

10/1.3

Amstrad et le jeu en binôme

Heureux possesseur d'un Amstrad-CPC depuis quelque temps, vous commencez à regretter que ce dernier manque d'originalité dans les ouvertures de votre jeu d'échecs favori.

Pas étonnant ! Depuis le temps que vous le défiez, vous les connaissez toutes par cœur.

De même, avec votre programme d'Othello, vous connaissez toutes les astuces pour obliger monsieur CPC à s'incliner.

Ce qu'il vous manque, c'est un adversaire à la hauteur, qui possède une vision globale du jeu, pour vous astreindre à une réflexion différente à chaque partie. Un humain donc.

Justement, votre cousin, aussi à l'aise que vous contre le programme de Dames françaises, recherche aussi un adversaire à sa taille.

Première pensée : jouer tous les deux devant l'ordinateur (en version deux adversaires, l'ordinateur devenant le support du jeu). Mais cette solution s'avère vite assez peu pratique, chacun devant céder sa place entre chaque coup, ce qui perturbe la réflexion, et accumule le temps perdu.

Deuxième solution, sortir le jeu grandeur nature, relégué au fond d'un placard poussiéreux au grenier, depuis l'entrée de l'ordinateur au foyer. Hélas, après vérification, vous vous apercevez qu'une partie de pions a été irrémédiablement perdue entre les lattes du plancher par le petit dernier.

Deux claviers sur un même Amstrad ? N'y pensons pas non plus, la réalisation est trop complexe.

Deux Amstrads connectés ensemble ? Une solution onéreuse, un ordinateur supplémentaire dépasse votre budget. Et votre cousin n'est pas prêt à déplacer son micro-ordinateur à chacune de ses visites. Vous non plus d'ailleurs, des transports répétés ne sont pas sans risques pour la mécanique des lecteurs.

« Que faire donc ? »

Votre ouvrage *Comment exploiter les ressources et augmenter les performances de votre Amstrad* vous souffle la réponse : « Utilisez votre MINI-

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

TEL, mis gratuitement à votre disposition par l'administration des PTT, si vous possédez une ligne téléphonique ».

« Oui, mais comment ? Et à moindres frais, s'il vous plaît ».

Pas de problème, pour une trentaine de francs, quelques minutes de réalisation, et de la minutie, en suivant nos conseils, vous aurez le matériel nécessaire à la connexion de votre Amstrad et d'un MINITEL.

Nous vous proposerons aussi des réalisations logicielles pour exploiter cette liaison simple, notamment dans le domaine du jeu à deux partenaires.

I. Une connexion simple de l'Amstrad au MINITEL

Nous avons déjà effectué la liaison d'un Amstrad-CPC avec un MINITEL, mais celle-ci demandait l'acquisition de l'interface RS-232C AMSTRAD (quelques centaines de francs supplémentaires), l'achat de quelques composants et la disponibilité d'un matériel de gravure, pour réaliser un petit circuit imprimé.

LE MATÉRIEL

Pour la réalisation que nous vous proposons, bien que plus modeste dans ses performances que la liaison RS-232C — mais suffisante pour notre cas — nous avons uniquement besoin de :

- soit un connecteur plat pour bord de carte, au format 2 × 17 broches femelle ;
- soit un connecteur Centronic (plus onéreux) 36 broches femelle ;
- soit une fiche DIN femelle 5 broches à fixer et une fiche DIN mâle supplémentaire. Mais cette réalisation demandera plus de temps, car il faudra usiner à la scie et à la lime le boîtier de votre unité centrale ;
- une fiche DIN mâle 5 broches ;
- entre un mètre et un mètre cinquante de fils de petite section, en trois couleurs (nous vous proposons dans la suite : marron, rouge et orange, mais libre à vous de modifier les couleurs, ne l'oubliez pas en câblant).

LA FICHE PÉRI-INFORMATIQUE DU MINITEL

La liaison que nous allons effectuer utilise la fiche péri-informatique du MINITEL (fiche DIN femelle à cinq broches qui se trouve à l'arrière). Aussi pas de risque de voir sa facture téléphonique augmenter, car nous ne passerons pas par la ligne de téléphone.

Les broches de cette fiche, décrite en Partie 10 chapitre 1.1, nous procureront deux signaux intéressants et une ligne de masse :

- RX : qui transporte les données utilisées par le MINITEL ;
- TX : est la ligne sur laquelle transitent les signaux émis par le MINITEL.

La ligne RX est composée d'une entrée sur un composant logique à seuil de type TTL avec des protections appropriées contre les surtensions.

La tension d'utilisation recommandée est de 5 Volts.

Les composants internes de votre Amstrad fonctionnent sous cette tension, et ce ne sont pas les possibilités de connexions qui manquent, sur les connecteurs arrière du boîtier.

La ligne TX nous posera par contre quelques problèmes.

Cette liaison est assurée par l'intermédiaire d'un composant logique en technologie TTL, donc une utilisation sous 5 Volts possible, mais elle est de type collecteur ouvert. Son modèle se présente selon la représentation de la figure 1.

Les signaux transitant, ayant, soit une valeur de zéro volt, pour un niveau logique « 0 » ; soit 5 volts, pour un niveau logique « 1 », deux cas se présentent :

— lorsque la sortie doit laisser apparaître un niveau logique zéro, celle-ci est virtuellement connectée à la masse ;

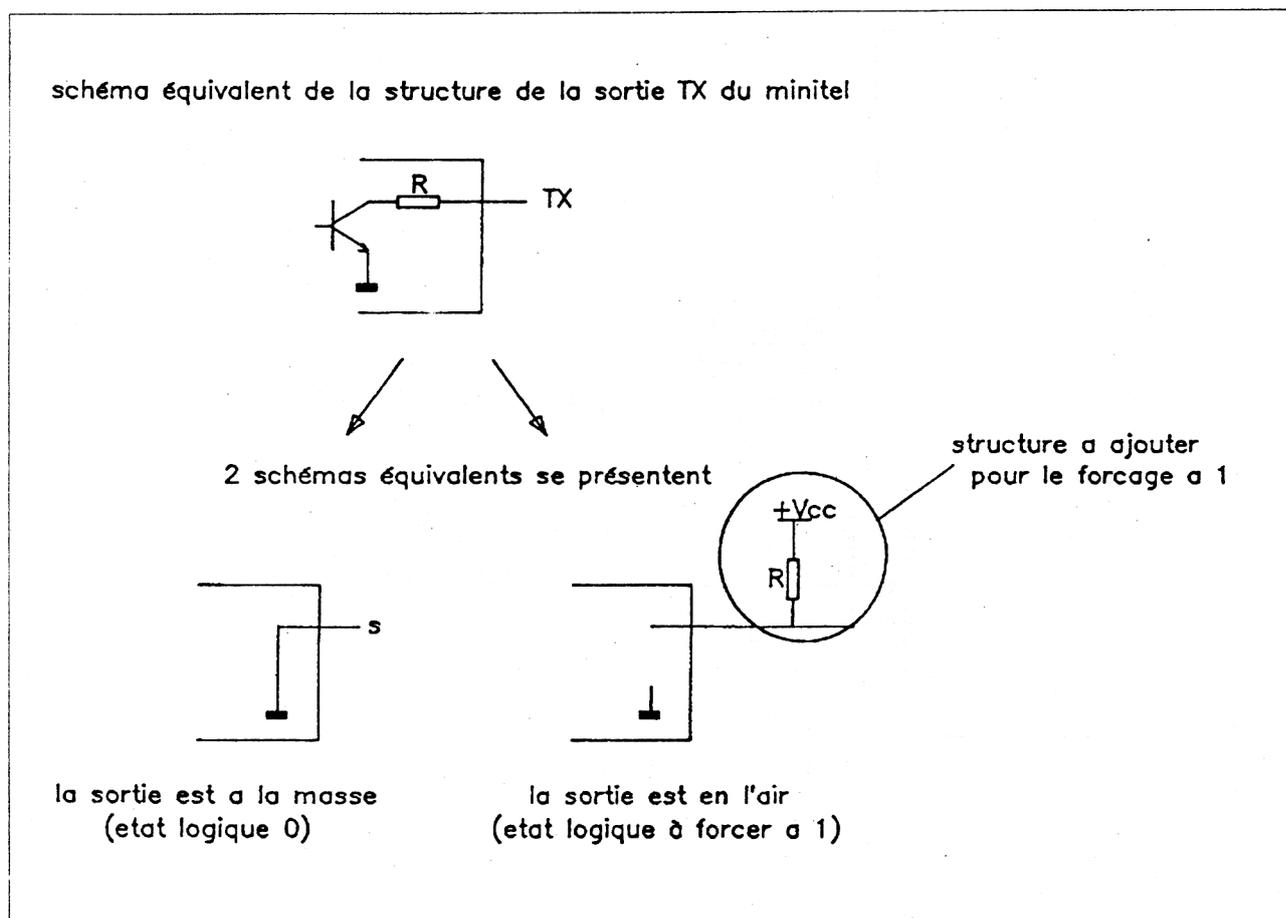


Fig. 1

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

— lorsque la sortie doit laisser apparaître un niveau logique un, elle est virtuellement déconnectée, et laisse le point s en l'air. Or, en électronique, un point en l'air ne présente aucune tension valable (hormis tous les parasites que l'on peut mesurer). Il est donc nécessaire de forcer cette sortie au niveau logique un, en la rappelant à la tension d'alimentation +Vcc, par l'intermédiaire d'un composant résistif de valeur quelques kilohms. Le terme utilisé pour ce composant est : résistance de PULL-UP.

Dans un premier temps, ce dernier cas nous a posé problème, car il nous oblige à ajouter un composant à notre montage, et à récupérer la tension +Vcc sur un des connecteurs, donc à ajouter un fil supplémentaire.

RÉSOLUTION DU PROBLÈME SUR AMSTRAD

Heureusement, nous vous avons fourni les schémas des cartes mères des AMSTRAD-CPC (Partie 2, Chapitre 2), aussi, après une étude approfondie, il nous a suffit de repérer une structure intéressante autour du PIO 8255, que nous avons représenté en figure 2.

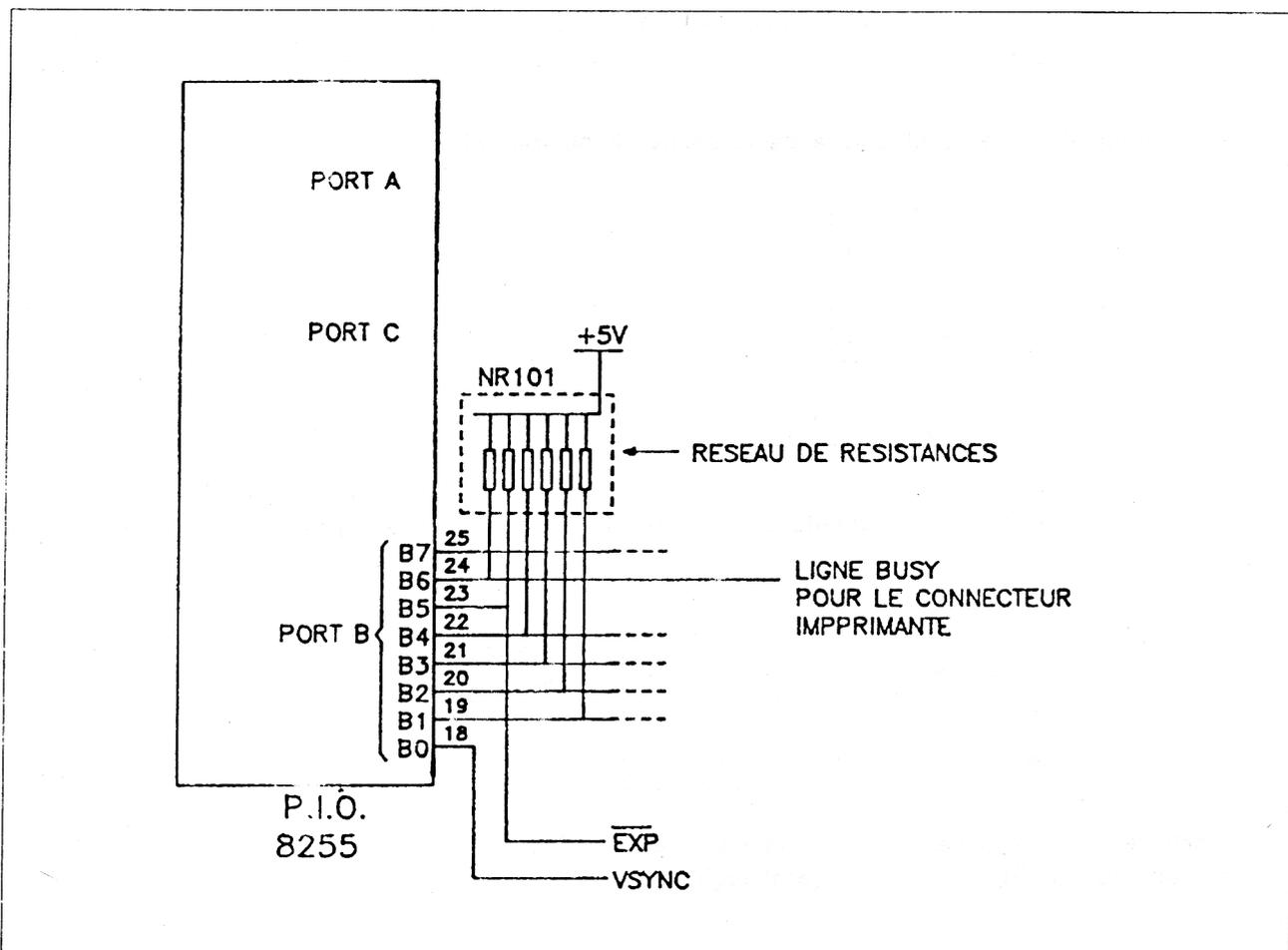


Fig. 2 : Les connexions autour du PIO 8255.

Vous pouvez noter que les lignes B1, B2, B3, B4, B5, B6, du port B sont, chacune, connectée à un composant résistif. Les six composants sont regroupés, au niveau de la représentation, dans un rectangle en pointillé nommé NR101. Il s'agit en fait de ce que l'on appelle un réseau de résistances : celui-ci contient 6 résistances, qui ont toutes un point commun, et ici relié au +5 Volts.

Ainsi, nous pourrions utiliser une de ces sorties pour la connecter à la ligne TX du minitel.

B1, B2, B3, B4 sont utilisées pour l'initialisation de l'AMSTRAD, et, selon les modèles, certaines de ces broches sont connectées à la masse. De plus, aucune n'apparaît sur un des connecteurs à l'arrière de l'ordinateur.

B5 est réservée à la détection d'un éventuel périphérique, aussi, nous n'y toucherons pas.

Il nous reste B6, qui est connectée à la ligne $\overline{\text{BUSY}}$ de l'interface imprimante. Après réflexion, nous avons jugé intéressant d'utiliser cette ligne, il suffira de déconnecter l'imprimante lors de l'utilisation du minitel.

Cette possibilité nous permettra aussi d'utiliser une ligne de sortie des données, D0 par exemple, que nous connecterons à l'entrée RX du minitel.

La possibilité d'utiliser la masse présente sur plusieurs broches de ce connecteur est, de plus, fort passionnante.

LA RÉALISATION LA PLUS ÉCONOMIQUE

Cette première réalisation est la plus économique, et s'adresse aux possesseurs d'un AMSTRAD-CPC dont les connecteurs arrières sont réduits à leur plus simple expression : un bord de carte du circuit imprimé.

Voici donc la liste du matériel nécessaire pour cette réalisation :

- un connecteur femelle encartable sur un bord de carte, deux fois 17 connexions ;
- une fiche DIN mâle 5 broches à souder en l'air ;
- trois fils de couleurs (rouge - orange - marron), de préférence solidaires, et d'une longueur de 1 mètre ;
- un fer à souder 20 à 30 watts ;
- un peu de soudure à l'étain.

