moulin à paroles

Aux temps héroïques de la science fiction, les robots se comportaient exactement comme des êtres humains; ils étaient notamment dotés de la parole, ce qui en faisait quasiment nos égaux. Par la suite, les écrivains et auteurs de cette littérature ont appris les possibilités réelles et surtout les limites des ordinateurs, et la parole n'a plus été réservée qu'à une élite parmi les "super cerveaux électroniques". Mais l'histoire suit son cours, et nous en sommes arrivés aujourd'hui à un point du progrès où un circuit unique est capable de fournir un vocabulaire de quelques centaines de mots, et qu'il appartient à un système à microprocesseur de gérer convenablement.

Dans un précédent article ("puces bavardes" Elektor Septembre 1981), nous avons analysé plusieurs systèmes de synthèse de la parole. Diverses raisons nous ont conduit à considérer le système "solid State Speech" de Texas Instruments comme le plus intéressant, du moins pour l'amateur de systèmes à microprocesseurs. Il apparaît en premier lieu qu'il est en mesure de produire un signal de sortie comparable à une voix humaine filtrée par une ligne téléphonique: si l'on ne peut donc pas encore parler de "haute-fidélité" en la matière, il n'en est pas moins qu'un accent américain reste hautement reconnaissable! En outre, le codage est fait de manière "logique", c'est à dire qu'il est suffisamment pénétrable pour qu'on en puisse déduire des règles permettant de **coder des mots nouveaux** -sans qu'il soit donc nécessaire de faire appel à un gros système d'analyse par ordinateur-!

Maintenant que le choix est arrêté sur le système Texas Instruments, il reste encore à opter pour l'une ou l'autre version disponibles: la plus ancienne, TMS 5100, conçue essentiellement pour des jeux et autres application parlantes de ce genre; ou la deuxième plus récente, TMS 5200, qui a été conçue pour les systèmes à microprocesseurs. On pourra être surpris de notre choix, puisqu'il se porte sur le 5100! Il y a à cela au moins deux bonnes raisons: le vocabulaire disponible avec cette version est plus étendu, et l'application qu'en propose T1 est meilleure! Il ne faut que quelques modifications et ajouts pour que le système soit compatible (moyennant un circuit d'interface adéquat, bien sûr!) avec n'importe quel système à microprocesseur. Le principe de base de la synthèse vocale proprement dite fera l'objet d'explications ultérieures. Retenons pour l'instant qu'un flux sériel de bits doit être appliqué à l'entrée du VSP (Voice Synthesis Processor) pour que celui-ci se mette à parler. Prenons le mot "help"; il lui faudra en tout et pour tout 534 bits, soit un peu moins que 67 octets. A partir de cet exemple, on peut se faire une idée de l'étendue de mémoire qu'il faut pour un vocabulaire de quelques centaines de mots. Afin d'éviter de réquisitionner abusivement la mémoire disponible sur le système à microprocesseur, la mémoire vocabulaire se trouve sur le circuit de synthèse de parole, avec son adressage et son unité de commande propre.

La figure 1 illustre la structure globale du "moulin à parole". La moitié inférieure de ce schéma montre la mémoire et la circuiterie de commande. Pour commencer, la première adresse d'un mot donné doit être chargée dans le tampon/compteur d'adresses. Comme l'adressage se fait sur un format de seize bits, la première adresse est chargée en deux octets (soit deux fois huit bits): on commence par l'octet de poids faible que l'on place sur le bus de données, puis $\overline{\text{LDA 1}}$ voit son niveau logique inversé brièvement, après quoi c'est l'octet d'adresse de poids fort qui est chargé lorsque $\overline{\text{LDA Ø}}$ reçoit une impulsion. Le compteur de bits est remis à zéro lorsque le niveau logique de $\overline{\text{LDA 1}}$ change. Une fois que la première adresse a été chargée, l'unité est en mesure de recevoir l'ordre de "parler". Chaque impulsion d'horloge I/O du VSP incrémente le compteur de bits, de sorte que le convertisseur de flux de bits parallèle/série choisit un nouveau bit dans l'octet de mémoire-vocabulaire sélectionné. La même impulsion d'horloge I/O rythme l'introduction de chaque bit concerné dans une bascule, qui transmet le flux de bits au processeur de synthèse vocale. Lorsque le compteur de bits en a vu passer huit, il incrémente le tampon/compteur d'adresses, qui à son tour sélectionne l'octet de mémoire suivant.

Comme on le voit sur le schéma synoptique, la liaison entre le convertisseur de flux et la bascule qui lui fait suite peut être interrompue, et les deux côtés peuvent être amenés en relation avec le système à microprocesseur hôte. Ainsi des données de la mémoire-vocabulaire peuvent être chargées en mémoire vive du système à microprocesseur via la sortie Y, après avoir été modifiée pour l'obtention d'un mot ou d'un phrase différents, la donnée peut être réinjectée sur l'entrée D. Entre le circuit de synthèse vocale et le système à microprocesseur il faut bien sûr un circuit d'interface convenable -que nous publierons très prochainement-.

La partie supérieure du schéma synoptique montre le processeur vocal lui-même (VSP). Il est doté de deux entrées

Tableau 1.

CØ, C1:

Ordre	CØ	C1
initialisation	1	1
parole	0	1
test busy *	0	0
(invalidé)	1	0

* test "synthèse en cours"

$\overline{\text{Busy}}$:	lorsqu'elle est validée, cette ligne passe au niveau logique haut à la fin du mot
CCLK:	horloge de commande du processeur.
LDA1,LDAØ:	validation du chargement de l'octet faible et fort de l'adresse DØ . . . D7
$\overline{\text{I/O}}$,D,Y:	signaux de commande pour la mémoire/vocabulaire externe.

Tableau 1. Voici les trois commandes mises en oeuvre par les entrées de commande CØ et C1.

