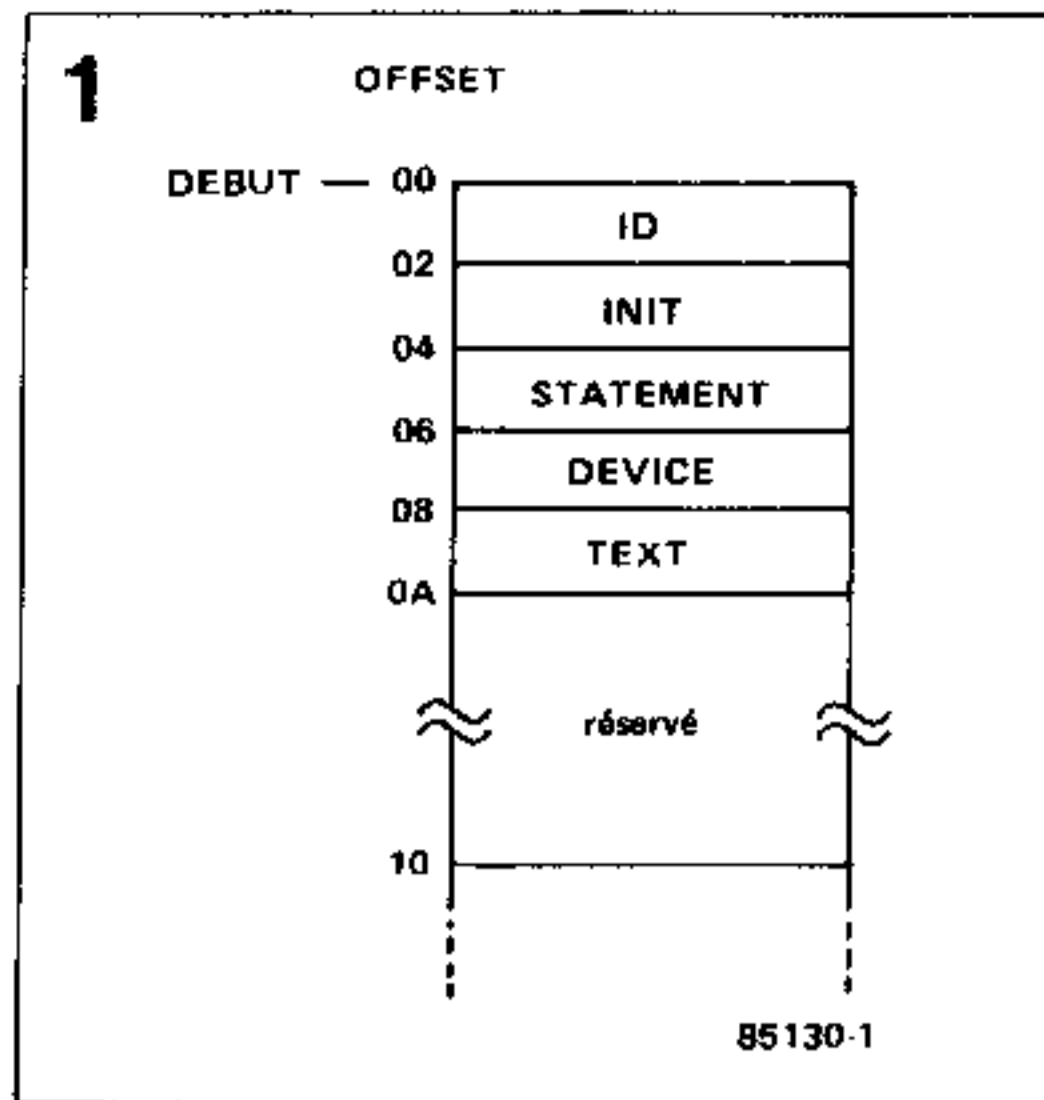


Figure 1. La présence de ces données en début d'EPROM est nécessaire pour que le logiciel sache où il en est.

carte de visite indiquant le type du contenu implanté dans le connecteur. Cet identificateur est toujours présent au début de programme en ROM ou en EPROM. La figure 1 donne la configuration type de ce code.

ID (Identification): ces deux octets font la différence entre une page vide et une ROM. Dans le dernier cas, ID contient les valeurs 41_{hex} et 42_{hex}, c'est-à-dire le caractère ASCII "A" ou "B".