Après avoir consulté les chapitres 7 et 8 de la Partie 10, rappelant les principes de base des connecteurs, et le soudage, vous pourrez souder les fils sur le connecteur et la fiche DIN selon le schéma de câblage de la figure 3.

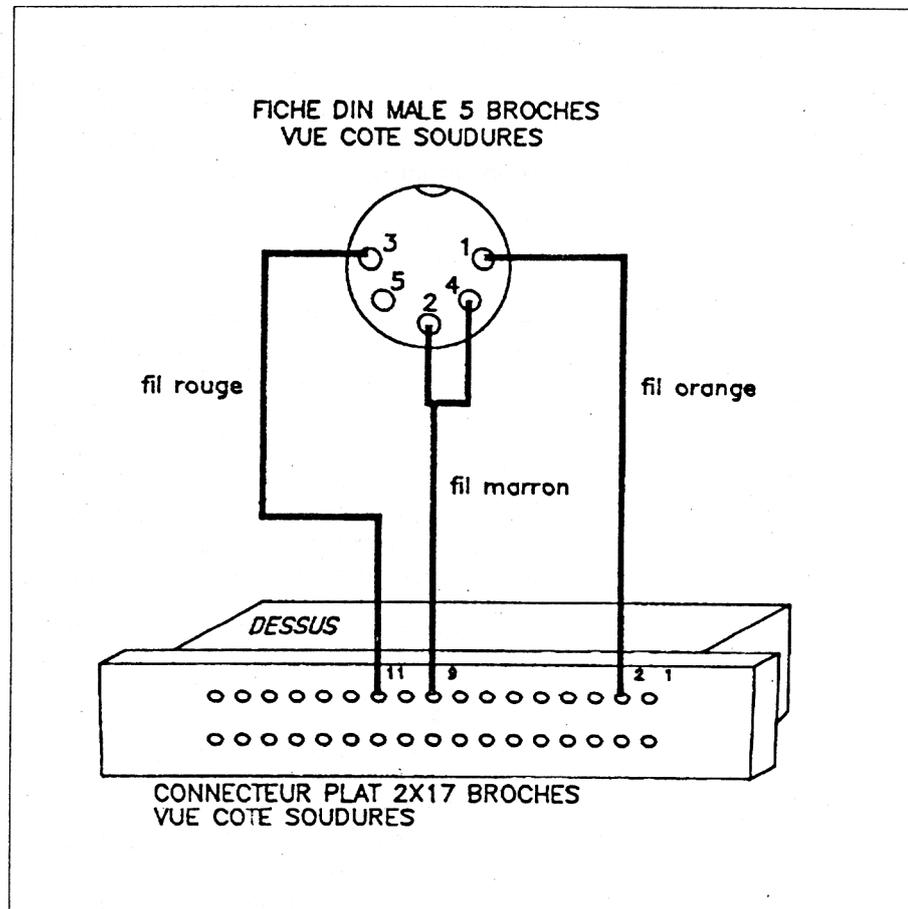


Fig. 3 : Câblage connecteur plat ↔ fiche DIN.

La broche 2 du connecteur, supportant la liaison D0 du port imprimante, est à connecter, par l'intermédiaire du fil orange, à la ligne RX, en broche 1 de la fiche minitel.

Le fil rouge relie la broche 11 (ligne $\overline{\text{BUSY}}$ du port imprimante).

Nous avons utilisé la masse présente en broche 9 du connecteur, que nous avons reliée aux broches 2 et 4 de la fiche DIN.

Les fils correctement soudés, vous pouvez remonter la fiche DIN.

Vous vous assurerez aussi de la qualité des soudures sur le connecteur, et vous pourrez éventuellement isoler les liaisons à l'aide d'un ruban auto-collant.

Repérer tout de suite, par une étiquette, le dessus du connecteur afin de ne pas inverser sa connexion, ce qui risquerait d'être fatal tant pour votre AMSTRAD que pour votre minitel.

Vous pouvez maintenant procéder aux branchements, et étudier les logiciels proposés dans la suite de notre étude.

DEUXIÈME RÉALISATION PRATIQUE

D'un coût légèrement supérieur, cette deuxième réalisation a le mérite de ne pas abîmer le bord de carte, surtout si vous êtes obligé de connecter très souvent votre imprimante ou le MINITEL.

Cette réalisation est surtout valable pour les modèles de CPC possédant un connecteur imprimante mâle de type Centronic, mais vous pourrez aussi l'utiliser sur le câble plat reliant votre ordinateur à l'imprimante.

Le matériel nécessaire est identique au précédent montage, mais il vous faudra remplacer le connecteur pour bord de carte par un connecteur Centronic deux fois 36 broches femelle.

La figure 4 représente les différentes connexions à effectuer.

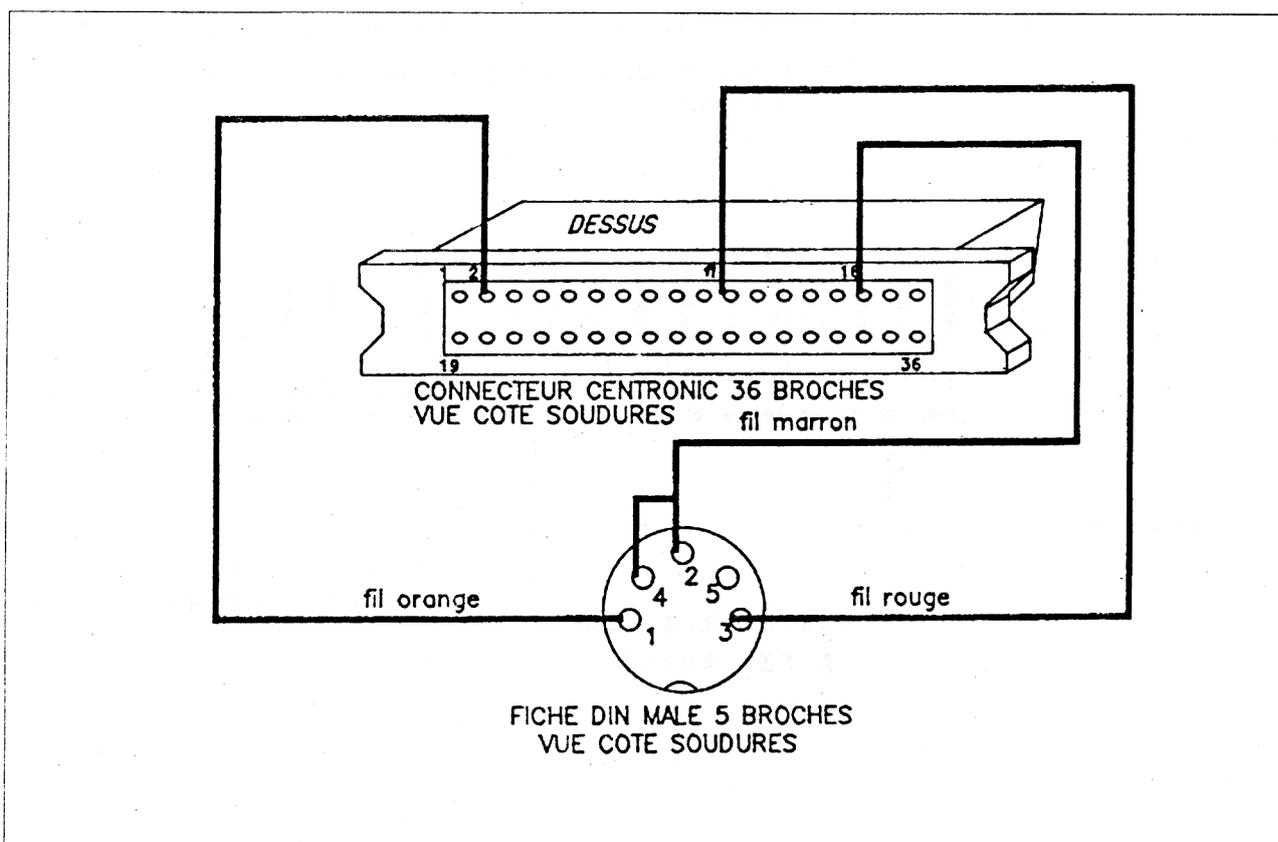


Fig. 4 : Câblage fiche Centronic ↔ fiche DIN.

Les broches 2 (D0) et 11 ($\overline{\text{BUSY}}$) sont à connecter respectivement aux broches 1 (RX) et 3 (TX) par l'intermédiaire des fils orange et rouge.

La masse a cette fois-ci été reprise sur la broche 16 du connecteur Centronic, pour être reliée, via le fil marron, aux broches 2 et 4 de la fiche DIN.

UNE RÉALISATION PLUS FIABLE

Afin d'éliminer l'utilisation du connecteur deux fois 17 broches, source d'erreur de connexion, comme indiqué plus haut (dieu sait si les informaticiens sont distraits !), nous avons réalisé une connexion beaucoup plus pratique, éliminant tout risque d'erreur lors de l'utilisation.

Cette réalisation vous demandera un peu de sang froid, et beaucoup de minutie, car nous allons ouvrir le ventre de votre AMSTRAD pour y souder directement les fils de liaisons.

Le matériel nécessaire sera :

- une fiche DIN femelle 5 broches à fixer sur châssis ;
- deux vis et écrous de fixation pour la fiche DIN femelle ;
- deux fiches DIN mâles 5 broches volantes ;
- trois fils de couleurs (comme précédemment), un fer à souder et de la soudure à l'étain.

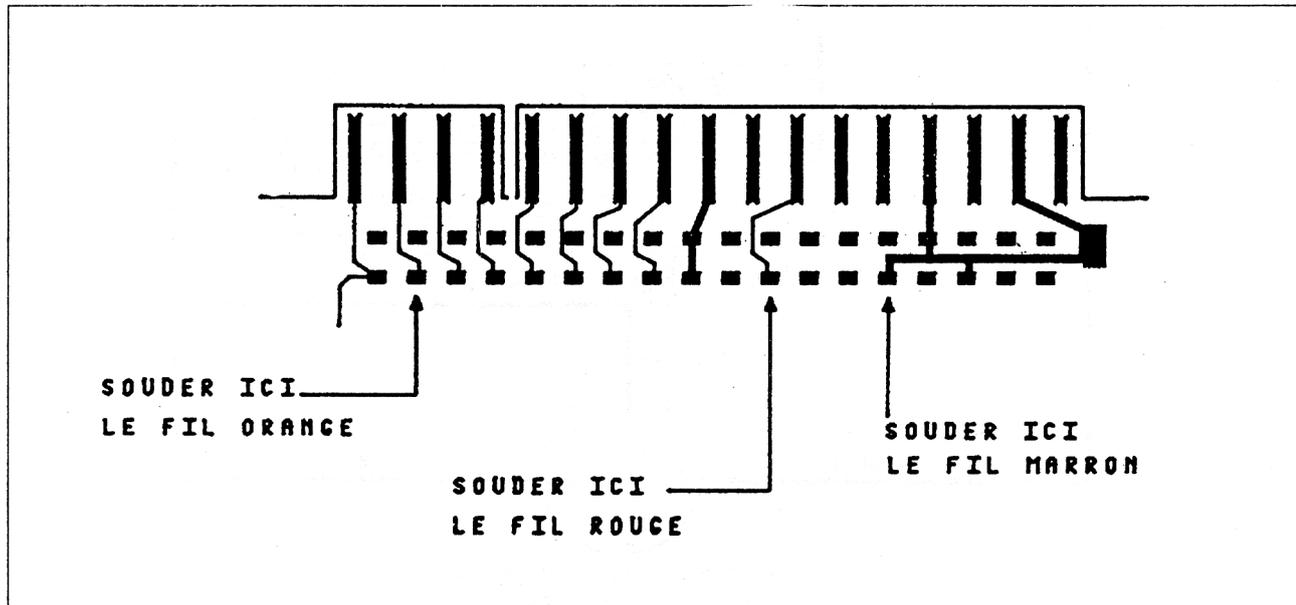


Fig. 5

Comme représenté figure 5, trois soudures seront à effectuer sur des pastilles de circuit-imprimé, visibles près du connecteur bord de carte de l'interface imprimante.

Vous essaierez de ne pas trop chauffer les pastilles pour éviter leur décollement de la carte mère.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Chacun des fils sera ensuite relié sur les broches de la fiche DIN femelle : le fil orange sur la broche 1, le fil rouge sur la broche 3, et le fil marron sur les broches 2 et 4.

Repérez un endroit pour fixer la fiche DIN (entre le connecteur imprimante et le connecteur d'extension, par exemple).

Ajoutez suffisamment pour permettre à la fiche de se positionner.

Percez deux trous pour les vis de fixations, et fixez la fiche, après avoir éliminé les éventuelles bavures.

Il est temps maintenant de réaliser le cordon à l'aide de fils et des deux fiches DIN mâles (les fils doivent être soudés en vis-à-vis sur les fiches), ou d'utiliser un cordon du commerce, où les liaisons ne sont pas croisées.

Vous pourrez ensuite connecter AMSTRAD et MINITEL comme indiqué en figure 6.

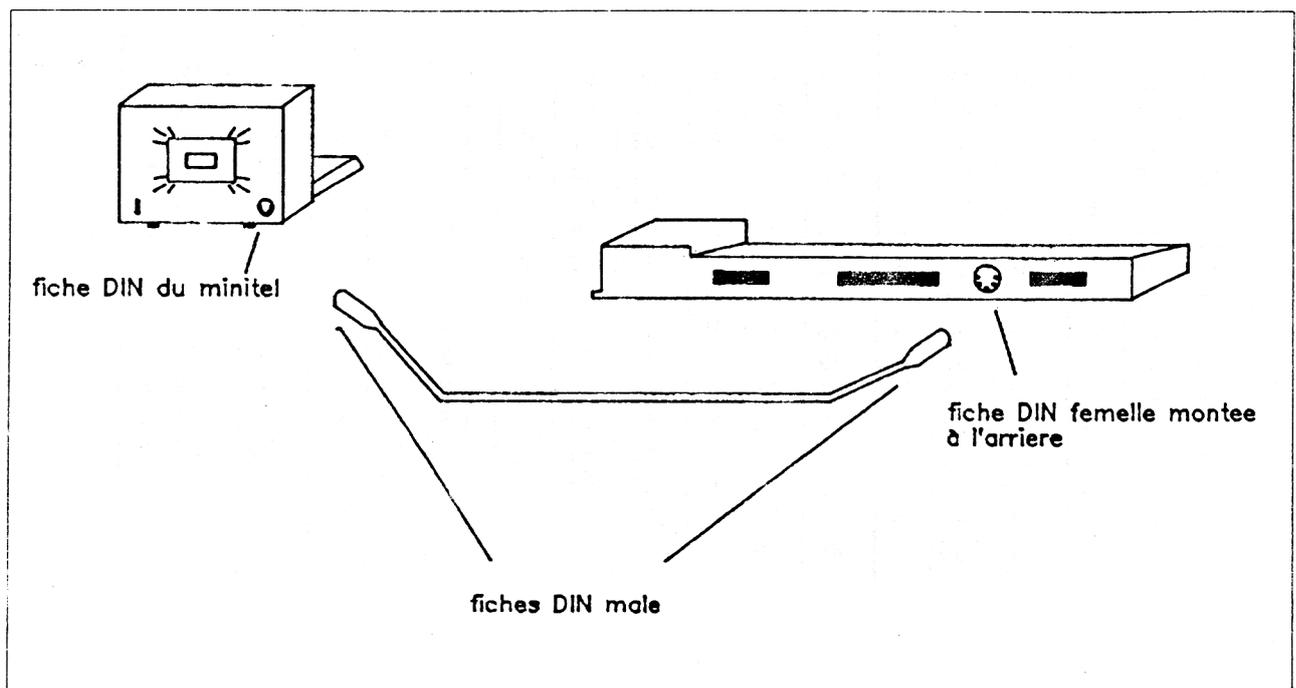


Fig. 6

II. Le logiciel d'exploitation

Une fois connectés ensemble, le MINITEL et votre ordinateur ne pourront pas dialoguer sans un logiciel adéquat, gérant cette liaison, car le format des données sur le MINITEL est différent de celui de votre AMSTRAD.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

LE FORMAT DES DONNÉES

Votre AMSTRAD travaille sous un format de données 8 bits, et ce en parallèle, c'est-à-dire que les 8 bits sont communiqués aux différents composants simultanément.