1

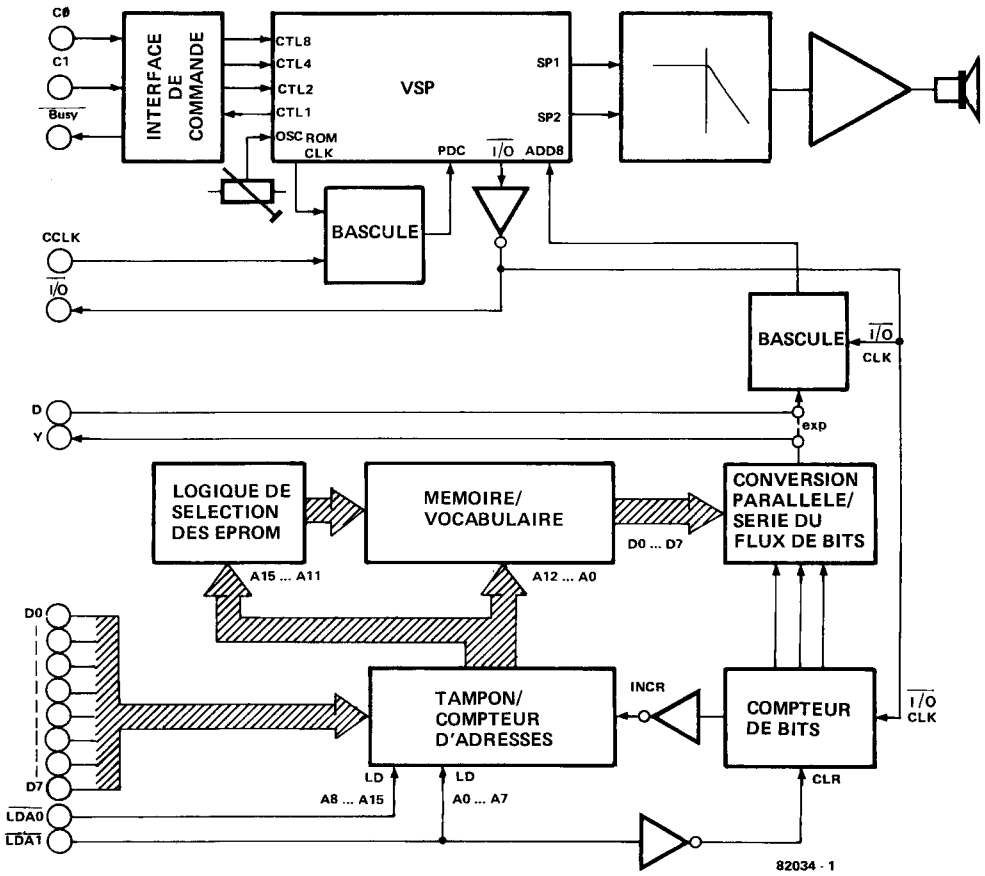

Figure 1. Schéma synoptique du "moulin à parole".

2

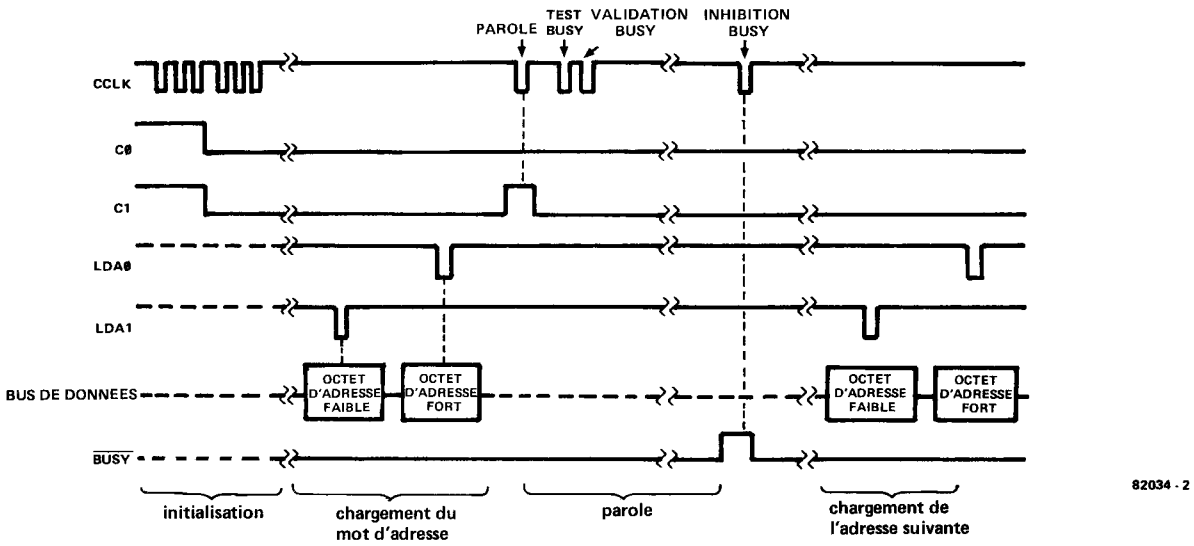

Figure 2. Les différents signaux de commande doivent être envoyés au circuit de synthèse de la parole dans l'ordre prévu.

de commande à gauche, CØ et C1; comme le montre le tableau 1, celles-ci lui fournissent les ordres "reset" (initialisation), "talk" (parole) et "test busy" (test "synthèse en cours"). Ce signal de contrôle agit sur la sortie "busy" (occupé) qui passe au niveau logique haut (si elle est validée) dès la fin d'une séquence parlée.

Le VSP contient un oscillateur d'horloge; entre autres fonctions, celui-ci détermine la hauteur du signal parlé. Afin de synchroniser l'entrée externe CCLK (control clock - commande de l'horloge) et cette horloge intégrée, les deux signaux sont appliqués à une bascule, dont le signal de sortie est appliqué à l'entrée PDC (processor data clock). Le VSP indique qu'il attend le bit suivant de l'octet de mémoire-vocabulaire en inversant la polarité de sa sortie $\overline{I/O}$; comme nous l'avons décrit ci-dessus, cette inversion de niveau logique provoque l'introduction du bit suivant dans la bascule, et la remise à jour du compteur de bits. Lorsque l'on introduit des données pour la parole depuis la mémoire vive extérieure (celle du système à microprocesseur en l'occurence) la sortie $\overline{I/O}$ est indispensable pour réaliser une bonne synchronisation. Pour finir, les deux sorties vocales sont appliquées à l'entrée d'un