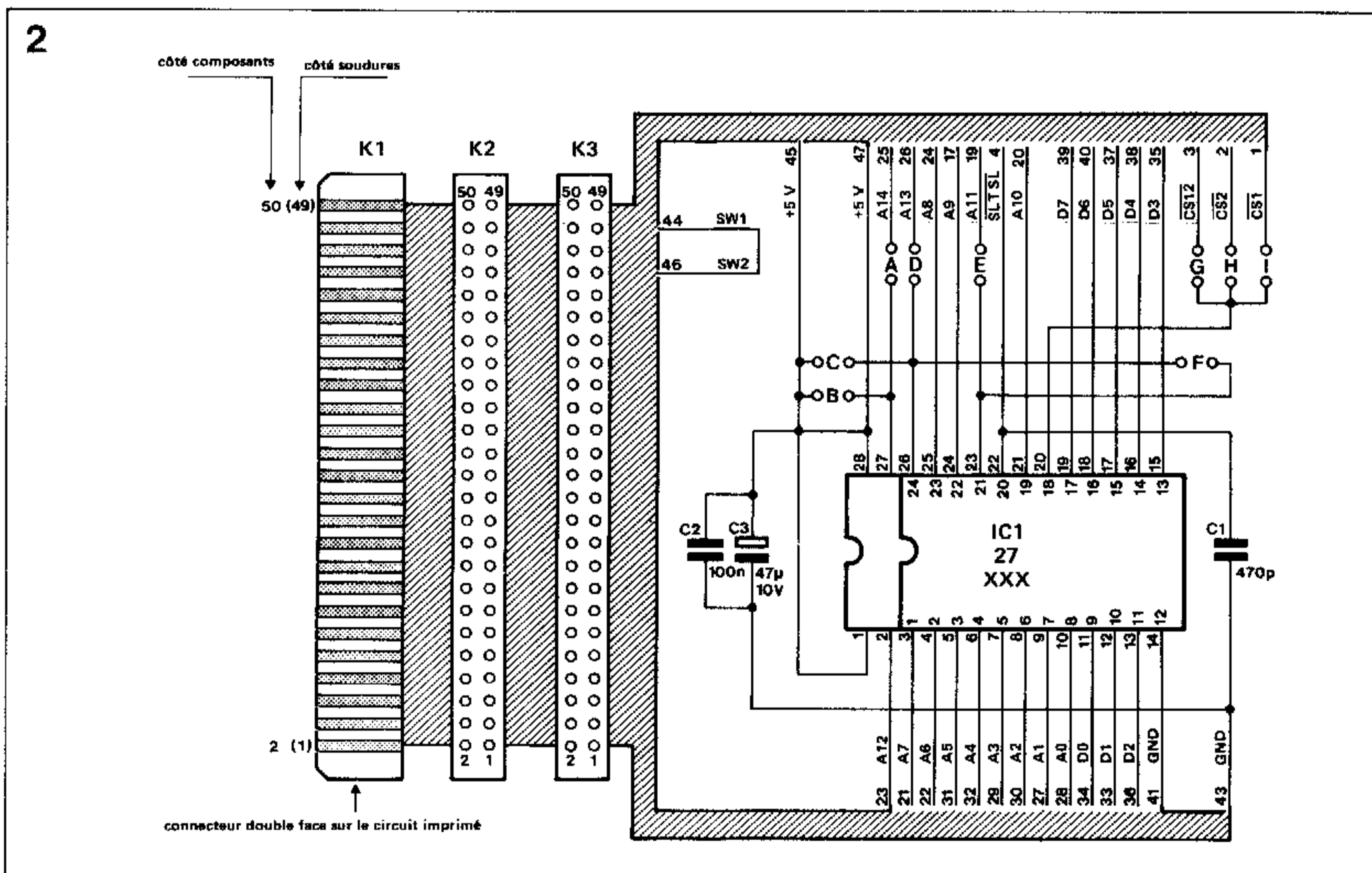
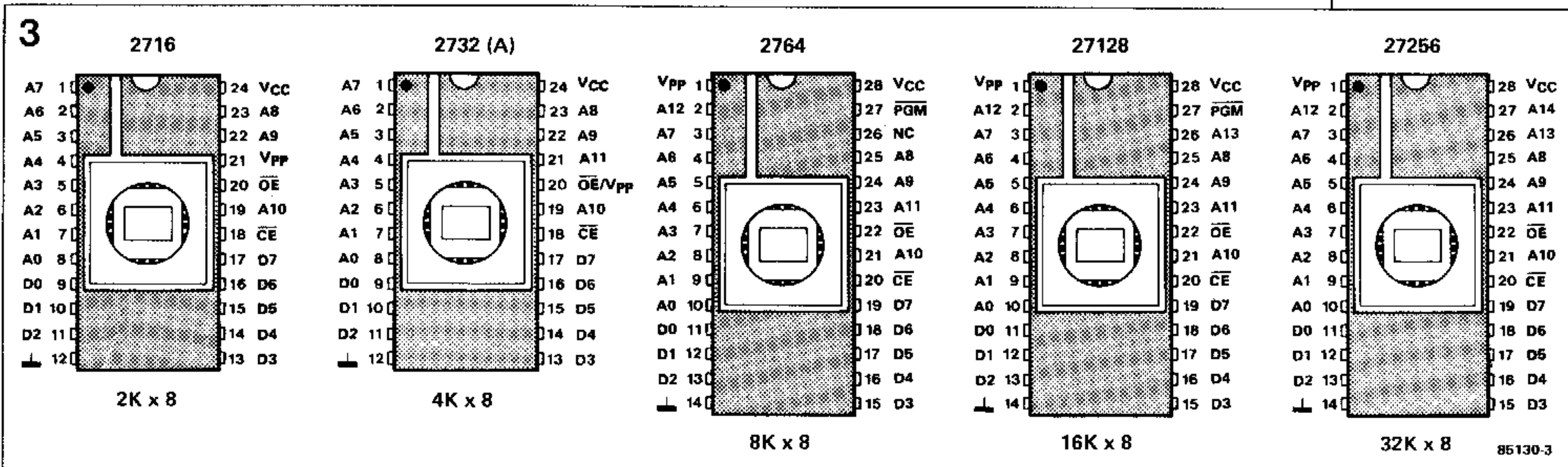


Figure 2. Schéma du circuit pour cartouche. La platine est conçue pour recevoir n'importe quelle EPROM à l'appellation comprise entre 2716 et 27256.

Figure 3. Broches comparées des EPROM utilisables. L'implantation de cavaliers permet de définir le type d'EPROM choisi.