Par contre, le MINITEL communique avec le matériel extérieur, par l'intermédiaire de la prise périinformatique, sous un format 7 bits, et ce en série. C'est-à-dire que les bits sont envoyés l'un à la suite de l'autre sur un même fil.

De plus, les sept bits envoyés doivent être précédés d'un bit de start, et suivis d'un bit de parité paire, et d'un bit de stop signalant la fin du caractère.

20	32		30	48	0	40	64	⊕	50	80	P	60	96	⊖	70	112	p
21	33	!	31	49	1	41	65	A	51	81	Q	61	97	a	71	113	q
22	34	"	32	50	2	42	66	B	52	82	R	62	98	b	72	114	r
23	35	#	33	51	3	43	67	C	53	83	S	63	99	c	73	115	s
24	36	\$	34	52	4	44	68	D	54	84	T	64	100	d	74	116	t
25	37	%	35	53	5	45	69	E	55	85	U	65	101	e	75	117	u
26	38	&	36	54	6	46	70	F	56	86	V	66	102	f	76	118	v
27	39	'	37	55	7	47	71	G	57	87	W	67	103	g	77	119	w
28	40	(38	56	8	48	72	H	58	88	X	68	104	h	78	120	x
29	41)	39	57	9	49	73	I	59	89	Y	69	105	i	79	121	y
2A	42	*	3A	58	:	4A	74	J	5A	90	Z	6A	106	j	7A	122	z
2B	43	+	3B	59	:	4B	75	K	5B	91	[6B	107	k	7B	123	□
2C	44	,	3C	60	<	4C	76	L	5C	92	\	6C	108	l	7C	124	▣
2D	45	-	3D	61	=	4D	77	M	5D	93]	6D	109	m	7D	125	□
2E	46	.	3E	62	>	4E	78	N	5E	94	^	6E	110	n	7E	126	□
2F	47	/	3F	63	?	4F	79	O	5F	95	□	6F	111	o	7F	127	■

Fig. 7

La figure 7 représente la forme générale des informations transitant sur une ligne série communiquant avec un MINITEL.

L'état de repos est caractérisé par une tension de + 5 Volts présente sur la ligne.

Un bit de start à zéro Volt est envoyé, puis arrive le bit de poids le plus faible D0 du caractère. D1, D2, D3, D4, D5 et D6 suivent, puis P le bit de parité paire, qui est constitué d'un bit positionné à + 5 Volts si le nombre

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

de bits à +5 Volts du caractère est impair (le nombre total de bits à +5 Volts sera ainsi pair), ou positionné à zéro Volts si ce nombre est déjà pair. Un bit de stop placé à +5 Volts arrive enfin.

Si un deuxième caractère arrive, comme représenté sur la figure 7, la forme de la communication se reproduit : bit de start à zéro, D0, D1, ..., D6, bit de parité, et bit de stop à +5 Volts.

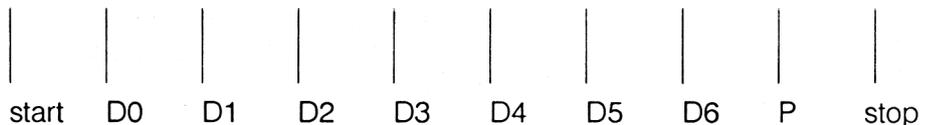
Si aucun caractère ne suit, la ligne reste au +5 Volts, état de repos.

Prenons pour exemple l'envoi du caractère C : ce caractère est représenté par l'octet &43, qui correspond au nombre binaire D7 D6 D5 D4 D3 D2 D1 D0 = 0 1 0 0 0 0 1 1. Comme seulement 7 bits sont utilisés, nous aurons D6 D5 D4 D3 D2 D1 D0 = 1 0 0 0 0 1 1.

Après le bit de start, seront donc envoyés les bits : 1 (D0), 1 (D1), 0, 0, 0, 0, et 1 (D6). Le bit de parité égal à 1 suivra, puis le bit de stop.

Les niveaux de tensions seront donc, dans l'ordre :

0 V - 5 V - 5 V - 0 V - 0 V - 0 V - 0 V - 5 V - 5 V - 5 V



En plus de ce format, chacun des bits doit être envoyé à une certaine vitesse, vitesse qui doit être synchronisée avec la possibilité de réception du MINITEL.

Or par la prise péritel, à l'initialisation, les données transitent à une vitesse de 1 200 bauds, ce qui correspond dans le cas du MINITEL à 1 200 bits par seconde. Ce qui correspond, pour chaque bit, à une durée d'environ $1/1\ 200 = 833\ 10^{-6} = 833$ micro-secondes.

Nous pouvons donc déjà écrire un algorithme général pour l'envoi d'un caractère par l'AMSTRAD, reçu par le MINITEL :

— DEBUT

- Envoyer le bit de start à 0
- Attendre 833 micro-secondes
- Envoyer le bit D0
- Attendre 833 micro-secondes
-
- Envoyer le bit D6
- Attendre 833 micro-secondes
- Envoyer le bit de parité

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

- Attendre 833 micro-secondes
- Envoyer le bit de stop à 1 (+5 Volts)
- Attendre 833 micro-secondes
- FIN

Pour la réception, il suffit de connaître comment est lue une donnée par la majorité des composants spécifiques gérant une ligne série :

Le composant attend en permanence une transition de +5 Volts à zéro Volt (bit de start). Comme une ligne série est généralement relativement longue, il lit à nouveau l'état de la ligne un demi-bit plus tard (c'est-à-dire ici : $833 / 2 =$ environ 416 micro-secondes plus tard), et vérifie que le bit de start est encore présent, sinon il recommence son attente. Ceci permet de « contrer » les éventuels parasites.

Les bits de données sont ensuite lus toutes les 833 micro-secondes (la lecture s'effectuera donc au milieu de chaque bit de donnée, ce qui évitera les erreurs dues aux temps de transition), puis la parité. S'il y a incompatibilité entre la parité et la donnée, le composant le signale (par l'affichage d'un caractère d'erreur — un point d'interrogation renversé — sur le MINITEL).

Suite à la réception du bit de stop, le composant se positionne de nouveau en attente d'un bit de stop. Voici donc l'algorithme général de réception que nous utiliserons sur l'AMSTRAD :

- DEBUT
 - REPETER
 - REPETER
 - Lire l'état de la ligne
 - JUSQU'A ce qu'un bit de start soit présent
 - Attendre 416 micro-secondes
 - Lire l'état de la ligne
 - JUSQU'A ce que le bit de start soit présent
 - Attendre 833 micro-secondes
 - Lire le bit D0
 - Attendre 833 micro-secondes
 - ...
 - Lire le bit D6
 - Attendre 833 micro-secondes

- Lire la parité
- Attendre 833 micro-secondes
- Attendre 833 micro-secondes
 - CO
 - pour « oublier » le bit de stop
 - FINCO
- FIN

LES PROCÉDURES UTILITAIRES

Au vu des explications précédentes, nous allons donc réfléchir aux algorithmes des procédures, ou sous-programmes permettant de gérer la liaison série par les lignes choisies sur le port imprimante : génération de la parité, formation d'une trame au format du MINITEL, temporisations (833 et 416 micro-secondes), envoi de la trame, réception.

Comme les vitesses de transmission sont relativement élevées, le choix du langage de programmation évitera le Basic, et préconisera le langage machine — Turbo Pascal aurait pu convenir, mais tout le monde ne le possède pas. Rassurons tout de suite le programmeur Basic, nous fournirons le programme lui permettant d'entrer les codes.

La gestion de parité

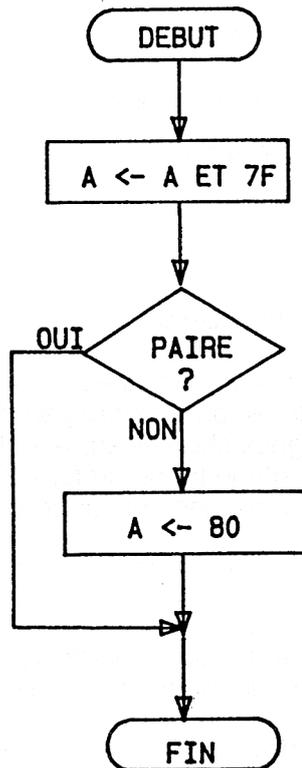
Chaque caractère envoyé ne comportant que 7 bits, nous annulerons automatiquement le bit numéro 8 (D7) et l'utiliserons pour y placer la parité calculée :

- DEBUT
 - Annuler le bit D7
 - SI le nombre de bits à 1 est impair
 - Placer D7 à 1
 - FINSI
- FIN

Ce qui permet de construire l'ordinogramme suivant, directement exploitable sur l'AMSTRAD-CPC, en langage d'assemblage.

La formation de la trame

La parité étant calculée, il faut ensuite créer une trame au format du MINITEL :



START D6 D5 D4 D3 D2 D1 D0 PARITE STOP

— DEBUT

— Enregistrer le bit de start à zéro

— POUR | DE 1 A 8

— Ajouter les 7 bits et la parité

— FINPOUR

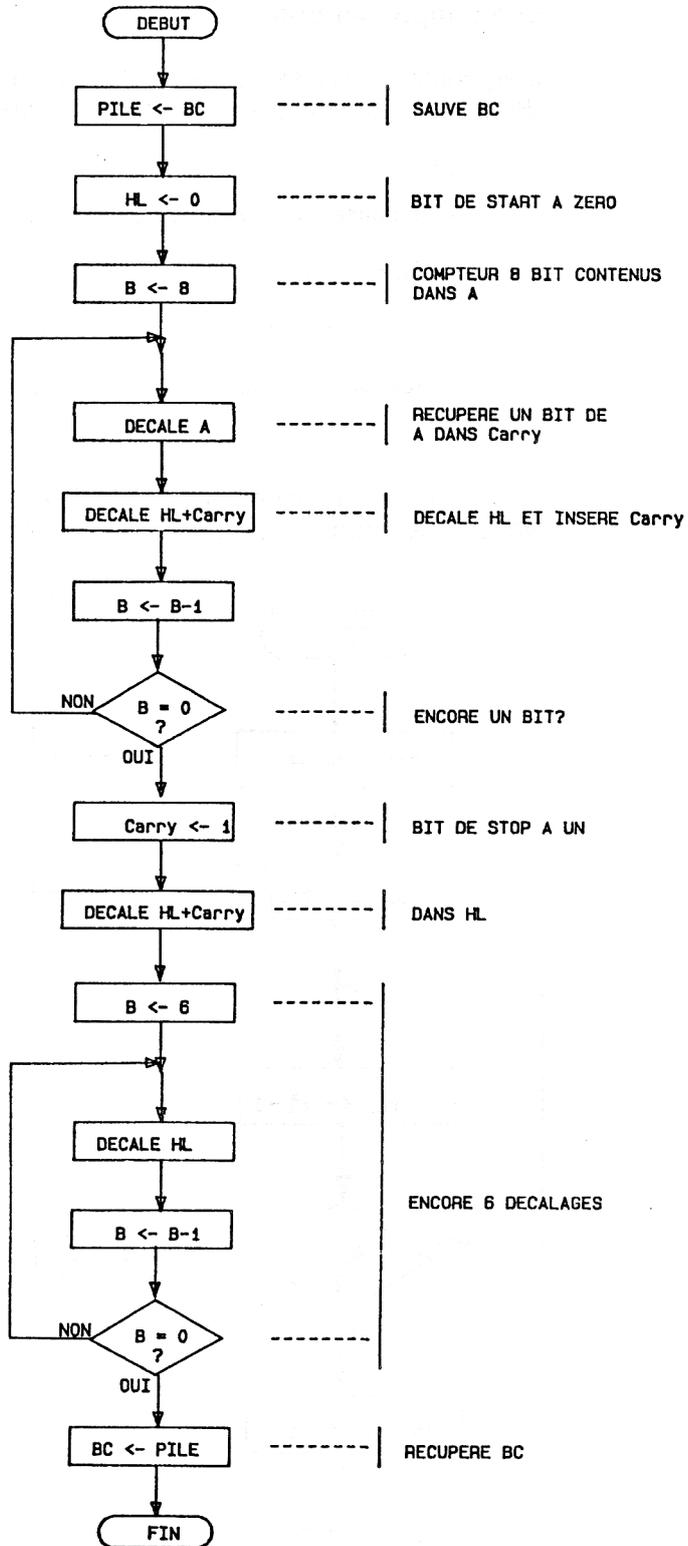
— Ajouter le bit de stop

— FIN

La traduction de cet algorithme en ordinogramme est un peu plus complexe, car il faut compter sur un registre double, tel HL, pour mémoriser la trame de 10 bits, et la préparer pour l'émission. Ce qui nous donne l'ordinogramme page ci-après :



Partie 10 : Fabrication de circuits additionnels pour AMSTRAD



Les temporisations

L'algorithme de la temporisation consistera à boucler un certain nombre de fois, ce qui fera perdre du temps, entre deux bits à envoyer.