3

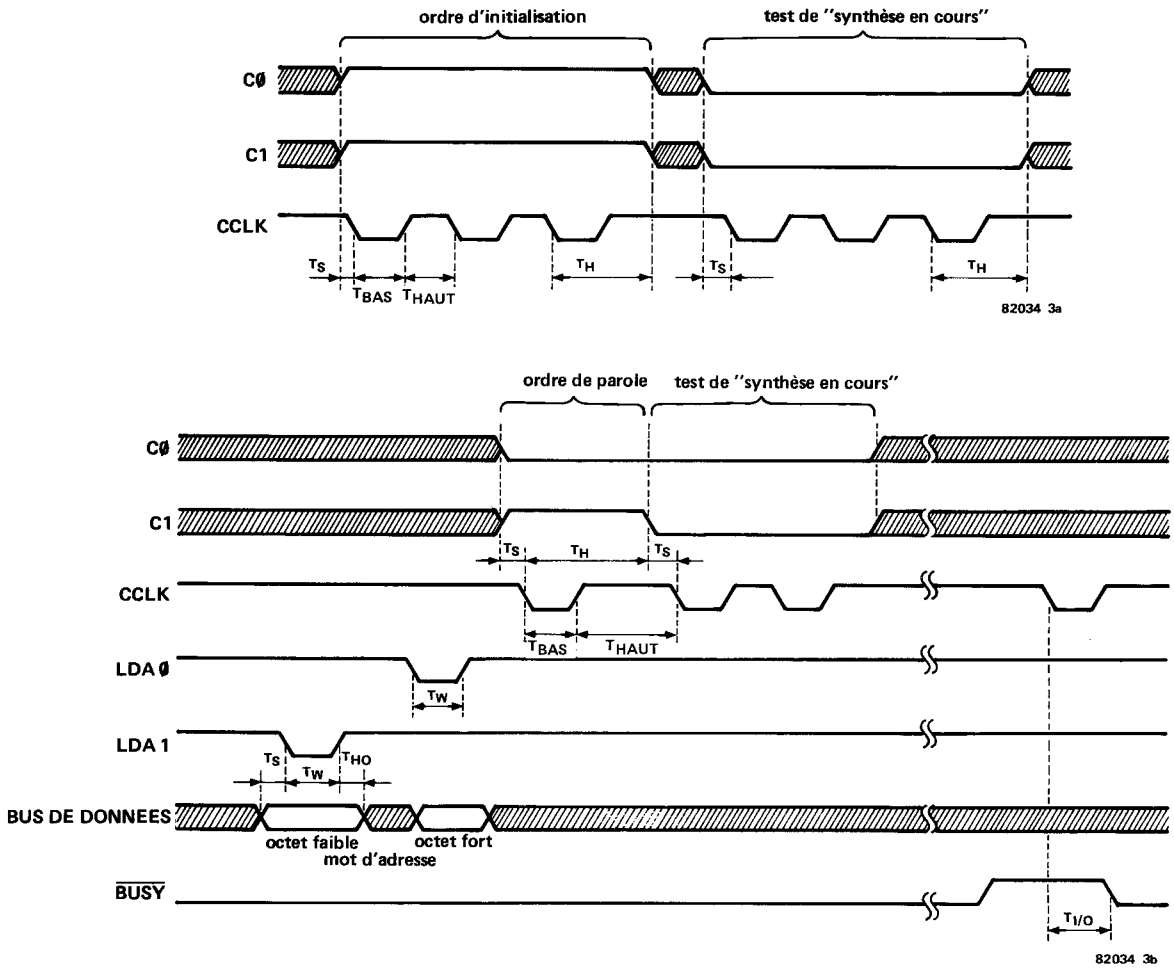

Figure 3. Nous savons que les signaux de commande doivent respecter certaines limites en ce qui concerne leur durée. La figure 3a montre ainsi la durée de ces signaux au cours de la procédure d'initialisation, tandis que la figure 3b illustre la situation que l'on a lorsque le "moulin à parole" 's exprime".

Tableau 2.

	min	max
T_S	0	—
T_{DOWN}	T = 6,25 µS	—
T_{UP}	T = 6,25 µS	—
T_H	1¾T = 10,9 µS	—
T_W	20 ns	—
T_{HO}	0	—
$T_{I/O}$	1¼T = 7,8 µs	8,1 µS

$T = T_{ROMCLK} = 6{,}25\ \mu s$

Tableau 2. Telles sont les limites dans lesquelles doivent rester les différents signaux de commande.

filtre passe-bas, d'où elles attaquent l'amplificateur et le haut-parleur.

Synchronisation des signaux

Il est indispensable que les différents signaux de commande doivent apparaître selon une chronologie rigoureuse; c'est ce qui apparaît sur la figure 2. Après la mise sous-tension, il faut initialiser le circuit. On applique pour cela un niveau logique haut à CØ et C1 (c'est à dire "reset" et on inverse le niveau logique de l'entrée CCLK trois fois de suite; puis CØ et C1 sont mis au niveau logique bas (test busy) et le niveau logique à l'entrée CCLK est de nouveau inversé trois fois de suite. A présent, le circuit est prêt pour une séquence de synthèse de la voix.

Pour "sortir" un mot, l'octet d'adresse de poids faible est chargé sur le bus de données, et $\overline{LDA\ 1}$ est mis au niveau logique bas pendant un court instant; puis c'est l'octet d'adresse de poids fort qui est chargé sur le bus de données et $\overline{LDA\ Ø}$ reçoit une impulsion. Ensuite, C1 est mis au niveau logique haut (CØ reste au niveau logique bas), ce qui équivaut à l'ordre "talk", et le niveau logique à l'entrée CCLK est inversé. Ceci amorce la sortie parole. Entre temps, C1 est revenu au niveau logique bas, et l'entrée CCLK voit son niveau logique inversé deux fois. Ceci valide la sortie "busy", qui passe par conséquent au niveau logique, haut dès la fin du mot en cours de synthèse. A ce moment, une nouvelle impulsion sur CCLK remet le VSP en position d'attente pour le mot suivant.

Tous les signaux de commande doivent être synchronisés comme l'indiquent la figure 3 et le tableau 2. La section de la figure 3a correspond à la phase d'initialisation; le point important de cette procédure est la longueur des impulsions sur CCLK, qui doit être suffisante pour que la synchronisation soit assurée avec l'oscillateur "ROMCLK" du VSP. C'est à dire que T_{bas} et T_{haut} doivent durer chacun 6,25 µs au moins, dans la plupart des applications. La figure 3b illustre le diagramme des impulsions pendant la parole. La période de T_W, pour la charge de l'octet de poids faible et l'octet de poids fort de l'adresse du mot à charger en mémoire-vocabulaire, doit être assez longue pour que le verrouillage du tampon/compteur d'adresses soit assuré, soit 20 ns ou plus. Les portions hachurées sur les lignes CØ et C1, de même que sur le bus de données indiquent que les niveaux logiques n'ont pas d'importance sur ces lignes pendant ce temps-là.