Listing 1. Ce programme de vidage mémoire offre de belles perspectives lors de l'analyse de programmes BASIC résidents en RAM; il donne une liste de tous les octets en format hexadécimal présents à n'importe quel endroit de la RAM. Il peut être mis en EPROM pour constituer un outil de travail très pratique.

Listing 1.

DUMP

```

10 CLS
20 INPUT "start";A
30 INPUT "end";B
40 FOR C = A TO B
50 LPRINT USING "\ \";HEX$(C);:LPRINT " ";
60 FOR D=0 TO 15
70 LPRINT USING "\\";HEX$(PEEK(C+D));:LPRINT " ";
80 NEXT
90 C=C+15:LPRINT " ":LPRINT " "
100 NEXT
110 END
    
```

Liste des composants

Condensateurs:

- C1 = 470 p
- C2 = 100 n
- C3 = 47 µ/10 V

Semiconducteurs:

- IC1 = 2716, 2732(A), 2764, 27128, 27256 (EP)ROM, selon l'application prévue

Divers:

- K3 = 3 socles pour connecteur encartable au pas de 2,54 mm (2 x 3, 2 x 2 et 2 x 4)
- 4 cavaliers enfichables un connecteur encartable femelle 2 x 25 broches
- Et pour Spectravidéo uniquement**
- K2 = socle encartable 2 x 25 broches au pas de 2,54 mm
- 2 connecteurs femelle enfichables pour câble plat 50 brins
- un morceau de câble en nappe 50 brins de 7 cm de longueur