— DEBUT

— Initialiser une variable de comptage

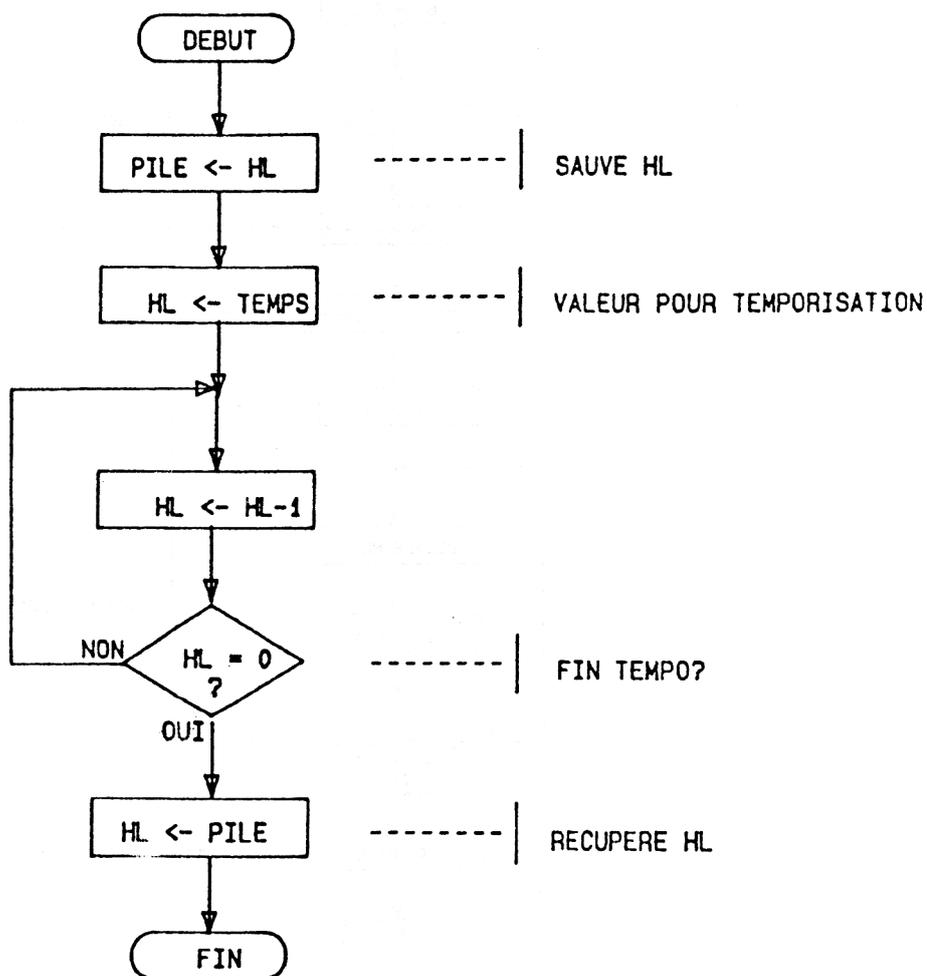
— REPETER

— Décrémenter la variable de comptage

— JUSQU'A ce que la valeur soit nulle

— FIN

La valeur d'initialisation sera ainsi différente selon le temps à attendre. Nous obtiendrons l'ordinogramme suivant :



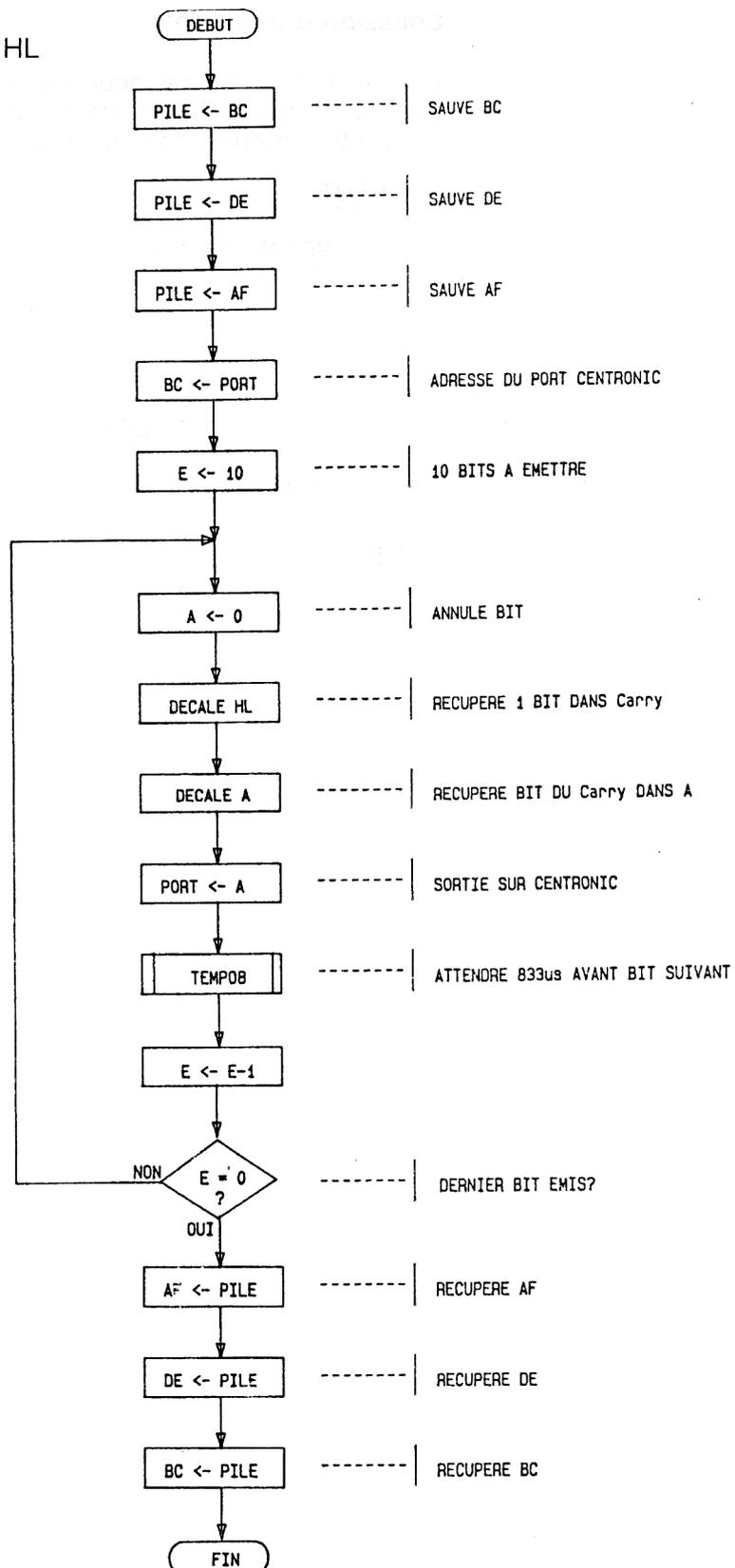
Emission d'une trame

L'émission consistera à récupérer un à un les bits de la trame, pour les émettre sur la ligne D0 du port imprimante, en intercalant une temporisation de 833 micro-secondes entre chaque bit.

- DEBUT
- POUR I DE 1 A 10
 - Récupérer un bit de la trame
 - Emettre ce bit
 - PROCEDURE : Attendre 833 micro-secondes
- FINPOUR
- FIN

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

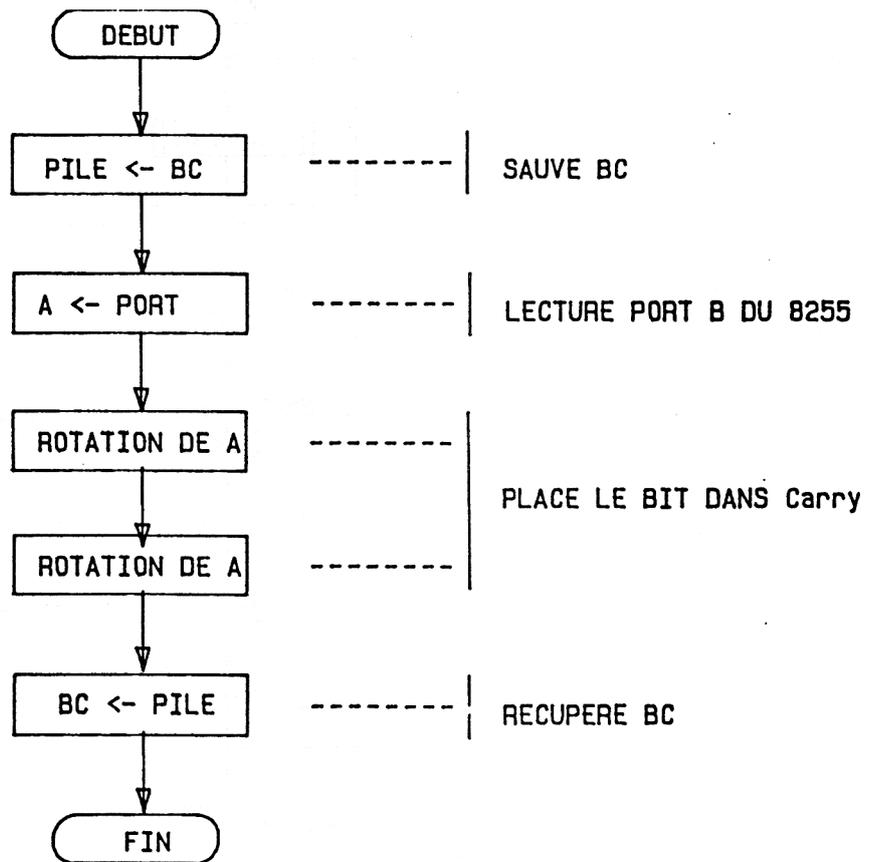
L'ordinogramme gérant l'émission de la trame sur le registre double HL se présente ainsi :



Réception d'un bit

La réception d'un bit s'effectuera par lecture de la ligne $\overline{\text{BUSY}}$.

Le bit intéressant se trouvera en position D6 (ligne B6 du port B du 8255), nous le décalerons donc deux fois vers la gauche (le bit de poids le plus fort étant à gauche) pour le récupérer dans le carry, ce qui nous permet de tracer l'ordinogramme suivant :



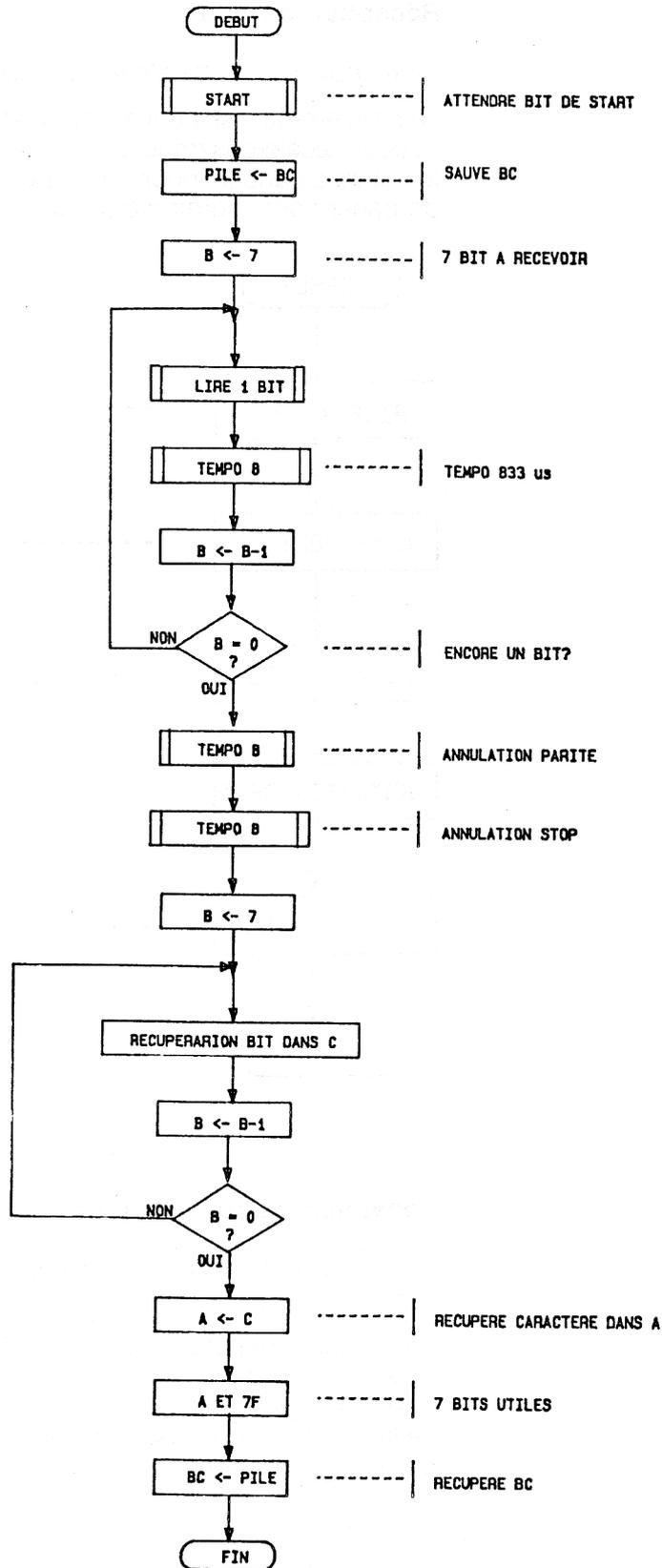
Réception complète d'un caractère

La réception d'un caractère complet de 7 bits passe par la lecture d'une trame entière de 10 bits.

L'algorithme général a été décrit plus haut, aussi nous nous contenterons de retenir que, pour des raisons évidentes de rapidité, chacun des bits sera empilé dans la pile, pour être enfin reconstitué, sans tenir compte de la parité (on supposera que le caractère arrive sans erreur, et en éliminant le bit de stop).

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

L'ordinogramme suivant retiendra toutes ces explications :



L'algorithme du programme principal

Pour l'exploitation de notre nouvelle connexion, nous avons prévu la gestion de la liaison série à partir du Basic.

Les programmeurs en langage machine pourront étudier le logiciel écrit sous Assembleur pour réutiliser tous les points d'entrée intéressants (nous donnerons suffisamment d'explications pour que leur tâche soit facilitée).

Nous avons prévu trois routines principales dans ce programme :

- envoi du code ASCII d'un caractère, ou d'un code spécifique, vers le MINITEL, via le fil RX ;
- envoi d'une chaîne de caractères vers le MINITEL, toujours sur RX ;
- réception du code ASCII d'un caractère émis par le MINITEL ou, éventuellement, d'un code spécifique, via la ligne TX.

Chacune de ces possibilités sera ainsi accessible par l'instruction CALL du Basic.

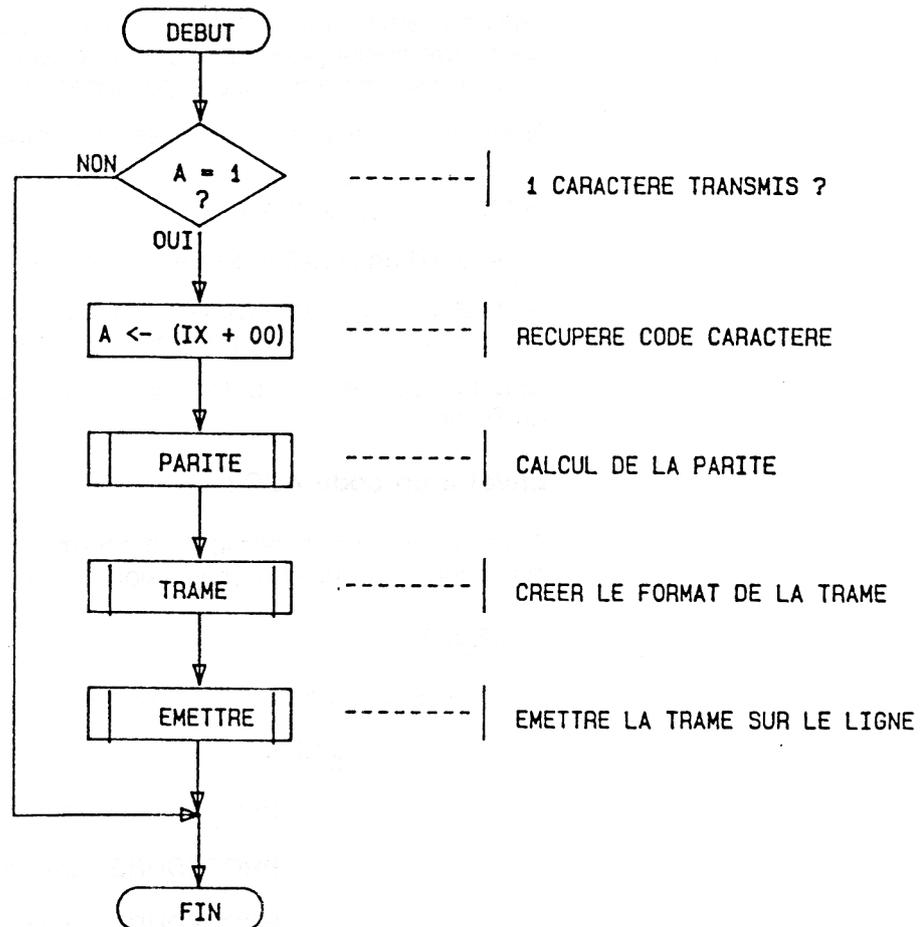
Envoi d'un code ASCII

Après vérification du nombre de paramètres transmis par CALL, le code est récupéré pour être émis selon l'algorithme suivant :

- DEBUT
 - SI un paramètre est transmis
 - ALORS
 - Récupérer le paramètre
 - PROCEDURE : calculer la parité
 - PROCEDURE : créer la trame
 - PROCEDURE : émettre la trame
 - FINSI
- FIN

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Les différentes procédures étant décrites plus haut, en tant que sous-programmes, voici l'ordinogramme résultant :



Envoi d'une chaîne de caractères

L'envoi d'une chaîne de caractères consistera à récupérer un à un les codes ASCII de chacun des caractères et à les émettre grâce au précédent algorithme :

— DEBUT

— SI une chaîne est transmise

— ALORS

— Récupérer le nombre de caractères de la chaîne

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

- POUR I DE 1 AU nombre de caractères
 - Récupérer l° caractère
 - PROCEDURE : émettre le code ASCII
 - CO
par l'algorithme précédent
 - FINCO
 - FINPOUR
- FINSI
- FIN

Après adaptation à l'assembleur Z80, nous obtenons l'ordinogramme présenté à la page suivante.