Le circuit

La structure d'ensemble du circuit de la figure 4 correspond à celle du schéma synoptique que nous avons étudié à la figure 1. Partons du haut: T1, T2 et T3 convertissent les signaux sur CØ et C1 en signaux de commande tels que les requiert le processeur, tandis que N2 tamponne la sortie BUSY. P1 détermine

4

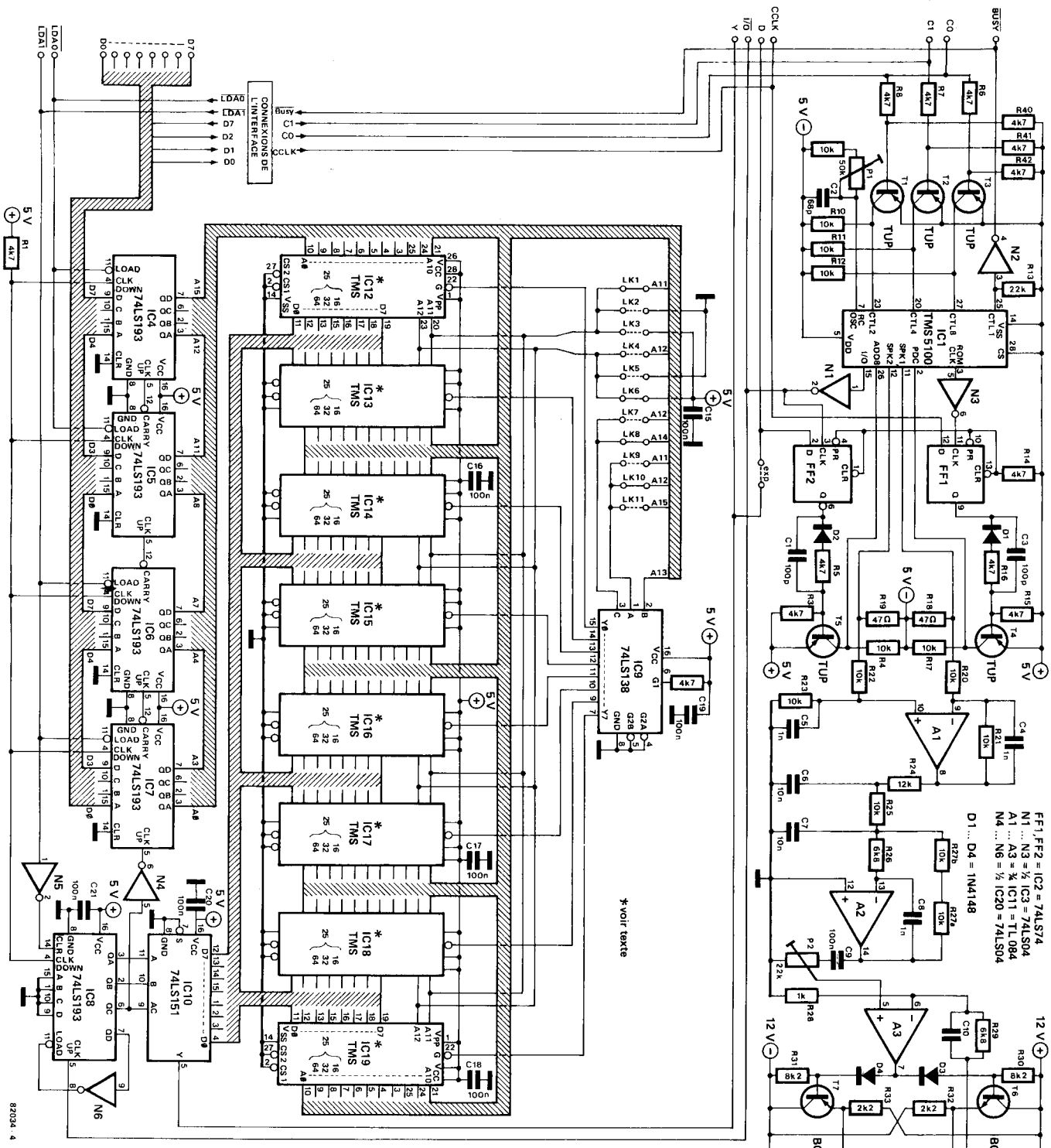

Figure 4. Schéma de principe complet du montage de synthèse de la parole. Comme on peut s'en apercevoir, la disposition des composants suit d'assez près celle des blocs correspondants du schéma synoptique.

Tableau 3.

	E	R	P	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	FRAME TYPE
	0000													SILENCE
HEL	0100	0	00000	10011	01110	1001	0111							UV
	0111	1	00000											UV – REPEAT
	1101	0	10010	10000	10100	1000	0110	0111	1000	1010	100	101	010	V
	1101	1	10011											V – REPEAT
	1110	1	10011											V – REPEAT
	1101	0	10100	01101	01111	1010	1010	1001	0111	1000	100	101	101	V
	1101	0	10100	01110	01011	1000	1100	1101	1000	0100	100	011	101	V
	1101	0	10011	10001	01010	0110	1001	1111	1011	0101	010	000	110	V
	1011	1	11010											V –REPEAT
	1010	0	10010	01101	00111	1000	1100	1111	0111	0010	001	010	110	V
	1001	1	10001											V – REPEAT
	1001	1	01110											V – REPEAT
	1000	1	01101											V – REPEAT
	0010	0	01110	00101	00101	1101	1001	1110	0101	0111	001	011	011	V
	0000													SILENCE
	0000													SILENCE
	0000													SILENCE
P	0111	0	00000	10100	01011	1011	1000							UV
	0111	0	00000	10001	01011	1011	0110							UV
	0101	1	00000											UV – REPEAT
	0011	0	00000	10011	00111	1010	0110							UV
	0010	0	00000	10010	00101	1011	0101							UV
	0000													SILENCE
	1111													STOP CODE

V = Voisé E = Energie P = Hauteur
UV = Dévoisé R = Répétition $K_1 \ldots K_{10}$ = Paramètres des filtres

Tableau 3. La séquence de codes numériques que voici va faire dire "help" à la puce loquace de Texas Instruments!