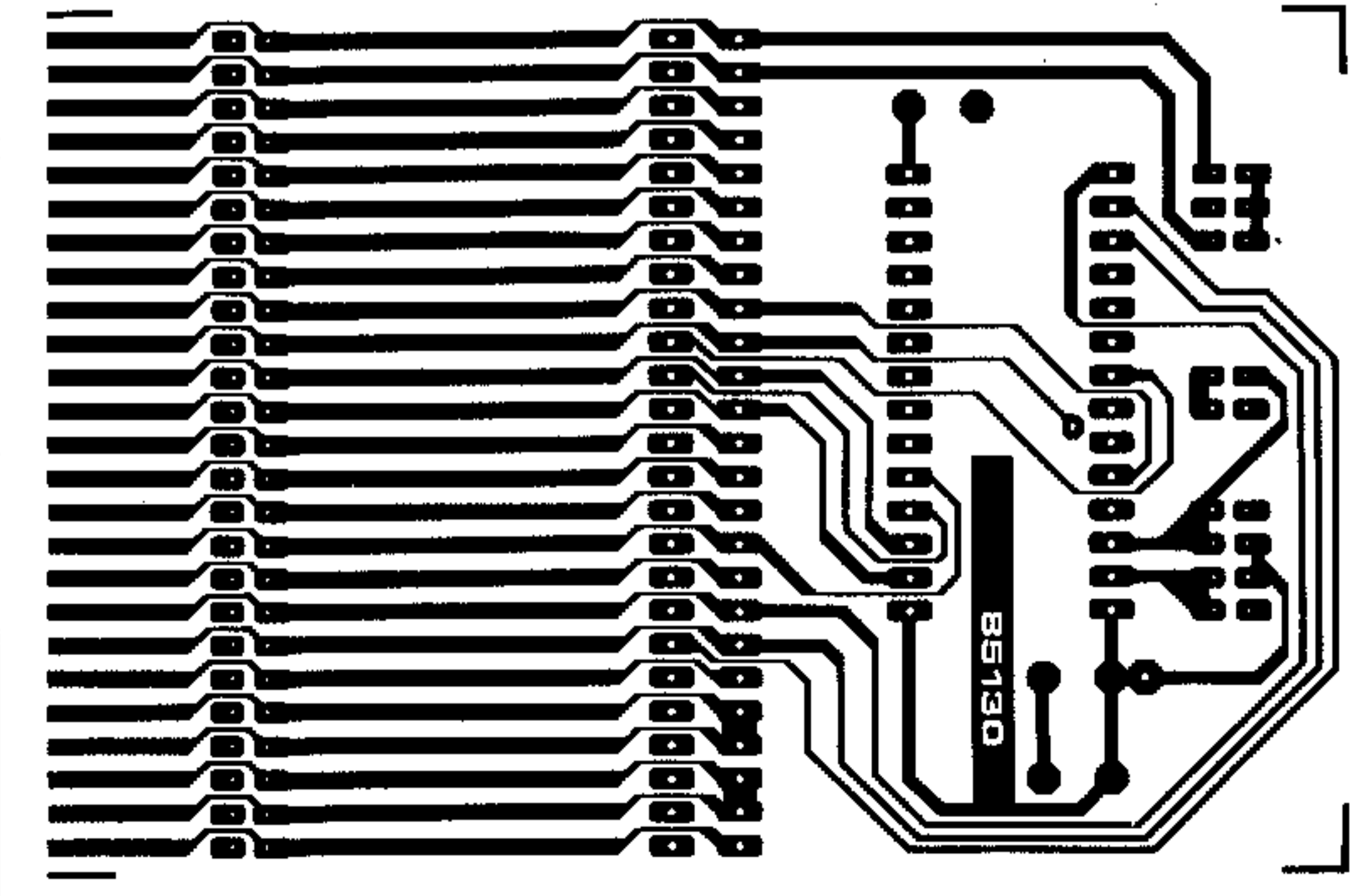
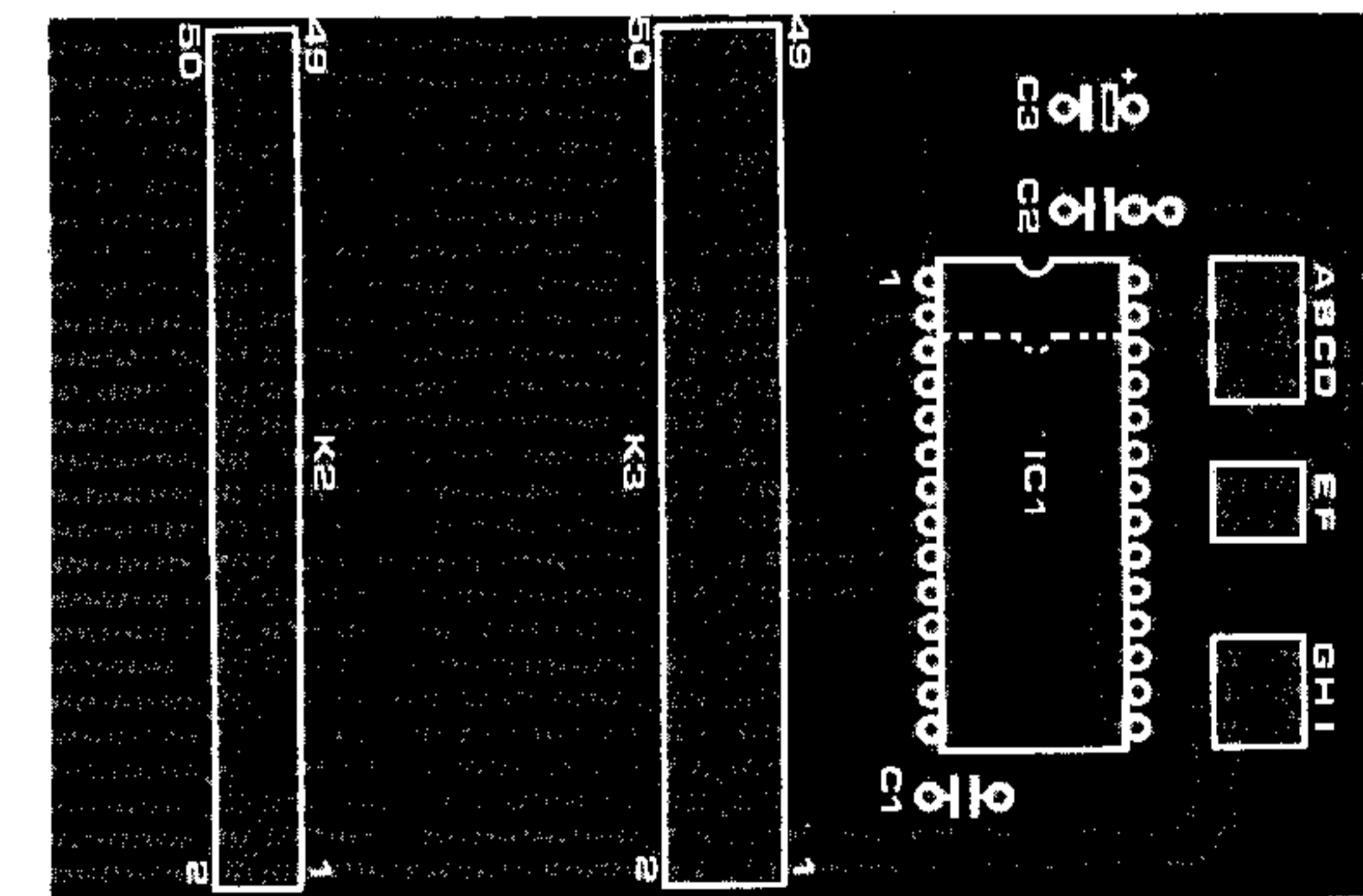
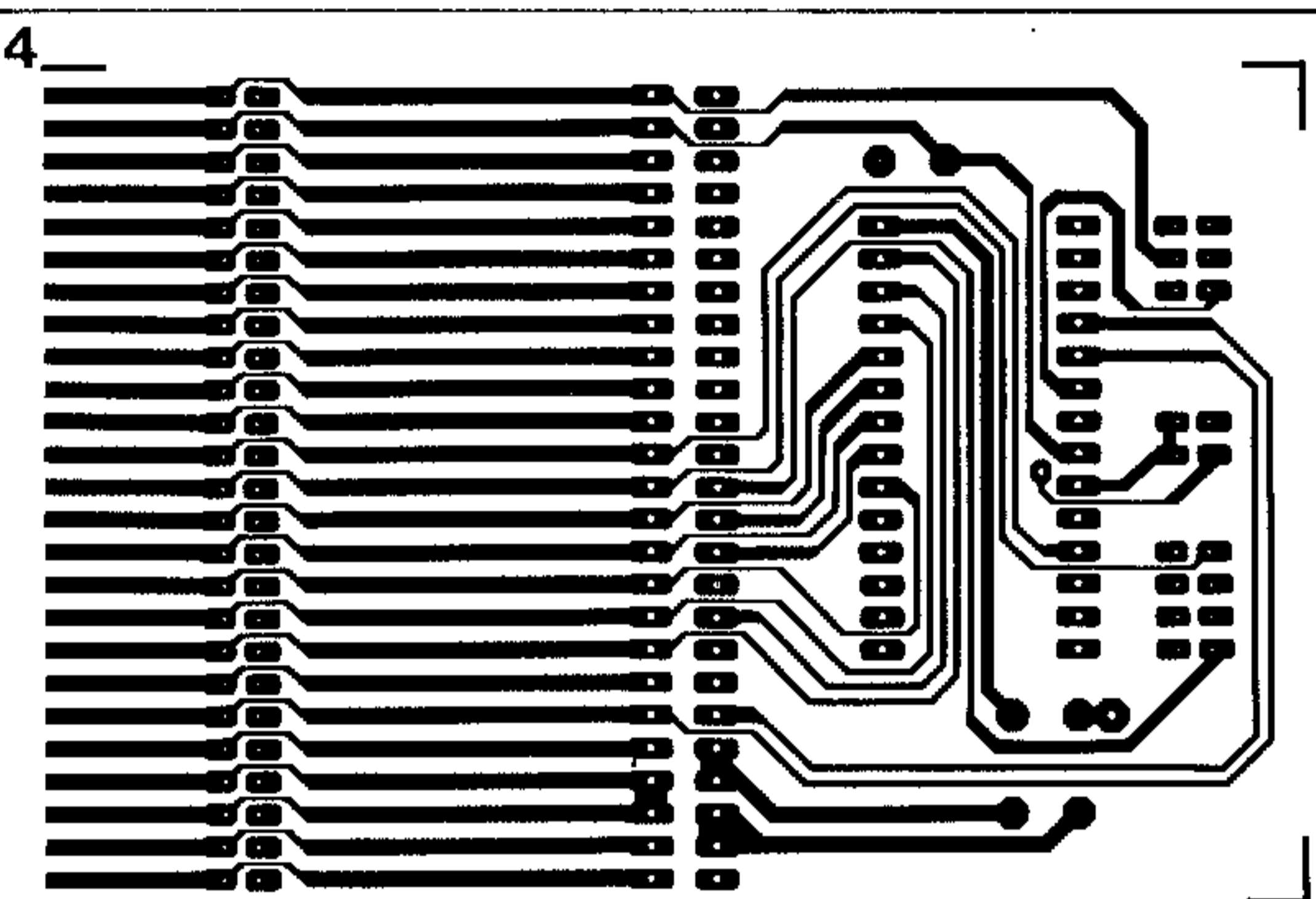