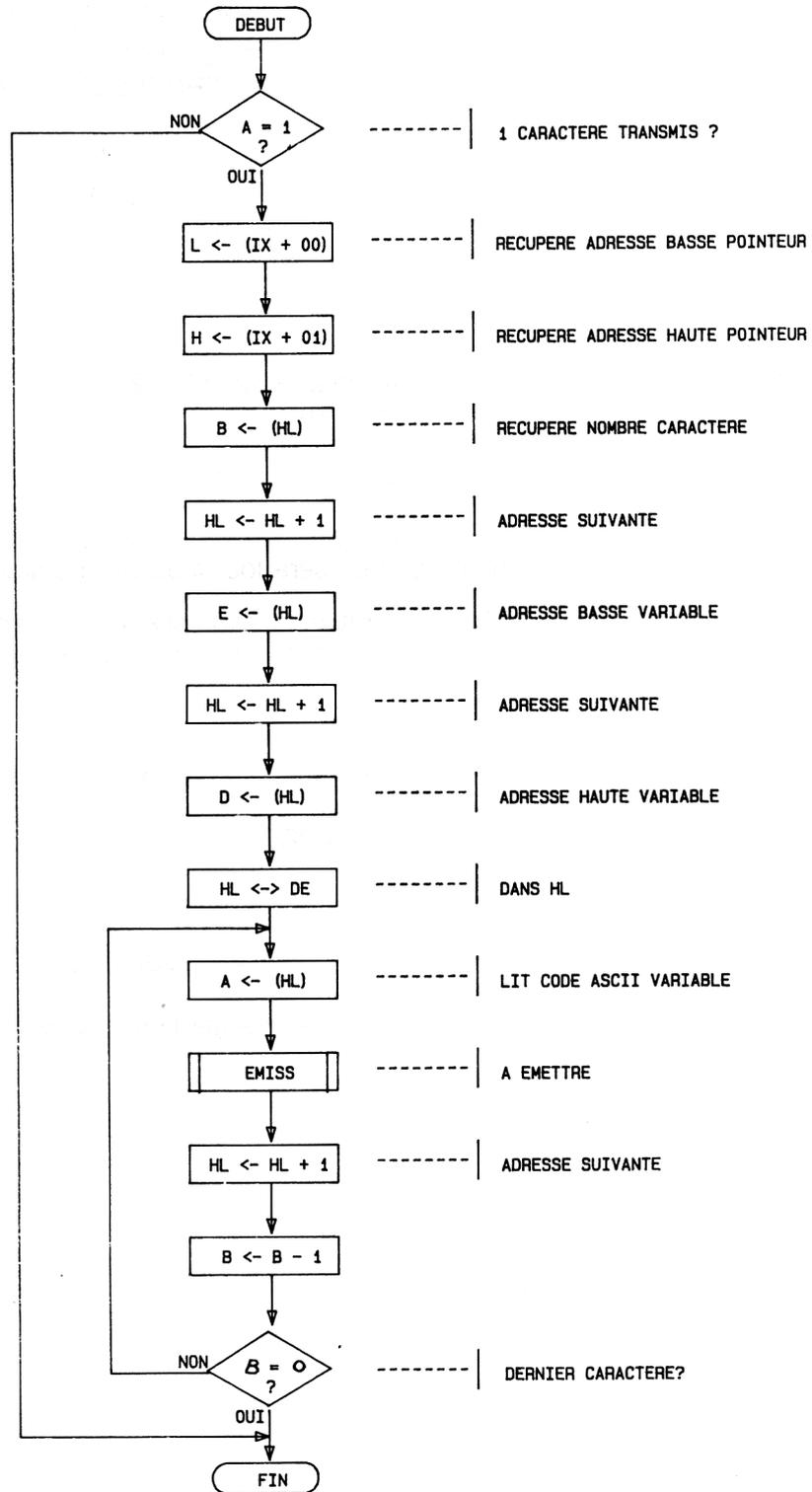
La réception d'un code

La réception d'un code est effectuée par l'intermédiaire d'une variable, dont le pointeur sera fourni au programme en langage machine.

Après récupération de l'adresse de la variable, le caractère sera attendu, autant de temps que cela sera nécessaire, puis rangé dans la variable.

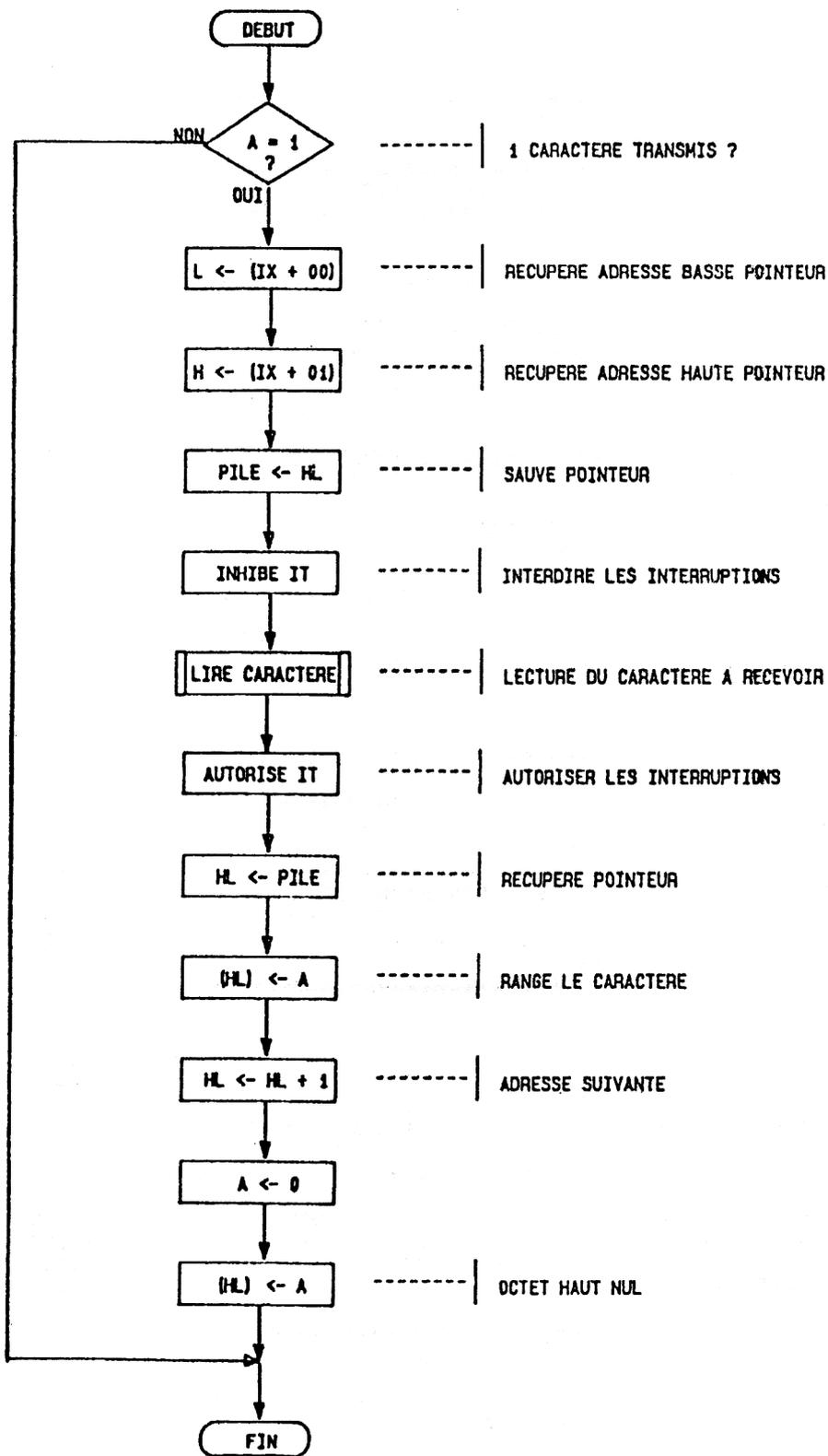
- DEBUT
 - SI une variable a été transmise
 - ALORS
 - Récupérer l'adresse de cette variable
 - PROCEDURE : Lire le caractère à recevoir
 - Ranger le caractère à l'adresse de cette variable
 - FINSI
- FIN

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD



Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Ordinogramme adapté de l'algorithme de la page 24 :



Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

LE LISTING ASSEMBLEUR

Grâce aux ordinogrammes précédents, nous avons écrit le programme assembleur suivant, largement commenté :

```

1      ;*****
2      ;* TRANSMISSION CPC VERS *
3      ;* MINITEL PAR LIAISON *
4      ;* CENTRONICS DU PORT *
5      ;* IMPRIMANTE *
6      ;*****
7      ;
8      ;
9      ;*** TABLE DES EQUIVALENCES ***
10     ;
11     ;PORT DU PIO POUR LIGNE BUSY
12     PORBUS: EQU 0F500H
13     ;
14     ;PORT IMPRIMANTE POUR LIGNE DONNEE
15     PORIMP: EQU 0EF00H
16     ;
17     ;POUR TEMPORISATION 833 ms
18     TEMP8: EQU 06DH
19     ;
20     ;POUR TEMPORISATION 833/2 ms
21     TEMP4: EQU 036H
22     ;
23     ;*****
24     ;
25     ;
26     ;*** ORIGINE D'ASSEMBLAGE ***
27     ;
28     ORG 0A000H
29     ;
30     ;*** ORIGINE DE CHARGEMENT ***
31     ;
32     LOAD 0A000H
33     ;
34     ;
35     ;*****
36     ;* EMISSION D'UN CODE *
37     ;* ASCII CONTENU DANS *
38     ;* UNE VARIABLE NUMERIQUE *
39     ;* NUMERIQUE *
40     ;*****
41     ;
42     EM: EQU $
43     A000 FE01 CP 01 ;UN CARACTERE TRANSMIS
44     A002 C0 RET NZ ;SINON RETOUR
45     ;
46     A003 DD7E00 LD A,(IX+00H) ;RECUPERE CARACTERE
47     ;
48     EMASC: EQU $
49     ;
50     A006 CD89A0 CALL PARITE ;CALCUL PARITE
51     A009 CD91A0 CALL TRAME ;FORMATION TRAME
52     A00C CDA7A0 CALL EMETTRE ;EMISSION TRAME
53     A00F C9 RET ;FIN
54     ;
55     ;*****
56     ;
57     ;
58     ;*****
59     ;* EMISSION D'UNE *
60     ;* CHAINE DE CARACTERES *

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

61          ;* CONTENUE DANS UNE          *
62          ;* VARIABLE ALPHANUMERIQUE *
63          ;* OU EXPLICITE             *
64          ;*****
65          ;
66          EMCAR:      EQU  S
67 A010 FE01          CP   01H          ;UNE CHAINE?
68 A012 C0            RET   NZ          ;SINON RETOUR
69
70 A013 DD6E00        LD   L,(IX+00H)   ;RECUPERE
71 A016 DD6601        LD   H,(IX+01H)   ;ADRESSE POINTEUR
72 A019 46            LD   B,(HL)       ;RECUPERE LE NOMBRE
73          ;DE CARACTERES DE LA CHAINE
74 A01A 23            INC   HL          ;ADRESSE SUIVANTE
75 A01B 5E            LD   E,(HL)       ;OCTET BAS
76 A01C 23            INC   HL          ;ADRESSE SUIVANTE
77 A01D 56            LD   D,(HL)       ;OCTET HAUT
78          ;DE L'ADRESSE DE LA CHAINE
79 A01E EB            EX   DE,HL        ;DANS HL POUR
80          ;SERVIR DE POINTEUR
81          ;
82          EMCA1:     EQU  S
83 A01F 7E            LD   A,(HL)       ;UN CARACTERE
84 A020 E5            PUSH HL          ;SAUVE HL
85 A021 C5            PUSH BC          ;ET BC
86 A022 CD06A0        CALL EMASC       ;EMETTRE LE CARACTERE
87 A025 C1            POP  BC          ;RECUPERER
88 A026 E1            POP  HL          ;LES REGISTRES
89 A027 23            INC   HL          ;ADRESSE SUIVANTE DE
90          ;LA CHAINE DE CARACTERES
91 A028 10F5          DJNZ EMCA1       ;ENCORE UN, ALORS
92          ;RETOURNER POUR LE CARACTERE
93          ;SUIVANT
94 A02A C9            RET               ;SINON FIN
95          ;
96          ;*****
97          ;
98          ;
99          ;*****
100         ;* RECEPTION DU CODE          *
101         ;* ASCII D'UN CARACTERE*
102         ;* EMIS PAR LE MINITEL *
103         ;*****
104         ;
105         RE:        EQU  S
106 A02B FE01          CP   01H          ;UN PARAMETRE SEULEMENT?
107 A02D C0            RET   NZ          ;NON ALORS RETOUR
108         ;
109 A02E DD6E00        LD   L,(IX+00H)   ;ADRESSE BASSE
110 A031 DD6601        LD   H,(IX+01H)   ;ADRESSE HAUTE
111         ;DE LA VARIABLE ENTIERE TRANSMISE
112 A034 E5            PUSH HL          ;A SAUVER
113 A035 F3            DI               ;INHIBER INTERRUPTION
114 A036 CD40A0        CALL RE7BIT      ;RECEVOIR UN CARACTERE
115         ;SUR LA LIGNE
116 A039 FB            EI               ;AUTORISER INTERRUPTION
117 A03A E1            POP  HL          ;RECUPERER ADRESSE
118 A03B 77            LD   (HL),A      ;ET SAUVER
119         ;LE CODE DU CARACTERE RECU
120 A03C 23            INC   HL          ;ADRESSE SUIVANTE

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

121                ; POUR POINTER L'OCTET HAUT
122 A03D AF        XOR  A                ; ET ANNULER
123 A03E 77        LD   (HL),A          ; CET OCTET HAUT
124 A03F C9        RET                  ; ET FIN
125                ;
126                ;*****
127                ;
128                ;
129                ;*** ZONE DES SOUS-PROGRAMMES ***
130                ;
131                ;
132                ;*****
133                ;* RECEPTION ET RANGEMENT *
134                ;* D'UN CARACTERE DANS *
135                ;* LE REGFISTRE A *
136                ;*****
137                ;
138 RE7BIT:         EQU  $
139                ;
140 A040 CD61A0     CALL STWAIT          ; ATTENDRE BIT DE START
141 A043 C5        PUSH BC              ; SAUVER BC
142 A044 0607     LD   B,7              ; SEPT BIT A RECEVOIR
143                ;
144 RE7BI1:         EQU  $
145 A046 CD7FA0     CALL LECTUR          ; LIRE UN BIT
146 A049 F5        PUSH AF              ; ET LE SAUVER
147                ; DANS LA PILE PARAMETRES
148 A04A CDC0A0     CALL TEMPO8          ; ATTENDRE 833 ms
149 A04D 10F7     DJNZ RE7BI1          ; ENCORE UN BIT A LIRE
150 A04F CDC0A0     CALL TEMPO8          ; ELIMINE PARITE
151 A052 CDC0A0     CALL TEMPO8          ; ELIMINE BIT STOP
152 A055 0607     LD   B,7              ; SEPT CARACTERES
153                ; A RECUPERER DE LA PILE
154 RE7BI2:         EQU  $
155 A057 F1        POP  AF              ; RECUPERE UN BIT
156 A058 CB11     RL   C                ; ET RANGER DANS C
157 A05A 10FB     DJNZ RE7BI2          ; ENCORE UN BIT
158 A05C 79        LD   A,C            ; SINON REPRENDR DANS A
159 A05D E67F     AND  07FH            ; ET GARDER LES
160                ; SEPT BITS UTILES
161 A05F C1        POP  BC              ; RECUPERER BC
162 A060 C9        RET                  ; ET FIN
163                ;
164                ;*****
165                ;
166                ;
167                ;*****
168                ;* ATTENTE DU BIT DE START *
169                ;* PRECEDENT LE CODE SERIE *
170                ;* D'UN CARACTERE ENVOYE *
171                ;* PAR LE MINITEL *
172                ;*****
173                ;
174 STWAIT:         EQU  $
175 A061 CD7FA0     CALL LECTUR          ; LIRE L'ETAT
176                ; DE LA LIGNE BUSY
177 A064 38FB     JR   C,STWAIT        ; SI PAS START
178                ; ALORS RECOMMENCER
179 A066 CD72A0     CALL TEMPO4          ; ATTENDRE UN
180                ; DEMI BIT DE START

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

181 A069 CD7FA0          CALL LECTUR          ;ET RELIRE
182 A06C 38F3           JR   C,STWAIT       ;SI CE N'EST PLUS
183                   ;LE BIT DE START ALORS RECOMMENCER
184 A06E CDC0A0         CALL TEMPO8         ;SINON ATTENDRE
185                   ;TEMPO POUR BIT SUIVANT
186 A071 C9            RET              ;ET RETOUR
187                   ;
188                   ;*****
189                   ;
190                   ;
191                   ;*****
192                   ;* TEMPORISATION D'UN DEMI *
193                   ;* BIT POUR VERIFICATION *
194                   ;* DU BIT DE START *
195                   ;*****
196                   ;
197 TEMPO4:             EQU $
198 A072 E5             PUSH HL              ;SAUVER HL
199 A073 F5             PUSH AF              ;ET AF
200 A074 213600        LD   HL,TEMP4       ;OCTET POUR LA
201                   ;DUREE DE LA TEMPORISATION
202                   ;
203 TEMP41:             EQU $
204 A077 2B            DEC  HL              ;UNE BOUCLE DE MOINS
205 A078 7D            LD   A,L             ;HL
206 A079 B4            OR   H              ;EGAL A ZERO
207 A07A 20FB         JR   NZ,TEMP41     ;NON ALORS RECOMMENCER
208 A07C F1            POP  AF              ;RECUPERER
209 A07D E1            POP  HL              ;LES REGISTRES
210 A07E C9            RET              ;ET FIN
211                   ;
212                   ;*****
213                   ;
214                   ;
215                   ;*****
216                   ;* LECTURE D'UN BIT SUR *
217                   ;* LA LIGNE BUSY DU PORT *
218                   ;* IMPRIMANTE *
219                   ;*****
220                   ;
221 LECTUR:             EQU $
222 A07F C5            PUSH BC              ;SAUVER BC
223 A080 0100F5        LD   BC,PORBUS     ;ADRESSE LIGNE BUSY
224 A083 ED78         IN   A,(C)         ;LIRE ETAT DU PORT
225 A085 17            RLA              ;RECUPERER LE
226 A086 17            RLA              ;BIT DANS CARRY
227 A087 C1            POP  BC              ;RECUPERER BC
228 A088 C9            RET              ;ET RETOUR
229                   ;
230                   ;*****
231                   ;
232                   ;
233                   ;*****
234                   ;* GENERATION DE LA *
235                   ;* PARITE SUR LE 8IEME *
236                   ;* BIT DU CARACTERE ASCII *
237                   ;*****
238                   ;
239 PARITE:             EQU $
240 A089 E67F         AND  07FH          ;ANNULE BIT 7

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

241 A08B EA90A0          JP    PE,PARIT1          ;PARITE PAIRE?
242 A08E F680           OR     080H          ;NON ALORS METTRE
243                    ;UNE PARITE PAIRE SUR BIT 7
244                    PARIT1: EQU $          ;PARITE PAIRE
245 A090 C9             RET              ;FIN
246                    ;
247                    ;*****
248                    ;
249                    ;
250                    ;*****
251                    ;* CREATION D'UNE TRAME *
252                    ;* DE DIX BIT A EMETTRE *
253                    ;* DANS LE REGISTRE HL *
254                    ;*****
255                    ;
256                    TRAME: EQU $
257 A091 C5             PUSH BC
258 A092 2E00          LD     L,0          ;TOUT
259 A094 2600          LD     H,0          ;A ZERO
260 A096 0608          LD     B,08H       ;8 BIT A TRAITER
261                    ;
262                    TRAM1: EQU $
263 A098 1F            RRA
264 A099 ED6A          ADC  HL,HL
265 A09B 10FB          DJNZ TRAM1          ;PAS FINI?
266 A09D 37            SCF              ;FLAG A 1
267 A09E ED6A          ADC  HL,HL          ;POUR BIT DE STOP
268                    ;
269 A0A0 0606          LD     B,6          ;ENCORE 6 FOIS
270                    TRAM2: EQU $
271 A0A2 29            ADD  HL,HL          ;A DECALER SANS CARRY
272 A0A3 10FD          DJNZ TRAM2          ;TERMINE?
273                    ;
274 A0A5 C1            POP  BC
275 A0A6 C9            RET
276                    ;
277                    ;*****
278                    ;
279                    ;
280                    ;*****
281                    ;* EMISSION D'UNE TRAME *
282                    ;* DE DIX BIT CONTENUE *
283                    ;* DANS HL SUR LE BIT DE *
284                    ;* POIDS FAIBLE DU PORT *
285                    ;* IMPRIMANTE D0 *
286                    ;*****
287                    ;
288                    EMETTRE: EQU $
289 A0A7 C5            PUSH BC          ;SAUVER
290 A0A8 D5            PUSH DE          ;LES
291 A0A9 F5            PUSH AF          ;REGISTRES
292 A0AA F3            DI              ;INHIBER LES INTERRUPTION
293                    ;
294 A0AB 0100EF        LD     BC,PORIMP   ;ADRESSE PORT
295                    ; IMPRIMANTE
296 A0AE 1E0A          LD     E,0AH       ;DIX CARACTERES A
297                    ;EMETTRE
298                    EMET1: EQU $
299 A0B0 AF            XOR  A          ;ANNULE A ET CARRY
300 A0B1 29            ADD  HL,HL        ;ROTATION DU BIT

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

301 ;FORT DE H DANS CARRY
302 A0B2 17 RLA ;RECUPERE CARRY SUR
303 ;BIT FAIBLE DE A
304 A0B3 ED79 OUT (C),A ;SORTIR LE
305 ;BIT SUR LE PORT CONCERNE
306 A0B5 CDC0A0 CALL TEMPO8 ;ATTENTE POUR
307 ;PROCHAIN BIT
308 A0B8 1D DEC E ;PROCHAIN BIT
309 A0B9 20F5 JR NZ,EMET1 ;S'IL Y A?
310 A0BB FB EI ;AUTORISER A
311 ;NOUVEAU LES INTERRUPTIONS
312 A0BC F1 POP AF ;RECUPERER
313 A0BD D1 POP DE ;LES
314 A0BE C1 POP BC ;REGISTRES
315 A0BF C9 RET ;FIN
316 ;
317 ;*****
318 ;
319 ;
320 ;*****
321 ;* TEMPORISATION PERMETTANT *
322 ;* L'ATTENTE D'UN DELAI DE *
323 ;* 833 MICROSECONDES ENTRE *
324 ;* DEUX BITS ENVOYES EN *
325 ;* SERIE SUR LE PORT *
326 ;* DE L'IMPRIMANTE *
327 ;*****
328 ;
329 TEMPO8: EQU $
330 A0C0 E5 PUSH HL ;SAUVER HL
331 A0C1 F5 PUSH AF ;ET AF
332 A0C2 216D00 LD HL,TEMP8 ;NOMBRE DE BOUCLES
333 ;POUR UNE TEMPO DE 833 ms
334 ;
335 TEMP81: EQU $
336 A0C5 2B DEC HL ;UNE BOUCLE DE MOINS
337 A0C6 7D LD A,L ;HL
338 A0C7 B4 OR H ;EGAL ZERO?
339 A0C8 20FB JR NZ,TEMP81 ;NON ALORS RECOMMENCER
340 A0CA F1 POP AF ;RECUPERER
341 A0CB E1 POP HL ;AF ET HL
342 A0CC C9 RET ;RETOUR
343 ;
344 ;*****
345 ;
346 ;
347 END

```

Après sauvegarde du programme en version mnémoniques, vous pouvez assembler ce programme, corriger les éventuelles erreurs de frappe, et sauver la version binaire, comprise entre les adresses &A000 et &A0CC.

Nous vous conseillons de nommer cette version sous l'appellation MINITEL.BIN, car nous la réutiliserons ultérieurement, dans nos programmes d'application.

L'émission du code ASCII d'un caractère débute en ligne 43, adresse &A000.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

On retrouve l'appel des trois procédures : le calcul de la parité en ligne 50 (adresse &A006), la création de la trame de 10 bits en ligne 51 (adresse &A009), et l'émission de la trame en ligne 52 (adresse &A00C).

L'émission de la chaîne débute en ligne 67, adresse &A010, par test de la présence de la chaîne.

Les caractères récupérés, on utilise la routine précédente par un appel en ligne 86 (adresse &A022).

La réception d'un code débute en ligne 106, à l'adresse &A02B du programme.

Cette routine appelle la procédure RE7BIT (appel effectué en ligne 114, adresse &A036).

Suite à ces trois routines, viennent les zones de sous-programmes, à partir de la ligne 140, adresse &A040.

Ligne 140, adresse &A040 : sous-programme de réception de 7 bits. En sortie le caractère sur sept bits est contenu dans le registre A.

Ligne 175, adresse &A061 : sous-programme d'attente du bit de start. Pas de conditions d'entrée ni de sortie.

Ligne 198, adresse &A072 : temporisation de 416 micro-secondes, équivalente à un demi-bit sur une liaison série à 1 200 bauds.

Ligne 222, adresse &A07F : sous-programme de lecture d'un bit sur la ligne BUSY. Le bit lu se trouve placé dans le registre des indicateurs, en affectant le Carry.

Ligne 240, adresse &A089 : sous-programme ajoutant une parité paire au nombre 7 bit contenu dans le registre A, résultat sur huit bits, la parité étant sur D7.

Ligne 256, adresse &A091 : création d'une trame de 10 bits dans le double registre HL, à partir de l'octet (7 bits + parité) contenu dans A. Le format de HL en sortie est le suivant : Start D0 D1 D2 D3 D4 D5 D6 Parité Stop
0 0 0 0 0 0.

Ligne 289, adresse &A0A7 : Emission sur la ligne D0 du port imprimante des dix bits utiles de la trame. En sortie, la trame est perdue.

Ligne 330, adresse &A0C0 : temporisation de 833 micro-secondes, correspondante à un bit, à 1 200 bauds.

LE CHARGEUR BASIC

Afin de ne pas défavoriser le programmeur Basic, nous vous fournissons ci-dessous le listing du chargeur Basic des codes hexadécimaux, correspondant au programme Assembleur précédent.

```

10 REM *****
20 REM ***   GESTION D'UNE LIAISON   ***
30 REM ***   SERIE ENTRE UN AMSTRAD   ***
40 REM ***           ET UN MINITEL PAR           ***
50 REM ** LA FICHE PERI-INFORMATIQUE **
60 REM *****
70 REM
80 REM *** CHARGEUR BASIC ***
90 REM
100 ADR = &A000
110 I = 0
120 SOMME = 0
130 RESTORE 400
140 READ AS
150 IF AS = "XX" THEN 250
160 BS = "&" + AS
170 B = VAL (BS)
180 SOMME = SOMME + B
190 POKE ADR,B
200 ADR = ADR + 1
210 I = I + 1
220 PRINT I;CHR$(13);
230 GOTO 140
240 REM
250 IF SOMME <> 29203 THEN CLS:PRINT CHR
$(7);"ERREUR DANS LES DATAS":FOR I = 1 T
O 1000: NEXT I:LIST
260 REM
270 REM *** SAUVEGARDE ***
280 REM
290 MODE 1
300 PRINT "POUR SAUVEGARDER LA ROUTINE E
N BINAIRE"
310 PRINT "SAVE ";CHR$(34);"MINITEL.BIN"
;CHR$(34);",B,&A000,";RIGHT$(STR$(I),LEN
(STR$(I))-1)
320 PRINT
330 PRINT
340 PRINT "CHARGEMENT PAR"
350 PRINT
360 PRINT "MEMORY &9FFF"
370 PRINT "LOAD ";CHR$(34);"MINITEL.BIN"
;CHR$(34);",&A000"
380 PRINT
390 REM
400 REM *** CODES HEXADECIMAUX ***
410 REM
420 REM ***   EMISSION D'UN CODE   ***
430 REM ***   CONTENU DANS UNE   ***
440 REM ***   VARIABLE NUMERIQUE   ***
450 DATA FE,01,C0,DD,7E,00,CD,89

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

460 DATA A0,CD,91,A0,CD,A7,A0,C9
470 REM
480 REM *** EMISSION D'UNE CHAINE ***
490 REM *** DE CARACTERE CONTENUE ***
500 REM *** DANS UNE VARIABLE ALPHA ***
510 REM *** NUMERIQUE OU EXPLICITE ***
520 DATA FE,01,C0,DD,6E,00,DD,66
530 DATA 01,46,23,5E,23,56,EB,7E
540 DATA E5,C5,CD,06,A0,C1,E1,23
550 DATA 10,F5,C9
560 REM
570 REM *** RECEPTION D'UN CODE ***
580 REM *** ASCII D'UN CARACTERE ***
590 REM *** EMIS PAR LE MINITEL ***
600 DATA FE,01,C0,DD,6E
610 DATA 00,DD,66,01,E5,F3,CD,40
620 DATA A0,FB,E1,77,23,AF,77,C9
630 REM
640 REM *** ZONE DES SOUS PROGRAMMES ***
650 REM
660 REM *** RECEPTION ET RANGEMENT ***
670 REM *** D'UN CARACTERE DANS LE ***
680 REM *** REGISTRE A ***
690 DATA CD,61,A0,C5,06,07,CD,7F
700 DATA A0,F5,CD,C0,A0,10,F7,CD
710 DATA C0,A0,CD,C0,A0,06,07,F1
720 DATA CB,11,10,FB,79,E6,7F,C1
730 DATA C9
740 REM
750 REM *** ATTENTE DU BIT DE ***
760 REM *** START PRECEDANT UN ***
770 REM *** CARACTERE ENVOYE PAR LE ***
780 REM *** MINITEL ***
790 DATA CD,7F,A0,38,FB,CD,72
800 DATA A0,CD,7F,A0,38,F3,CD,C0
810 DATA A0,C9
820 REM
830 REM *** TEMPORISATION D'UN DEMI ***
840 REM *** BIT POUR BIT DE START ***
850 DATA E5,F5,21,36,00,2B
860 DATA 7D,B4,20,FB,F1,E1,C9
870 REM
880 REM *** LECTURE D'UN BIT SUR ***
890 REM *** LA LIGNE BUSY ***
900 DATA C5
910 DATA 01,00,F5,ED,78,17,17,C1
920 DATA C9
930 REM
940 REM *** GENERATION DE LA PARITE ***
950 REM *** SUR LE 8IEME BIT ***

```

```

960 DATA E6,7F,EA,90,A0,F6,80,C9
970 REM
980 REM *** CREATION D'UNE TRAME DE ***
990 REM ***      10 BITS A EMETTRE      ***
1000 DATA C5,2E,00,26,00,06,08
1010 DATA 1F,ED,6A,10,FB,37,ED,6A
1020 DATA 06,06,29,10,FD,C1,C9
1030 REM
1040 REM *** EMISSION D'UNE TRAME DE ***
1050 REM *** 10 BIT VERS LE MINITEL ***
1060 DATA C5
1070 DATA D5,F5,F3,01,00,EF,1E,0A
1080 DATA AF,29,17,ED,79,CD,C0,A0
1090 DATA 1D,20,F5,F3,F1,D1,C1,C9
1100 REM
1110 REM *** TEMPORISATION DE 833 ***
1120 REM *** MICROSECONDES POUR LA ***
1130 REM ** TRANSMISSION A 1200 BAUDS **
1140 DATA E5,F5,21,6D,00,2B,7D,B4
1150 DATA 20,FB,F1,E1,C9
1160 END
1170 REM *** FIN DE DATAS ***
1180 DATA XX

```

Si vous n'avez commis aucune erreur de frappe, vous pourrez sauvegarder le programme binaire créé par l'instruction :

```
SAVE "MINITEL.BIN",B,&A000,205
```

Toute réutilisation ultérieure devra s'effectuer par les commandes :

```
MEMORY &9FFF
LOAD "MINITEL.BIN",&A000
```

UTILISATION DU LOGICIEL

Comme prévu, vous possédez trois façons de communiquer avec le MINITEL grâce au programme binaire MINITEL.BIN.