5

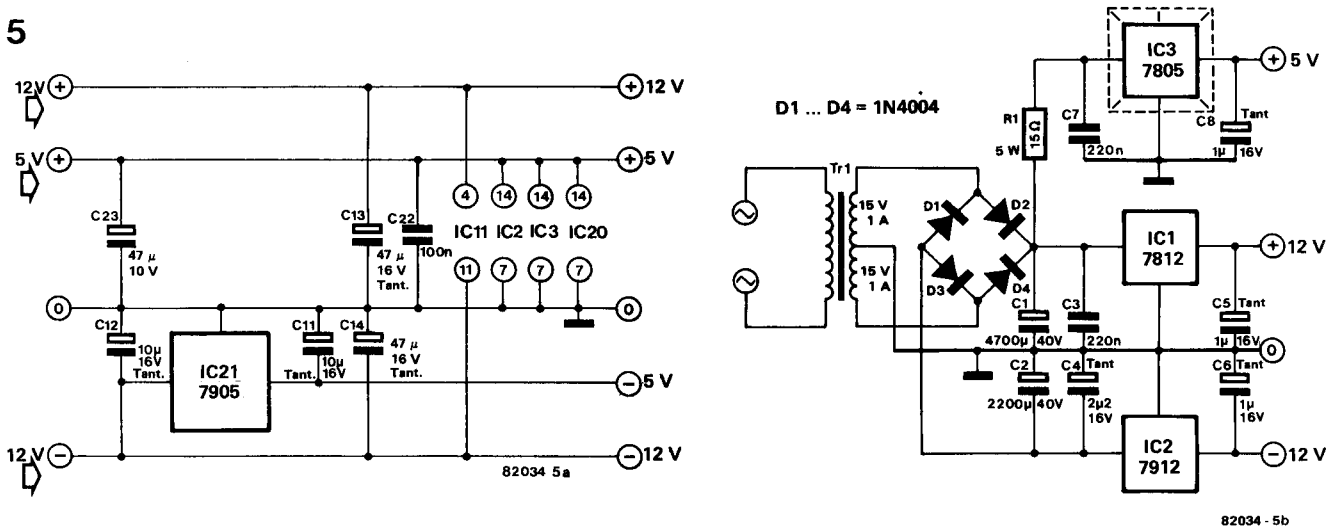

Figure 5. Dans la plupart des cas, il est possible de dériver les diverses tensions nécessaires à l'alimentation du "moulin à paroles" à partir de l'alimentation de l'ordinateur hôte, (figure 5a). Mais il existe l'alternative de construire une alimentation séparée simple en suivant le schéma ci-dessus, (figure 5b).

la fréquence de l'oscillateur intégré: il est convenablement ajusté lorsque la fréquence relevée sur la broche 3 d'IC1 est de 160 kHz. Il n'est pas nécessaire de disposer d'un fréquencemètre pour cela; il suffit que la voix émise soit normale (ni celle de Donald Duck, ni celle d'Ivan Rebroff). En principe, la position médiane de P1 devrait convenir. Il faut noter, toutefois, que ce réglage affecte la durée minimale des impulsions sur CCLK; les 6,25 µs que nous mentionnions ci-dessus correspondent à 160 kHz! L'entrée CCLK est synchronisée avec la sortie ROMCLK à la broche 3 par l'intermédiaire d'une bascule (FF1); le signal ainsi obtenu est appliqué via T4 à l'entrée PDC du VSP (IC1). La deuxième bascule associée à T5 rythme l'arrivée du flux de bits sur l'entrée ADD8 d'IC1, sous le contrôle de la sortie E/S (I/O). Les sorties "parole" (SPK1 et SPK2) sont appliquées à un filtre passe-bas (A1 et A2), puis à l'amplificateur de sortie (A3, T6 . . . T9) et c'est P2 qui détermine l'amplitude du signal de sortie.

6

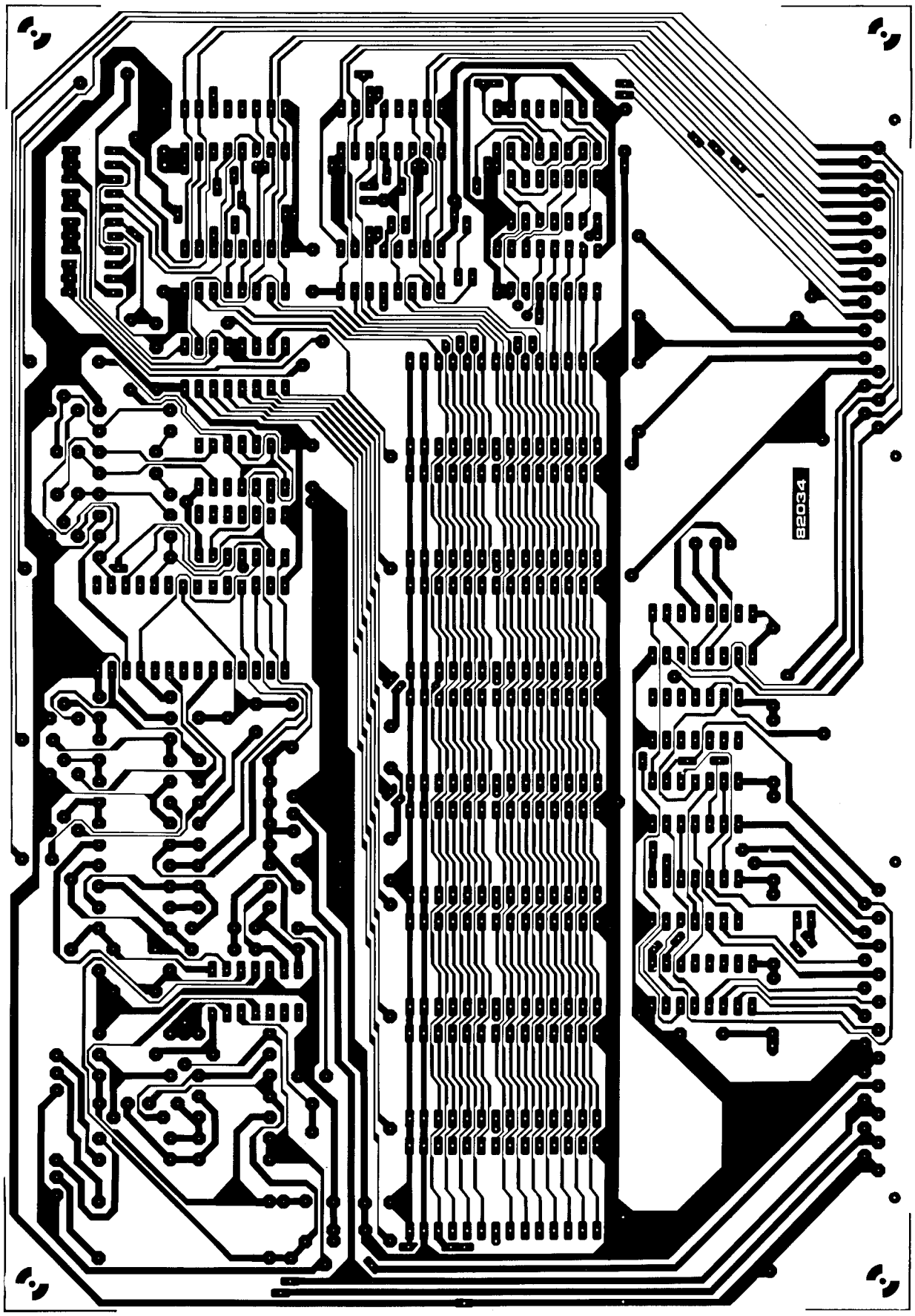

Figure 6. Représentation du circuit imprimé "qui-crache-la-parole"!