Figure 4. Bien que de petites dimensions, ce circuit imprimé n'en comporte pas moins deux connecteurs d'extension et une "grosse" EPROM.

INIT (Initialisation): ces deux octets contiennent l'adresse d'une routine d'initialisation spécifique pour la cartouche. Si cette dernière n'est pas nécessaire, ces deux octets sont à 0.

STATEMENT: ces deux octets contiennent l'adresse d'un sous-programme plus étendu de traitement d'instructions présent dans la cartouche si tant est que celui-ci soit nécessaire (sinon, ces octets sont à 0). Pour de plus amples informations, nous vous renvoyons au manuel d'utilisation de votre propre MSX.

DEVICE: ce double octet contient l'adresse d'un sous-programme étendu de traitement d'organe matériel (device) présent dans la cartouche. La valeur par défaut est 0. Pour de plus amples informations, nous vous renvoyons au manuel d'utilisation de votre ordinateur MSX.

TEXT: ces deux octets contiennent l'adresse de début du texte BASIC (en caractères ASCII ou pointeurs d'instructions (token)) dans la cartouche. En BASIC, les instructions sont représentées par le numéro d'ordre de cette instruction dans une table de consultation. Dans une seconde table de consultation, se trouve au même numéro d'ordre l'adresse de début de la routine correspondant à l'exécution de l'instruction en question. S'il ne s'agit pas d'un programme BASIC, ces octets sont à 0. Une belle occasion pour les programmeurs en BASIC de stocker leurs programmes spécifiques en EPROM. Nous reviendrons ultérieurement à ce point particulier.

Toutes les adresses mentionnées précédemment doivent se trouver dans l'EPROM, l'octet de poids faible en premier, suivi de l'octet de poids fort, technique commune en langage machine qui n'aurait pas de quoi surprendre un programmeur tant soit peu expérimenté.

Le schéma

Un coup d'oeil à la figure 2 montre que le choix du terme de schéma est quelque peu exagéré. Il serait plus juste de parler de support pour circuit intégré ennoblé, adapté au standard du connecteur pour cartouche MSX, avec tous les avantages que cela sous-entend. IC1 constitue la variable la plus importante de ce montage. Les XXX associés au 27 peuvent prendre toute valeur "informatique" comprise entre 16 et 256, correspondant à 2, 4, 8, 16, ou 32 Koctets. Pour vous permettre de différencier un type d'EPROM de l'autre, nous indiquons leurs brochages respectifs en figure 3. Le dessin du circuit imprimé est conçu de manière à ce qu'il soit possible

d'implanter indifféremment n'importe laquelle des EPROM indiquées plus haut, (implantation allant de pair avec la mise en place d'un ou de plusieurs cavaliers de jonction), cavaliers dont nous allons maintenant expliquer la fonction et l'usage.

Cavalier A: permet de faire la différence entre une 27526 et une 27128. Le décodage de la 27526 nécessite la présence de la ligne d'adresse A14, d'où l'implantation du cavalier A.

Cavalier B: permet d'appliquer le + 5 V à la broche 27 d'une 27128. La sélection 27256 ou 27128 implique la mise en place du cavalier A ou B respectivement.

Cavalier C: permet la mise au + 5 V de la broche 24 (connexion V_{cc}) dans le cas d'une EPROM à 24 broches (2716 ou 2732).