Connectez votre MINITEL à l'AMSTRAD à l'aide d'une des trois connexions proposées.

Attention :

Cette opération doit s'effectuer **HORS TENSION !!!**

Mettez sous tension, d'abord le MINITEL, puis le CPC.

Chargez MINITEL.BIN, et suivez les conseils ci-après.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Envoyer un code

Pour envoyer un code ASCII vers le MINITEL, procédez comme suit :

```
A = &41 : REM Code ASCII du caractère A  
CALL &A000,A
```

Vous devez alors voir s'afficher le caractère A sur l'écran du MINITEL.

Si rien ne se passe : vérifiez la liaison. Si elle est correcte, vous avez alors fait une erreur de frappe dans le programme MINITEL.BIN.

Vous pouvez ensuite essayer plusieurs codes ASCII des caractères de l'alphabet, pour vous assurer que le premier essai n'était pas le fruit du hasard.

Envoyer une chaîne de caractères

L'étape précédente passée avec succès, vous pouvez maintenant envoyer une chaîne de caractères, ce qui ne devrait pas poser de problème.

```
A$ = "Ceci est une chaîne de caractères"  
CALL &A010,A$
```

Le message doit s'afficher aussitôt sur l'écran télématique.

De même que pour la commande suivante :

```
CALL &A010,"Voici une deuxième chaîne"
```

Vous trouvez peut-être ce dernier résultat agaçant ? Cela ne fait rien, frappez plutôt :

```
CALL &A010,CHR$(10) + CHR$(13) + "PASSAGE A LA LIGNE"
```

Voilà qui doit vous rassurer : le MINITEL ne gère pas le passage à la ligne, il faut lui envoyer. Mais nous en reparlerons plus loin.

Recevoir un code

Pour recevoir un code de la part du MINITEL, lancer le sous-programme de réception par :

```
CODE% = 0  
CALL &A02B,@CODE%
```

A partir de ce moment, votre AMSTRAD n'est plus disponible — vérifiez-le en frappant une touche.

Il attend la frappe d'une touche sur le clavier du MINITEL.

Frappez donc une lettre, puis sur le CPC :

```
PRINT CHR$(CODE%)
```

pour voir s'afficher cette lettre.

Ainsi, pour recevoir un caractère du MINITEL, il faut d'abord initialiser une variable entière quelconque par `Variable% = 0`, puis appeler le sous-programme en `&A02B` en lui fournissant le pointeur de la variable (`CALL &A02B,@Variable`).

Le code ASCII de la touche retournée se trouve alors dans la variable entière.

III. Pour débiter un dialogue avec le MINITEL

Nous allons ici vous fournir les éléments de base nécessaires pour effectuer un dialogue correct avec le MINITEL et notamment les jeux de caractères.

LE JEU DE CARACTÈRES G0

Un premier jeu de caractères, nommé G0, est utilisable par le MINITEL, et fournit les caractères les plus utilisés.

Vous trouverez en figure 8 les codes Hexadécimaux, Décimaux et le caractère auquel ils correspondent.

20	32		30	48	0	40	64	Ⓞ	50	80	P	60	96	Ⓜ	70	112	p
21	33	!	31	49	1	41	65	A	51	81	Q	61	97	a	71	113	q
22	34	"	32	50	2	42	66	B	52	82	R	62	98	b	72	114	r
23	35	#	33	51	3	43	67	C	53	83	S	63	99	c	73	115	s
24	36	\$	34	52	4	44	68	D	54	84	T	64	100	d	74	116	t
25	37	%	35	53	5	45	69	E	55	85	U	65	101	e	75	117	u
26	38	&	36	54	6	46	70	F	56	86	V	66	102	f	76	118	v
27	39	'	37	55	7	47	71	G	57	87	W	67	103	g	77	119	w
28	40	(38	56	8	48	72	H	58	88	X	68	104	h	78	120	x
29	41)	39	57	9	49	73	I	59	89	Y	69	105	i	79	121	y
2A	42	*	3A	58	:	4A	74	J	5A	90	Z	6A	106	j	7A	122	z
2B	43	+	3B	59	:	4B	75	K	5B	91	[6B	107	k	7B	123	☐
2C	44	,	3C	60	<	4C	76	L	5C	92	\	6C	108	l	7C	124	☐
2D	45	-	3D	61	=	4D	77	M	5D	93]	6D	109	m	7D	125	☐
2E	46	.	3E	62	>	4E	78	N	5E	94	^	6E	110	n	7E	126	☐
2F	47	/	3F	63	?	4F	79	O	5F	95	☐	6F	111	o	7F	127	■

Fig. 8

Ce jeu de caractères est disponible dès la mise sous tension du MINITEL, ou en lui envoyant le code hexadécimal &0F (15 décimal), si l'on était dans le jeu de caractères G1.

LE JEU DE CARACTÈRES G1

Ce jeu de caractères correspond aux caractères appelés semi-graphiques.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

HEXA	DECI	SYMBOLE	HEXA	DECI	SYMBOLE
20	32		30	48	
21	33		21	33	
22	34		22	34	
23	35		23	35	
24	36		24	36	
25	37		25	37	
26	38		26	38	
27	39		27	39	
28	40		28	40	
29	41		29	41	
2A	42		2A	42	
2B	43		2B	43	
2C	44		2C	44	
2D	45		2D	45	
2E	46		2E	46	
2F	47		2F	47	

Fig. 9 : Codage des caractères alphamosaïques.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

HEXA	DECI	SYMBOLE	HEXA	DECI	SYMBOLE
60	96		70	112	
61	97		71	113	
62	98		72	114	
63	99		73	115	
64	100		74	116	
65	101		75	117	
66	102		76	118	
67	103		77	119	
68	104		78	120	
69	105		79	121	
6A	106		7A	122	
6B	107		7B	123	
6C	108		7C	124	
6D	109		7D	125	
6E	110		7E	126	
6F	111		5F	95	

Fig. 10 : Codage des caractères alphamosaïques.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Comme vous pouvez le constater en figures 9 et 10, le graphisme sur MINITEL n'est pas prêt de rivaliser avec votre AMSTRAD.

Pour accéder à ce jeu de caractères, il faut d'abord envoyer le code hexadécimal &0E (14 décimal).

La première forme correspond aux caractères semi-graphiques plein.

Il en existe une seconde forme, appelée semi-graphique disjoint, qui se caractérise par une séparation entre les pavés allumés, et que l'on obtient en envoyant la suite de codes : &1B – &5A (27 – 90), suite à l'appel par &0E.

Il est possible de retourner aux caractères semi-graphiques plein en envoyant : &1B – &59 (27 – 89).

Pour quitter le jeu de caractères semi-graphiques, il suffit d'envoyer le code &0F (15).

LE JEU DE CARACTÈRES G2

Le jeu de caractères G2 représenté en figure 11 permet d'accéder à certains caractères spéciaux (flèches, lire anglaise, ...), mais surtout de créer des caractères accentués.

Pour accéder à ce jeu, il suffit de transmettre le code hexadécimal &19 (25 décimal), puis le code du caractère de G2. Le retour au jeu de caractères G0 est automatique.

Par exemple, pour afficher le é (e accent aigu), il faut envoyer la séquence &19 – &42 – &65 (25 – 66 – 101), qui accède à G2 (&19), prépare l'accent aigu (&42), puis l'affiche avec le e (&65).

IV. Votre premier jeu : puissance 4

Fort de ces quelques connaissances sur le MINITEL, il est temps maintenant que vous profitiez de ces connaissances pour jouer avec un partenaire au jeu PUISSANCE 4.

RAPPEL DE LA RÈGLE

Vous possédez un damier vertical de sept colonnes sur six lignes, dans lequel vous pouvez placer des pions chacun votre tour.

Il suffit d'indiquer une colonne, et de lâcher le pion en haut de celle-ci, la gravitation faisant le reste, le pion tombe dans la case libre la plus basse.

Le but du jeu est d'aligner quatre de ses pions aussi bien verticalement, horizontalement, que dans une position diagonale, ce qui met un terme à la partie, en désignant le gagnant.

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

23	35	£	41	65	`
24	36	\$	42	66	'
26	38	#	43	67	^
2C	44	←	48	72	..
2D	45	↑	4B	75	3
2E	46	→	6A	106	œ
2F	47	↓	7A	122	æ
30	48	•			
31	49	±			
38	56	÷			
3C	60	1/4			
3D	61	1/2			
3E	62	3/4			

Fig. 11

LE JEU ENTRE AMSTRAD ET MINITEL

Grâce à notre réalisation, il sera possible de jouer à deux, chacun devant un écran.

Chacun des joueurs possède les pions ronds, les pions représentés par une croix sont ceux de l'adversaire.

Le joueur sur le CPC désignera la colonne où il désire déposer son pion en déplaçant un curseur à l'aide des deux touches fléchées droite et gauche, il validera la position choisie à l'aide de la touche espace.

Sur le MINITEL, le joueur déplacera son curseur à l'aide de la touche # (dièse) pour aller vers la gauche, et la touche * (étoile) pour un déplacement à droite (attention, les touches du MINITEL ne sont pas à répéti-

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

tion, il faudra frapper plusieurs fois pour plusieurs déplacements). La validation s'effectuera comme sur le CPC, par la barre d'espacement.

Le gagnant, ou la partie nulle, sera indiqué automatiquement par l'ordinateur gérant toute la partie.