7'

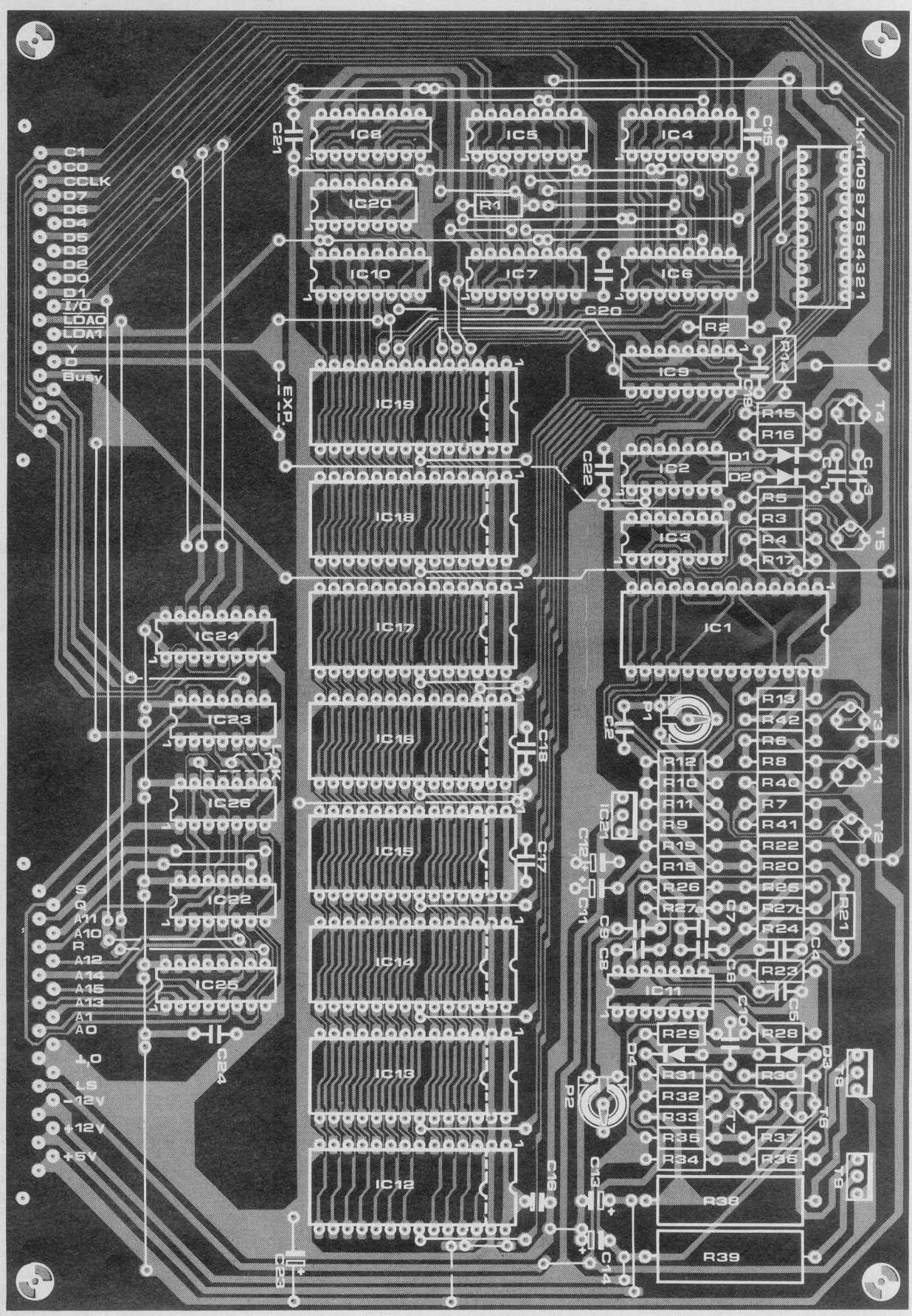

Figure 7. Implantation des composants.

Liste des composants

Résistances:
R1 . . . R3,R5 . . . R8,R14 . . . R16, R40 . . . R42 = 4k7
R4,R9 . . . R12,R17,R20 . . . R23,R25, R27a,R27b = 10 k
R13 = 22 k
R18,R19 = 47 Ω
R24 = 12 k
E26,R29 = 6k8
R28 = 1 k
R30,R31 = 8k2
R32,R33 = 2k2
R34,R35 = 82 k
R36,R37 = 22 Ω
R38,R39 = 2Ω2/3 W
P1 = 50 k ajustable
P2 = 22 k ajustable

Condensateurs:
C1,C3 = 100 p
C2 = 68 p
C4,C5,C8 = 1 n
C6,C7 = 10 n
C9,C15 . . . C22 = 100 n
C10 = 2n2
C11,C12 = 10 μ/16 V tantale
C13,C14 = 47 μ/16 V tantale
C23 = 47 μ/10 V

Semiconducteurs:
D1 . . . D4 = 1N4148
T1 . . . T5 = TUP
T6 = BC 183
T7 = BC 213
T8 = TIP 31
T9 = TIP 32
IC1 = TMS 5100
IC2 = 74LS74
IC3,IC20 = 74LS04
IC4 . . . IC8 = 74LS193
IC9 = 74LS138
IC10 = 74LS151
IC11 = TL 084
IC12 . . . IC19 = TMS 2532 *
IC21 = 7905
* voir texte

Liste des composants pour l'interface (figure 8)

Condensateurs:
C26 = 100 n

Semiconducteurs:
IC22 = 74LS02
IC23,IC24 = 74LS175
IC25 = 74LS138
IC26 = 74LS00

Liste des composants de l'alimentation (figure 5b)

Résistances:
R1 = 15 Ω/5 W

Condensateurs:
C1 = 4700 μ/40 V
C2 = 2200 μ/40 V
C3,C7 = 220 n
C4 = 2μ2/16 V tantale
C5,C6,C8 = 1 μ/16 V tantale