Tableau 1.

	A	B	C	D	E	F	G	H	I
27256	○—○			○—○	○—○		○—○		
27128		○—○		○—○	○—○			○*	○*
2764		○—○	○—○		○—○			○*	○*
2732			○—○		○—○			○*	○*
2716			○—○			○—○		○*	○*

○—○ - interconnexion
○* - au choix (ou H ou I)

85130-T1

Tableau 1. Récapitulation des cavaliers à implanter pour chaque type d'EPROM de la série 27XX et 27XXX. Le choix entre le cavalier H et I est fonction de la zone mémoire adoptée (voir texte).

Tableau 2.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
8000	0	7 L8007	80	A	0	9F Tk	0	16 L8016	80	14	0	85 Tk	22	87 s	74 t	61 a	
8010	72 r	74 t	22	3B ;	41 A	0 EOL	23 L8023	80	1E	0	85 Tk	22	65 e	6E n	64 d	22	
8020	3B ;	42 B	0 EOL	33 L8033	80	28	0	82 Tk	20	43 sp	20 C	20 sp	EF Tk	20 sp	41 A	20 sp	D9 Tk
8030	20 sp	42 B	0 EOL	4E L804E	80	32	0	9D Tk	E4 Tk	22	5C "	20 sp	20 sp	5C "	22	3B ;	
8040	FF Tk	9B Tk	28 (43 C	29)	3B ;	3A :	9D Tk	22 "	20 sp	20 sp	22 "	3B ;	0 EOL	5D L8050	80	
8050	3C *60	0	82 Tk	20 sp	44 D	EF Tk	11 0	20 sp	D9 Tk	20 sp	F Tk	F 15	0 EOL	7B L807B	80	46 *	
8060	0 70	9D Tk	E4 Tk	22 "	5C "	5C "	22 "	3B ;	FF Tk	9B Tk	28 (FF Tk	97 Tk	28 (43 C	F1 Tk	
8070	44 D	29)	29)	3B ;	3A :	9D Tk	22 "	20 sp	22 "	3B ;	0 EOL	81 L8081	80	50 *80	0	83 Tk	
8080	0 EOL	96 L8096	80	5A	0	43 C	EF Tk	43 C	F1 Tk	F Tk	F 15	3A :	9D Tk	22 "	20 sp	22	
8090	3A :	9D Tk	22 "	20 sp	22 "	0 EOL	9C L809C	80	64	0	83 Tk	0 EOL	A2 L80A2	80	6E	0	
80A0	81 Tk	0 EOL	0	0	8 A	41 0	0	C5	32	76	80	0	0	0	0	8	
80B0	42 B	0	C5	32	51	20	0	0	0	0	8	43 C	0	C5	32	59	
80C0	60	0	0	0	0	8	44 D	0	41	90	0	0	0	0	0	0	
80D0	3A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
80E0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
80F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

* : line number *nn*
 L : Link address *llhh*
 Tk : Token byte
 sp : space
 EOL : end of BASIC line
 / : end of program

Tableau 2. Vidage hexadécimal (et traduction) du programme DUMP tel qu'il réside dans la RAM d'un ordinateur MSX. Tous les octets ont été analysés, ce qui simplifie notablement la reconstitution du listing 1.

Cavalier D: relie la broche 26 à la ligne d'adresse A13 (pour une EPROM à 28 broches). Il faut mettre ce cavalier en place avec une 27256 ou une 27128. Dans le cas de la 2764, la broche 26 doit être mise à la tension positive: on implantera donc le cavalier C, (il **ne faut pas** implanter simultanément les cavaliers C et D).

Cavalier E: relie la broche 28 (d'un boîtier à 28 broches) et la broche 21 (dans le cas de la 2732) à la ligne d'adresse A11; ce cavalier doit être implanté avec tous les types d'EPROM évoqués, à l'exception de la 2716.

Cavalier F: par son intermédiaire, la broche V_{pp} d'une 2716 est mise à la tension d'alimentation positive.

Il reste à parler des cavaliers **G, H** et **I**. Par leur intermédiaire, la ligne de validation de l'EPROM en question, \overline{CS} peut être reliée soit à $\overline{CS1}$, soit à $\overline{CS2}$, soit à $\overline{CS12}$. $\overline{CS1}$ est le signal de sélection de la ROM pour l'espace mémoire, (les adresses donc), compris entre 4000 et 7FFF; $\overline{CS2}$ correspond à l'espace mémoire compris entre 8000 et BFFF, et $\overline{CS12}$ correspond à ces deux domaines simultanément (mémoire comprise entre 4000 et BFFF). On implantera soit $\overline{CS1}$ soit $\overline{CS2}$ pour des EPROM de numéro inférieur ou égal à 27128; pour une 27256, il faudra implanter $\overline{CS12}$. Le **tableau 1** récapitule tout ce qu'il vous faut savoir quant à l'implantation des cavaliers A...I. Vous disposez maintenant d'un support pour EPROM réellement universel.

Jusqu'à présent, nous n'avons mentionné que les EPROM. Il va sans dire qu'il est également possible de mettre dans ce support des PROM ou des ROM (à condition qu'il s'agisse d'une version compatible broche à broche avec l'un des types d'EPROM mentionnés ci-dessus).

Les lignes SW1 et SW2 du connecteur encartable à 50 broches sont interconnectés et servent de dispositif de détection pour la protection lors de l'implantation ou de l'extraction d'une cartouche: cette protection garantit une sécurité totale lors de la manipulation des cartouches (si tant est que l'on utilise une telle extension).