LE PROGRAMME

Nous ne développerons pas pour l'instant l'algorithme du programme, afin de vous permettre d'en profiter plus rapidement, seules quelques explications suivront les listings.

```

10 REM *****
20 REM ***      PUISSANCE      QUATRE      ***
30 REM ***              DEUX JOUEURS      ***
40 REM ***      VERSION AMSTRAD CONTRE      ***
50 REM ***              MINITEL          ***
60 REM *****
70 REM
80 MEMORY &9FFF
90 LOAD "MINITEL.BIN",&A000
100 REM
110 REM *** INITIALISATIONS ***
120 REM
130 DEFINT A-Z
140 CLIGNOS = CHR$(27) + CHR$(72):REM CL
IGNOTEMENT
150 DBLG$ = CHR$(27) + CHR$(79):REM DOUB
LE GRANDEUR
160 FLECHBASS = CHR$(25) + CHR$(47):REM
FLECHE VERS LE BAS
170 CALL &A000,&1B:CALL &A000,&3B:CALL &
A000,&60:CALL &A000,&58:CALL &A000,&52:R
EM BLOCAGE CLAVIER ECRAN DU MINITEL
180 REM
190 CALL &A000,12
200 CALL &A000,31
210 CALL &A000,64
220 CALL &A000,65
230 CALL &A000,32
240 CALL &A000,24
250 REM
260 REM *****
270 REM ***      DESSIN SUR LE CPC      ***
280 REM *****
290 REM
300 MODE 0
310 LOCATE 7,1
320 PRINT CHR$(24)+" PUISSANCE 4 "+CHR$(
24)

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

330 FOR I = 175 TO 632 STEP 64
340     MOVE I,70
350     DRAW I,260
360 NEXT I
370 FOR I = 68 TO 240 STEP 32
380     MOVE 174,I
390     DRAW 620,I
400 NEXT I
410 REM
420 REM *****
430 REM ***  DESSIN SUR MINITEL  ***
440 REM *****
450 REM
460 CALL &A000,31
470 CALL &A000,64
480 CALL &A000,75
490 AS=CHR$(27)+CHR$(93)+CHR$(32)+CHR$(2
7)+CHR$(78)+"PUISSANCE 4"+CHR$(27)+CHR$(
76)+CHR$(32)
500 CALL &A010,AS
510 REM *** DESSIN DES LIGNES VERTICALES
***
520 FOR I=70 TO 86 STEP 3
530     FOR K = I TO I+1
540         CALL &A000,31
550         CALL &A000,K
560         CALL &A000,75
570         FOR J = 1 TO 8
580             CALL &A000,124
590             CALL &A000,32
600             CALL &A000,32
610         NEXT J
620     NEXT K
630 NEXT I
640 REM
650 REM *** DESSIN DES LIGNES HORIZONTAL
ES ***
660 REM
670 FOR I = 72 TO 87 STEP 3
680     CALL &A000,31
690     CALL &A000,I
700     CALL &A000,76
710     FOR J = 1 TO 7
720         CALL &A000,126
730         CALL &A000,126
740         CALL &A000,32
750     NEXT J
760 NEXT I
770 REM
780 REM *** FLECHE EN POSITION INITIALE
***

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

790 REM
800 POSITION1% = 85:REM POSITION INITIAL
E SUR LE MINITEL
810 POSITION2% = 13:REM POSITION INITIAL
E SUR LE CPC
820 X = 4:REM POSITION INITIALE DANS TAB
LEAU
830 REM
840 REM *****
850 REM *** GESTION JEU SUR MINITEL ***
860 REM *****
870 REM
880 MINITEL% = 0
890 CALL &A000,31
900 CALL &A000,66
910 CALL &A000,65
920 CALL &A010,"C'EST A VOUS"
930 LOCATE 1,3
940 PRINT "C'EST AU MINITEL"
950 WHILE MINITEL% <> 32
960     CALL &A000,31
970     CALL &A000,69
980     CALL &A000,POSITION1%
990     LOCATE POSITION2%,3:PRINT CHR$(24
1)
1000     CALL &A010,DELGS
1010     CALL &A010,CLIGNOS
1020     CALL &A010,FLECHBASS
1030     REM
1040     REM *** LECTURE TOUCHE MINITEL *
**
1050     REM
1060     CALL &A02E,@MINITEL%
1070     REM
1080     REM *** VERIFICATION DEPLACEMENT
***
1090     REM
1100     IF MINITEL% = 35 AND POSITION1% <
94 THEN POSITION1% = POSITION1% + 3:POS
ITION2% = POSITION2% + 2:X = X + 1
1110     IF MINITEL% = 42 AND POSITION1% >
76 THEN POSITION1% = POSITION1% - 3:POS
ITION2% = POSITION2% - 2:X = X - 1
1120     REM
1130     REM PAS DE TOUCHE CONNUE
1140     REM
1150     IF MINITEL% <> 42 AND MINITEL% <>
35 AND MINITEL% <> 32 THEN CALL &A000,7
1160     REM
1170     REM *** EFFACEMENT ANCIENNE FLECH
E ***

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

1180 REM
1190 CALL &A000,31
1200 CALL &A000,69
1210 CALL &A000,POSITION1% - 3
1220 CALL &A010,DBLG$
1230 CALL &A000,32
1240 LOCATE POSITION2%+2,8
1250 PRINT CHR$(32)
1260 CALL &A000,31
1270 CALL &A000,69
1280 CALL &A000,POSITION1% + 3
1290 CALL &A010,DBLG$
1300 CALL &A000,32
1310 LOCATE POSITION2%-2,8
1320 PRINT CHR$(32)
1330 REM
1340 WEND
1350 REM
1360 REM *****
*
1370 REM *** POSITIONNEMENT DU PION **
*
1380 REM *****
*
1390 REM
1400 FOR I = 1 TO 6
1410 IF TABLEAU(X,I) <> 1 AND TABLEAU(X,I) <> 2 THEN NEXT I
1420 I = I-1
1430 IF I = 0 THEN CALL &A000,7:GOTO 880
1440 REM
1450 REM *** EFFACEMENT FLECHE ACTUELLE ***
1460 REM
1470 CALL &A000,31
1480 CALL &A000,69
1490 CALL &A000,POSITION1%
1500 CALL &A010,DELS$
1510 CALL &A000,32
1520 LOCATE POSITION2%,8
1530 PRINT CHR$(32)
1540 REM
1550 REM *** RECONSTITUTION DES DIFFERENTES POSITIONS SUR L'AMSTRAD ET SUR LE MINITEL ***
1560 REM
1570 TABLEAU(X,I) = 1:REM LE PION JOUE SUR LE MINITEL EST RANGE DANS LE TABLEAU
1580 XCPC = 7 + 2 * (X - 1):REM COORDONNEE X SUR LE CPC

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

1590 YCPC = 10 + 2 * (I - 1):REM COORDON
NEE Y SUR LE CPC
1600 XMINIT = 76 + 3 * (X - 1):REM COORD
ONNEE X SUR LE MINITEL
1610 YMINIT = 71 + 3 * (I - 1):REM COORD
ONNEE Y SUR LE MINITEL
1620 REM
1630 REM *** DESCENTE DU PION MINITEL **
*
1640 REM
1650 FOR J = 71 TO YMINIT STEP 3
1660 CALL &A000,31
1670 CALL &A000,J
1680 CALL &A000,XMINIT
1690 CALL &A010,DELGS
1700 CALL &A010,"O"
1710 FOR K = 1 TO 100: NEXT K
1720 CALL &A000,31
1730 CALL &A000,J
1740 CALL &A000,XMINIT
1750 CALL &A010,DELGS
1760 CALL &A010," "
1770 NEXT J
1780 REM
1790 REM *** DESCENTE DU PION DU CPC ***
1800 REM
1810 FOR J = 10 TO YCPC STEP 2
1820 LOCATE XCPC,J
1830 PRINT"X"
1840 FOR K = 1 TO 100:NEXT K
1850 LOCATE XCPC,J
1860 PRINT" "
1870 NEXT J
1880 REM
1890 REM *** POSITIONNEMENT ***
1900 REM
1910 CALL &A000,31
1920 CALL &A000,YMINIT
1930 CALL &A000,XMINIT
1940 CALL &A010,DELGS
1950 CALL &A010,"O"
1960 LOCATE XCPC,YCPC
1970 PRINT "X"
1980 REM
1990 REM *****
2000 REM *** VERIFICATION ALIGNEMENT ***
2010 REM *** REALISE SUR LE MINITEL ***
2020 REM *****
2030 REM
2040 GAGNE = 0

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

2050 FOR X = 1 TO 7
2060     FOR Y = 4 TO 6
2070         IF TABLEAU(X,Y) = 1 AND TAB
LEAU (X,Y-1) = 1 AND TABLEAU (X,Y-2) = 1
AND TABLEAU (X,Y-3) = 1 THEN GAGNE = 1
2080     NEXT Y
2090 NEXT X
2100 FOR X = 1 TO 6
2110     FOR Y = 4 TO 7
2120         IF TABLEAU(Y,X) = 1 AND TAB
LEAU (Y-1,X) = 1 AND TABLEAU (Y-2,X) = 1
AND TABLEAU (Y-3,X) = 1 THEN GAGNE = 1
2130     NEXT Y
2140 NEXT X
2150 FOR X = 1 TO 4
2160     FOR Y = 1 TO 3
2170         IF TABLEAU(X,Y) = 1 AND TAB
LEAU(X+1,Y+1) = 1 AND TABLEAU(X+2,Y+2) =
1 AND TABLEAU(X+3,Y+3) = 1 THEN GAGNE =
1
2180     NEXT Y
2190 NEXT X
2200 FOR X = 4 TO 7
2210     FOR Y = 1 TO 3
2220         IF TABLEAU(X,Y) = 1 AND TAB
LEAU(X-1,Y+1) = 1 AND TABLEAU(X-2,Y+2) =
1 AND TABLEAU(X-3,Y+3) = 1 THEN GAGNE =
1
2230     NEXT Y
2240 NEXT X
2250 IF GAGNE = 1 THEN GOTO 3980
2260 REM
2270 REM *** TEST SI JEU FINI ***
2280 REM
2290 FIN = 0
2300 FOR I = 1 TO 7
2310     IF TABLEAU(I,1) <> 1 AND TABLEAU
(I,1) <> 2 THEN FIN = 1
2320 NEXT I
2330 IF FIN <> 1 THEN GOTO 4310
2340 REM *** FLECHE EN POSITION INITIALE
***
2350 REM
2360 POSITION1% = 85:REM POSITION INITIA
LE SUR LE MINITEL
2370 POSITION2% = 13:REM POSITION INITIA
LE SUR LE CPC
2380 X = 4:REM POSITION INITIALE DANS TA
BLEAU
2390 REM

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

2400 REM *****
*
2410 REM *** GESTION JEU SUR LE CPC **
*
2420 REM *****
*
2430 REM
2440 AS = ""
2450 CALL &A000,31
2460 CALL &A000,66
2470 CALL &A000,65
2480 CALL &A010,"C'EST AU CPC"
2490 LOCATE 1,3
2500 PRINT "C'EST A VOUS" + SPACES(4)
2510 WHILE AS <> CHR$(32)
2520     CALL &A000,31
2530     CALL &A000,69
2540     CALL &A000,POSITION1%
2550     LOCATE POSITION2%,8
2560     PRINT CHR$(241)
2570     CALL &A010,DBLG$
2580     CALL &A010,CLIGNO$
2590     CALL &A010,FLECHBASS$
2600     REM
2610     REM *** LECTURE TOUCHE DU CPC *
**
2620     REM
2630     AS = INKEY$
2640     REM
2650     REM *** VERIFICATION DEPLACEMENT
SUR LE CPC ***
2660     REM
2670     IF AS = CHR$(243) AND POSITION2%
< 19 THEN POSITION1% = POSITION1% + 3:PO
SITION2% = POSITION2% + 2:X = X + 1
2680     IF AS = CHR$(242) AND POSITION2%
> 7 THEN POSITION1% = POSITION1% - 3:PO
SITION2% = POSITION2% - 2:X = X - 1
2690     REM
2700     REM PAS DE TOUCHE CONNUE
2710     REM
2720     IF AS <> CHR$(242) AND AS <> CHR
$(243) AND AS <> CHR$(32) AND AS <> "" T
HEN PRINT CHR$(7)
2730     REM
2740     REM *** EFFACEMENT ANCIEN CURSEU
R ***
2750     REM
2760     CALL &A000,31
2770     CALL &A000,69
2780     CALL &A000,POSITION1% - 3

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```

2790 CALL &A010,DBLGS
2800 CALL &A000,32
2810 LOCATE POSITION2%+2,8
2820 PRINT CHR$(32)
2830 CALL &A000,31
2840 CALL &A000,69
2850 CALL &A000,POSITION1% + 3
2860 CALL &A010,DBLGS
2870 CALL &A000,32
2880 LOCATE POSITION2%-2,8
2890 PRINT CHR$(32)
2900 REM
2910 WEND
2920 REM
2930 REM *****
*
2940 REM *** POSITIONNEMENT DU PION **
*
2950 REM *****
*
2960 REM
2970 FOR I = 1 TO 6
2980 IF TABLEAU(X,I) <> 1 AND TABLEAU(X,I) <> 2 THEN NEXT I
2990 I = I-1
3000 IF I = 0 THEN PRINT CHR$(7):GOTO 2440
3010 REM
3020 REM *** EFFACEMENT CURSEUR ACTUEL *
**
3030 REM
3040 CALL &A000,31
3050 CALL &A000,69
3060 CALL &A000,POSITION1%
3070 CALL &A010,DBLGS
3080 CALL &A000,32
3090 LOCATE POSITION2%,8
3100 PRINT CHR$(32)
3110 REM
3120 TABLEAU(X,I) = 2:REM PION DU CPC
3130 XCPC = 7 + 2 * (X - 1)
3140 YCPC = 10 + 2 * (I - 1)
3150 XMINIT = 76 + 3 * (X - 1)
3160 YMINIT = 71 + 3 * (I - 1)
3170 REM
3180 REM *** DESCENTE PION DU CPC ***
3190 REM
3200 FOR J = 10 TO YCPC STEP 2
3210 LOCATE XCPC,J
3220 PRINT "O"

```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```
3230 FOR K = 1 TO 100:NEXT K
3240 LOCATE XCPC,J
3250 PRINT " "
3260 NEXT J
3270 REM
3280 REM *** DESCENTE PION MINITEL ***
3290 REM
3300 FOR J = 71 TO YMINIT STEP 3
3310 CALL &A000,31
3320 CALL &A000,J
3330 CALL &A000,XMINIT
3340 CALL &A010,DBLGS
3350 CALL &A010,"X"
3360 FOR K = 1 TO 100: NEXT K
3370 CALL &A000,31
3380 CALL &A000,J
3390 CALL &A000,XMINIT
3400 CALL &A010,DBLGS
3410 CALL &A010," "
3420 NEXT J
3430 REM
3440 REM *** POSITIONNEMENT PIONS ***
3450 REM
3460 CALL &A000,31
3470 CALL &A000,YMINIT
3480 CALL &A000,XMINIT
3490 CALL &A010,DBLGS
3500 CALL &A010,"X"
3510 LOCATE XCPC,YCPC
3520 PRINT "O"
3530 REM
3540 REM *****
3550 REM *** VERIFICATION ALIGNEMENT ***
3560 REM *** REALISE SUR LE CPC ***
3570 REM *****
3580 REM
3590 GAGNE = 0
3600 FOR X = 1 TO 7
3610     FOR Y = 4 TO 6
3620         IF TABLEAU(X,Y) = 2 AND TAB
LEAU (X,Y-1) = 2 AND TABLEAU (X,Y-2) = 2
AND TABLEAU (X,Y-3) = 2 THEN GAGNE = 1
3630     NEXT Y
3640 NEXT X
3650 FOR X = 1 TO 6
3660     FOR Y = 4 TO 7
3670         IF TABLEAU(Y,X) = 2 AND TAB
LEAU (Y-1,X) = 2 AND TABLEAU (Y-2,X) = 2
AND TABLEAU (Y-3,X) = 2 THEN GAGNE = 1
3680     NEXT Y
```

```
3690 NEXT X
3700 FOR X = 1 TO 4
3710     FOR Y = 1 TO 3
3720         IF TABLEAU(X,Y) = 2 AND TAB
LEAU(X+1,Y+1) = 2 AND TABLEAU(X+2,Y+2) =
    2 AND TABLEAU(X+3,Y+3) = 2 THEN GAGNE =
    1
3730     NEXT Y
3740 NEXT X
3750 FOR X = 4 TO 7
3760     FOR Y = 1 TO 3
3770         IF TABLEAU(X,Y) = 2 AND TAB
LEAU(X-1,Y+1) = 2 AND TABLEAU(X-2,Y+2) =
    2 AND TABLEAU(X-3,Y+3) = 2 THEN GAGNE =
    1
3780     NEXT Y
3790 NEXT X
3800 IF GAGNE = 1 THEN GOTO 4140
3810 REM
3820 REM *** TEST SI JEU FINI ***
3830 REM
3840 FIN = 0
3850 FOR I = 1 TO 7
3860     IF TABLEAU(I,1) <> 1 AND TABLEAU
(I,1) <> 2 THEN FIN = 1
3870 NEXT I
3880 IF FIN <> 1 THEN GOTO 4310
3890 REM *** FLECHE EN POSITION INITIALE
***
3900 REM
3910 GOTO 800
3920 REM
3930 REM *****
3940 REM *** LE PARTENAIRE SUR ***
3950 REM *** LE MINITEL A GAGNE ***
3960 REM *****
3970 REM
3980 LOCATE 1,3
3990 PRINT "HELAS, VOUS AVEZ          PERDU"
4000 CALL &A000,31
4010 CALL &A000,67
4020 CALL &A000,67
4030 CALL &A010,"BRAVO, VOUS AVEZ GAGNE"
4040 CALL &A000,13:CALL &A000,10
4050 CALL &A010,"APPUYEZ SUR UNE TOUCHE
POUR UNE AUTRE PARTIE"
4060 CALL &A02B,@MINITEL%
4070 RUN 130
4080 REM
```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

```
4090 REM *****
4100 REM ***  LE PARTENAIRE  SUR  ***
4110 REM ***  L'AMSTRAD  A GAGNE  ***
4120 REM *****
4130 REM
4140 LOCATE 1,3
4150 PRINT"BRAVO, VOUS AVEZ"
4160 PRINT"GAGNE"
4170 PRINT"APPUYEZ SUR UNE
4180 PRINT"TOUCHE POUR UNE
4190 PRINT"AUTRE PARTIE
4200 CALL &A000,31
4210 CALL &A000,67
4220 CALL &A000,67
4230 CALL &A010,"HELAS, VOUS AVEZ PERDU"
4240 CALL &BB06
4250 RUN 130
4260 REM
4270 REM *****^*****
4280 REM ***  PARTIE  NULLE  ***
4290 REM *****
4300 REM
4310 CALL &A000,31
4320 CALL &A000,66
4330 CALL &A000,65
4340 CALL &A010,"PARTIE NULLE"
4350 CALL &A000,13
4360 CALL &A000,10
4370 CALL &A010,"APPUYEZ SUR UNE TOUCHE
POUR RECOMMENCER"
4380 LOCATE 1,3
4390 PRINT "PARTIE NULLE  "
4400 CALL &A02B,@MINITEL%
4410 RUN 130
```

Partie 10 : Fabrication de circuits additionnels pour AMSTRAD

Sauvez ce programme sous le nom PUIS4, et lancez-le en vérifiant que le programme MINITEL.BIN se trouve sur votre disquette, et que le connecteur AMSTRAD - MINITEL est branché.

Lignes 10 à 250 : il s'agit des initialisations relatives au minitel, au CPC et à diverses chaînes de caractères de contrôle du MINITEL.

Lignes 260 à 410 : dessin de la grille de jeu sur le CPC.

Lignes 420 à 770 : dessin de la grille de jeu sur le MINITEL.

Lignes 780 à 830 : positionne les flèches sur les colonnes centrales des deux grilles.

Lignes 840 à 1350 : sur le MINITEL, gère le déplacement de la flèche, vérifie la conformité du déplacement, attend aussi la validation pour déposer un pion.

Lignes 1360 à 1980 : anime la descente du pion posé par le MINITEL sur les deux écrans.

Lignes 1990 à 2390 : vérifie si une position n'est pas gagnante, sinon passe le clavier au CPC.

Lignes 2400 à 2920 : identique aux lignes 840 à 1350, mais pour le CPC.

Lignes 2930 à 3530 : identique aux lignes 1360 à 1980, mais pour le CPC.

Lignes 3540 à 3910 : vérifie si une position n'est pas gagnante, sinon passe le clavier au CPC.

Lignes 4090 à 4410 : affichage du gagnant selon les cas.