Semiconducteurs:
D1 . . . D4 = 1N4004
IC1 = 7812
IC2 = 7912
IC3 = 7805

Divers:
Tr1 = transfo secteur 2 x 15 V/1 A
radiateur pour IC3

La partie inférieure du circuit est occupée par la mémoire, avec sa logique de commande. IC4 . . . IC7 ne sont rien d'autre que le tampon/compteur d'adresses que nous avons déjà évoqué. Lorsque les entrées de chargement parallèle (broche 11) sont mises au niveau logique bas, via $\overline{LDA\ 0}$ et $\overline{LDA\ 1}$, l'octet du bus de données est transféré à la paire de circuits correspondante. Les sorties de ces circuits attaquent les entrées d'adresses d'IC12 . . . IC19 (les EPROM) et le sélecteur d'EPROM IC9. Le réseau de straps qui peut prêter à confusion, a été prévu pour que l'on puisse mettre en place différents types d'EPROM. Pour les 2716, il faut mettre en place les straps 2, 6, 7 et 9; les EPROM sont alors adressées dans l'ordre suivant. IC12, IC13, IC16, IC17, IC14, IC15, IC18, IC19; l'espace mémoire s'étend conséquemment de 0000 à 3FFF, en portions de 2 koctets.
Pour des 2732, comme sur le "moulin à paroles" standard, ce sont les straps 1, 6, 8 et 10 qu'il faut mettre en place. Les EPROM sont alors adressées séquentiellemtn d'IC12 à IC19, couvrant ainsi l'espace mémoire qui s'étend de 0000 à 7FFF. Les straps 1, 4, 8 et 11 seront mis en place avec des EPROM du type 2764; celles-ci couvrent tout l'espace adressable, de 0000 à FFFF, et sont adressées dans l'ordre suivant: IC12, IC14, IC16, IC18, IC13, IC15, IC17, IC19. Il faut observer que le dessin du circuit imprimé et le brochage sont destinés à recevoir des 2764; les autres types étant plus courts, on se réfèrera aux lignes pointillées; ceci signifie que la broche 1 d'une 2716 ou d'une 2732 correspondra à la broche 3 de l'emplacement prévu pour une 2764.
Enfin la partie inférieure à droite de la figure 4 reste à examiner. IC8 est le compteur de bits: le signal d'entrée $\overline{I/O}$ (horloge) est divisé par 8, pour sélectionner les 8 bits correspondant à un octet de la séquence. En fait, IC8 est un compteur 4 bits, mais on voit que le quatrième bit (Q_D) est appliqué à l'entrée "load", de sorte que l'on charge 0000 dès que cette sortie passe au niveau logique haut. Les trois bits de poids faible Q_A . . . Q_C commandent le multiplexeur de données IC10, dont la fonction est de choisir le bon bit dans l'octet de mémoire -vocabulaire. Chaque fois qu'un groupe de huit bits a été parcouru, une impulsion est envoyée sur IC8, via N4, et de là, à l'entrée de comptage d'IC7; ceci est destiné à incrémenter le compteur d'adresses.

Alimentation

Il n'y a que très peu à dire sur ce sujet. Le circuit principal comporte un nombre suffisant de condensateurs de découplage et de lissage comme on le voit sur la figure 5a, mais aussi un circuit intégré qui fournit les −5 V, à partir de la ligne −12 V qui lui est appliquée. De ce fait, il faudra donc pour alimenter ce circuit, les trois tensions suivantes, convenablement filtrées: +12 V/+5 V/0 V/−12 V. En règle générale ces tensions sont disponibles sur le système à microprocesseur avec lequel on utilisera le moulin à paroles. Dans le cas contraire, on utilisera le circuit que donne la figure 5b. Sous 5 V, il faut disposer de 300 mA. Le courant de repos absorbé par la ligne ± 12 V est de 50 mA, mais sera sensiblement supérieure si le niveau de sortie audio est élevé.

Synthèse vocale

Après avoir détaillé le circuit, il est temps à présent d'examiner le logiciel, et plus particulièrement la manière dont un mot donné est codé. Foncièrement, la structure du processeur vocal est identique à celle de l'organe phonatoire humain. C'est à dire qu'il procède d'une manière analogique à nos poumons, qui fournissent le souffle (énergie), aux cordes vocales qui produisent les sons (hauteur) et à la bouche, aux lèvres et à la langue qui modulent le timbre (filtre), lorsque les cordes vocales ne résonnent pas (sons dévoisés, comme S, F, etc) c'est un générateur de bruit qui est utilisé au lieu d'un générateur de fréquences discrètes. Toutes ces informations, pour un mot donné, sont contenues dans une séquence numérique de bits.
Prenons un exemple: le tableau 3 reproduit le code complet du mot "help". Le premier groupe de bits est 0000: silence. Puis 0100 produit l'énergie initiale; le bit de répétition est à zéro (nous y reviendrons) et la hauteur est donnée par 0000, ce qui signifie que le son en question est dévoisé. Pour un tel son dévoisé, les 18 bits suivants déterminent des paramètres de filtrage, comme on peut le voir. La ligne suivante commence avec un niveau d'énergie plus élevé (0111), suivi par le bit de répétition au niveau logique haut: les caractéristiques indiquées par les précédents paramètres de filtrage restent donc les mêmes! La hauteur reste 0000, c'est à dire que le son est toujours dévoisé. Les paramètres de filtrage étant inchangés, nous pouvons donc passer à la ligne suivante. Le niveau d'énergie augmente encore, il n'y a pas de répétition, et la hauteur est à

Tableau 4.