La platine est en outre dotée d'un certain nombre de connecteurs dont nous allons expliciter les fonctions. Nous avons baptisé K1 le connecteur mâle enfichable que constituent les pistes sur les deux côtés à l'extrémité du circuit imprimé; il peut être enfiché directement dans le connecteur destiné à recevoir une cartouche MSX. Le connecteur K2, connecteur mâle, est réalisé à l'aide d'une double barette Berg de 25 x 2 broches au pas de 2,54 mm. K3 est un

connecteur femelle encartable standard pour cartouche. Ce dernier connecteur doit être du même type que celui utilisé pour recevoir une cartouche ordinaire, c'est-à-dire au pas de 0,1 inch, soit 2,54 mm. Il est important de vérifier que l'on n'achète pas un connecteur au pas "européen" de 2,50 mm. La différence entre les deux est trop importante pour permettre de les utiliser indifféremment l'un à la place de l'autre.

La construction

La **figure 4** montre les dessins des deux côtés de la platine conçue pour l'extension cartouche pour MSX. Il s'agit d'un double face à trous métallisés solution adoptée pour réduire au minimum la taille de la platine. Il faudra préétamer les broches du connecteur enfichable K1 et les îlots de soudure. Comme le circuit n'exige qu'un unique support on en choisira un, non pas de bonne, mais d'excellente qualité, car il n'est pas exclu qu'il soit utilisé très intensivement. On pourrait éventuellement envisager l'utilisation d'un support FIN (à force d'insertion nulle), pour peu que l'on envisage de changer souvent d'EPROM (programmes différents).

Nous ne pensons pas que le circuit puisse poser des problèmes de soudure si tant est que l'on utilise un fer à souder à pointe assez fine (normale quoi!!!).

Mode d'emploi

Vous venez de terminer la réalisation de ce montage et la question que vous vous posez est bien évidemment celle-ci: que puis-je bien en faire? A l'image d'une cartouche ordinaire, cette extension peut recevoir un programme utilisé fréquemment, lorsque l'on ne supporte plus le lent chargement par l'interface cassette. Il peut tout aussi bien s'agir de jeux que de programmes utilitaires, (dévermineur entre autres) que l'on a ainsi à disposition par l'entremise d'une cartouche de fabrication maison. Comment faire pour stocker un programme BASIC dans un tel module?

Nous avons vu plus haut quelle était la forme prise par un programme stocké en EPROM dans une cartouche. S'il s'agit d'un programme en BASIC, deux choses seulement importent: ID, les deux octets des adresses XX00 et XX01 de l'EPROM et TEXT, l'adresse de début du programme placée aux adresses XX08 et XX09. Comme les 16 premiers octets de la cartouche de ROM sont

réservés, on pourra choisir l'adresse XX10 de l'(EP)ROM comme adresse de début du programme. Outre les adresses que nous venons d'indiquer, il suffit d'indiquer l'adresse (obligatoire) de l'EPROM à savoir 8000. A partir de maintenant, nous ne nous intéressons plus qu'aux adresses à compter de 8000 et au-delà.

Avant de pouvoir transférer un programme BASIC en cartouche (EPROM) il va nous falloir voir quel est le principe de stockage utilisé pour ce type de programme. Les choses se passent de la manière suivante:

A l'adresse de début (8010) on trouve invariablement la valeur 00 (deux octets) suivie d'une adresse de lien (link address) (deux octets) et d'un numéro de ligne (de deux octets lui aussi). On trouve ensuite une ligne de code BASIC sous la forme de pointeurs d'instructions (token), voir plus haut, ce qui signifie que les instructions ne sont pas mémorisées en format ASCII, mais sous la forme d'un code octet par octet. La ligne se termine par 00. On trouve ensuite une nouvelle adresse de lien suivie d'un numéro de ligne etc...

Avant de pouvoir mettre un programme en BASIC en EPROM, il va falloir lui faire subir un petit traitement. Il faut commencer par l'imprimer en format hexadécimal. Pour ce faire, il suffit de mettre la routine "DUMP" donnée en **listing 1** à la suite du programme en BASIC, en numérotant ses lignes à compter de 10000. Ainsi, après une instruction "RUN 10000" le programme demande l'adresse de début (start) et l'adresse de fin (end). En ce qui nous concerne, l'adresse de début est toujours 8000, l'adresse de fin dépend bien évidemment de la longueur du programme. Le résultat du traitement de cette routine BASIC par elle-même est donné en **tableau 2**: le listing de la zone mémoire dans laquelle elle réside.

Il nous faut ensuite augmenter de 10 (hex) toutes les adresses de lien. Pour ce faire, on commence par rechercher ces dernières dans l'ensemble du listing produit par "DUMP". Il n'y a pas de quoi s'arracher les cheveux, sachant que chaque adresse de lien pointe vers l'emplacement mémoire dans lequel se trouve l'adresse de lien suivante. Attention, l'adresse de lien et le numéro de ligne sont tous deux mis en mémoire de manière non cartésienne, l'octet de poids faible en premier, suivi de l'octet de poids fort: procédé habituel en langage machine. La fin du programme est indiquée par l'adresse de lien 0000. Comme la routine "DUMP" est accolée au programme à traiter, la fin

Tableau 3.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8000	41	42	0	0	0	0	0	0	10	80	0	0	0	0	0	0
8010	0	17	80	A	0	9F	0	26	80	14	0	85	22	73	74	61
8020	72	74	22	3B	41	0	33	80	1E	0	85	22	65	6E	64	22
8030	3B	42	0	43	80	28	0	82	20	43	20	EF	20	41	20	D9
8040	20	42	0	5E	80	32	0	9D	E4	22	5C	20	20	5C	22	3B
8050	FF	9B	28	43	29	3B	3A	9D	22	20	20	22	3B	0	6D	80
8060	3C	0	82	20	44	EF	11	20	D9	20	F	F	0	8B	80	46
8070	0	9D	E4	22	5C	5C	22	3B	FF	9B	28	FF	97	28	43	F1
8080	44	29	29	3B	3A	9D	22	20	22	3B	0	91	80	50	0	83
8090	0	A6	80	5A	0	43	EF	43	F1	F	F	3A	9D	22	20	22
80A0	3A	9D	22	20	22	0	AC	80	64	0	83	0	B2	80	6E	0
80B0	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

effective du programme est décalée de quelque 160 octets par rapport à la fin du programme à traiter. Il reste à trouver la fin réelle du programme, en recherchant par exemple le numéro de ligne 10000 (2710 hex). L'adresse de lien précédant immédiatement ce numéro sera remplacée par 0000.

Pour bien saisir le principe de fonctionnement de la procédure que nous venons de définir, la meilleure solution consiste à entrer la routine "DUMP" et à exécuter le programme (RUN) en choisissant &H8000 comme adresse de début et &H8100 comme adresse de fin.

Pour bien comprendre le procédé utilisé, il est bon de savoir qu'en BASIC, les instructions sont mises en mémoire sous la forme d'abréviation (gain de place) et que chacune d'entre elles a une valeur hexadécimale réservée. Un peu d'exercice aura vite fait de vous permettre d'en reconnaître quelques-unes telles que 82_{hex} = FOR, 9D_{hex} = LPRINT, EF_{hex} = "=", 83_{hex} = NEXT, F1_{hex} = "+", E4_{hex} = USING, etc. . .

Une étude comparative du résultat avec le tableau 2 expliquera bien des choses restées mystérieuses. Si vous ne disposez pas d'une imprimante, vous pouvez bien évidemment utiliser votre écran pour l'affichage des informations. Il vous faudra alors remplacer toutes les instructions LPRINT par PRINT. Il faudra en outre remplacer dans les lignes 60 et 90 le nombre 15 par le chiffre 7 ceci en raison du nombre maximum de caractères affichables par ligne.

Le tableau 3 donne le contenu hexadécimal à mettre dans l'EPROM. Les 16 premiers octets (ligne du haut) servent au code d'identification et doivent toujours être compris dans la programmation dans le cas présent. A partir de l'adresse 8010 commence le programme BASIC proprement dit. Les adresses de lien ombrées sont exactement celles du tableau 2 augmentées de 10 (hex). Le programme BASIC en ROM ainsi obtenu

démontre automatiquement, mais il est impossible de l'éditer ou d'y ajouter quoi que ce soit. Le problème est que l'on ne travaille pas en RAM!!! Cependant, toutes les variables sont stockées en RAM. Dans le tableau 2, on les trouve immédiatement à la suite du programme BASIC (à compter de 80A4 dans le cas présent). Lorsque l'on met dans le connecteur pour cartouche l'EPROM programmée selon les indications ci-dessus, le domaine de RAM commence à C000. On découvre les variables à partir de C010 (à vérifier à l'aide du programme "DUMP").

Il n'est pas impératif de disposer d'un connecteur pour cartouche pour pouvoir utiliser cette platine. Le Spectraideo est l'ordinateur MSX type non doté d'un connecteur de ce genre. Il comporte un connecteur de ports d'E/S répondant aux mêmes normes, de sorte qu'il suffit d'un petit morceau de câble plat à 50 brins de quelques centimètres de long doté à ses deux extrémités d'un connecteur femelle pour câble en nappe aux normes HE 10 de 2 x 25 broches. La photo de la figure 5 montre l'aspect d'une telle combi-

naison. Attention: le connecteur de sortie du Spectraideo possède une numérotation correspondant à celle du connecteur pour cartouche d'un autre ordinateur MSX, mais en réalité, il est câblé en miroir par rapport à ce dernier. Ceci explique pourquoi, la petite flèche indiquant la broche 1 sur le connecteur du Spectraideo ne correspond pas à la broche 1 mais à la broche 50. Un homme averti en vaut deux. Si vous respectez l'exemple donné en figure 5, vous ne devriez pas rencontrer le moindre problème.

On peut également utiliser ce circuit imprimé comme connecteur adaptateur entre un ordinateur MSX et un synthétiseur Yamaha. Il faut dans ce cas de scier le circuit imprimé entre les connecteurs K2 et K3 de manière à ne conserver qu'un couplage mâle - mâle.

Nous en avons terminé avec le circuit pour cartouche. Dans l'épisode suivant nous nous intéresserons à une platine de bus pour ordinateurs MSX.

A d'ici là!

Tableau 3. Voici les données à mettre dans l'EPROM pour disposer d'une cartouche utilisable que nous avons baptisée DUMP. A noter que les adresses ombrées (celles des adresses de lien et celles des emplacements mémoire) sont celles du tableau 2 augmentées de 10_{hex}.

Figure 5. L'adjonction d'un morceau de câble en nappe de 50 brins doté de deux connecteurs femelles permet aussi de connecter ce circuit au bus de sortie d'un Spectraideo.