ADRESSE (HEX)	MOT	ADRESSE (HEX)	MOT	ADRESSE (HEX)	MOT
EPROM 1					
0000	AGAIN	0566	DO	0A78	THIS
0048	DOWN	05A0	DOES	0A9E	USE
0084	HELLO	05FC	FOR	0AF6	WHAT
00D0	MESSAGE	0634	FROM	0B20	WHEN
0138	MISTAKE	0662	GOT	0B6C	WHERE
0198	NAME	069C	HAVE	0BB4	WILL
01CE	NEED	06DA	HOW	0C06	WITH
0222	PLEASE	0724	IN	0C5A	WOULD
0262	PUT	0760	IS	0C94	YES
02BC	REPEAT	079C	IT	0CC6	YOU
02CC	RIGHT	07B4	ME	0CF8	YOUR
0324	THANK	0800	MUCH	0D2E	A
036E	UP	082A	MY	0D6C	B
0388	WANT	0856	NO	0DA8	C
03CE	'S	0890	NOT	0DEC	D
03E4	ALL	08C4	NOW	0E36	E
041A	AN	0906	OF	0E60	F
0446	AND	0946	ON	0E94	G
0484	ANY	0970	OR	0EC4	H
04B4	ARE	099A	OUT	0EFE	I
04D0	AT	09D6	THE (E)	0F34	J
04F2	CAN	0A08	THE	0F80	K
0522	DID	0A44	THERE		
EPROM 2					
0000	L	0474	FOUR	09EC	FIFTY
004C	M	04C2	FIVE	0A46	SIXTY
008A	N	0510	SIX	0A7E	SEVENTY
00C2	O	054E	SEVEN	0AC4	EIGHTY
00EA	P	05A4	EIGHT	0AF0	NINETY
0114	Q	05D4	NINE	0B58	HUNDRED
014C	R	061E	TEN	0BC2	THOUSAND
0178	S	0652	ELEVEN	0C3E	EQUAL
01A0	T	069A	TWELVE	0C94	NUMBER
01F0	U	06F6	THIRTEEN	0D04	PERCENT
021E	V	0732	FOURTEEN	0D54	AMPS
0250	W	0774	FIFTEEN	0D94	DEGREES
0298	X	0800	SIXTEEN	0DF4	FARAD
02BE	Y	0864	SEVENTEEN	0E62	FREQUENCY
0300	ZED	08C2	EIGHTEEN	0ECE	HENRY
0346	ZERO	08FC	NINETEEN	0F18	HERTZ
03A4	ONE	0952	TWENTY	0F66	HOURS
03F6	TWO	0986	THIRTY		
0430	THREE	09B6	FORTY		
EPROM 3					
0000	MEGA	0530	CHECK	0ADE	DIRECTION
005E	MICRO	0566	BUTTON	0B5C	ENTER
00D2	MILLI	05B6	TELEPHONE	0BA6	FAST
010A	MINUS	0608	BUSY	0C00	SLOW
0172	OHMS	0656	INVALID	0C48	GO
01C6	PLUS	06F6	MONDAY	0C9C	STOP
01FA	POINT	0752	TUESDAY	0CEA	HIGH
023C	POWER	0800	WEDNESDAY	0D46	LOW
0282	SECONDS	0872	THURSDAY	0D9A	MOVE
02EA	TEMPERATURE	08D2	FRIDAY	0E10	RANGE
0362	TIME	0938	SATURDAY	0E7E	EXIT
03AC	READY	0984	SUNDAY	0EBE	CARDS
03F0	SWITCH	09DE	GOOD BYE	0F10	ATTACK
043E	CONTROL	0A36	DATE	0F4C	DESTROY
04A2	WARNING	0A6A	LEFT		
04EC	OFF	0A98	CHANGE		
EPROM 4					
0000	MERCI	03D2	NEUF	0A38	CINQUANTE
0078	ET	0445	DIX	0ADF	SOIXANTE
009A	ZERO	04A0	ONZE	0B82	CENT
00FE	UN	0540	DOUZE	0BC1	MILLE
0126	DEUX	05E8	TREIZE	0C62	HEURE
016A	TROIS	0687	QUARTORZE	0CF0	MOINS
01B5	QUATRE	0738	QUINZE	0E10	PLUS
0242	CINQ	07EE	SEIZE	0D49	SECONDES
02C4	SIX	098A	VINGT	0E71	AU REVOIR
0328	SEPT	08DE	TRENTE		
0369	HUIT	0994	QUARANTE		

** A noter que le premier chiffre de l'adresse du mot doit correspondre à la position de l'EPROM. Si elles sont montées séquentiellement dans les positions des IC12 . . . IC15, la première adresse de l'EPROM 2 sera 1000, l'EPROM 3 commencera à 2000, et l'EPROM 4 débutera à 3000.*

Tableau 4. Voici le vocabulaire contenu dans les diverses EPROM du "moulin à paroles".

présent définie par 10010, qui produit une certaine fréquence sonore. Pour les sons voisés, les paramètres de filtrage requis sont nettement plus précis; soit 39 bits pour les différentes courbes de filtrage de 10 filtres en tout. Heureusement, ces paramètres restent inchangés dans les deux lignes suivantes (bit de répétition) tandis que le niveau d'énergie et la hauteur croissent. Et ainsi de suite.

Le principe de base est assez clair. Lorsque l'on interroge un mot donné (dans l'intention d'en faire un autre mot par exemple!), les règles à suivre sont les suivantes:

— si les quatre premiers bits d'une ligne sont à zéro, oubliez-les; il s'agit d'un silence;

— dans le cas contraire, examinez le bit suivant (bit de répétition); s'il est au niveau logique bas, il sera suivi de paramètres de filtrage; s'il est au niveau logique haut, les cinq bits suivants sont destinés à déterminer la hauteur du son;

— si les bits de hauteur sont tous à zéro, c'est qu'il s'agit d'un son dévoisé, et les 18 bits suivants correspondent à des paramètres de filtrage. Pour les sons voisés (bits de hauteur différents de zéro), ce sont les 39 bits suivants qui déterminent les paramètres de filtrage;

— si les quatre permiers bits d'une ligne sont 1111, cela signifie que le mot est terminé.

Avec ces quelques informations, il est possible de décoder n'importe quel mot, et de là, d'en reconstituer d'autres par tâtonnements successifs, en modifiant le code existant. C'est ainis que nous avons tenté de "construire" le mot Elektor (qui ne figure pas dans le vocabulaire standard) et nous avons été passablement satisfaits du résultat. Un vocabulaire de base est indispensable, et de ce fait, disponible sur EPROM. Les mots qu'il contient figurent sur le tableau 4.

Réalisation et mise en oeuvre

Le dessin du circuit imprimé, avec sa face sérigraphiée pour l'implantation des composants, est reproduit sur les figures 6 et 7. Il faudra commencer par l'implantation des straps (EXP inclus) à l'exception de L ou K, dont nous reparlerons. Si les niveaux de sortie exigés sont élevés, il est préférable de munir T8 et T9 d'un petit radiateur. Outre les composants du circuit principal, il a également été prévu la mise en place des composants pour une interface compatible avec la plupart des microprocesseurs courants (IC22 . . . 26 et IC24). La connexion pourra être faite via un connecteur à 21 broches (DIN 41617) mâle, à broches coudées à 90°.

Tout système à microprocesseur doté de 14 lignes d'entrée/sortie au moins pourra être relié à cette interface; ces quatorze lignes apparaissent à gauche de la figure 4. Les lignes D, $\overline{I/O}$ et Y ne sont pas utilisées dans un premier temps. On s'en servira ultérieurement, lorsque l'on cherchera à modifier le code des

8

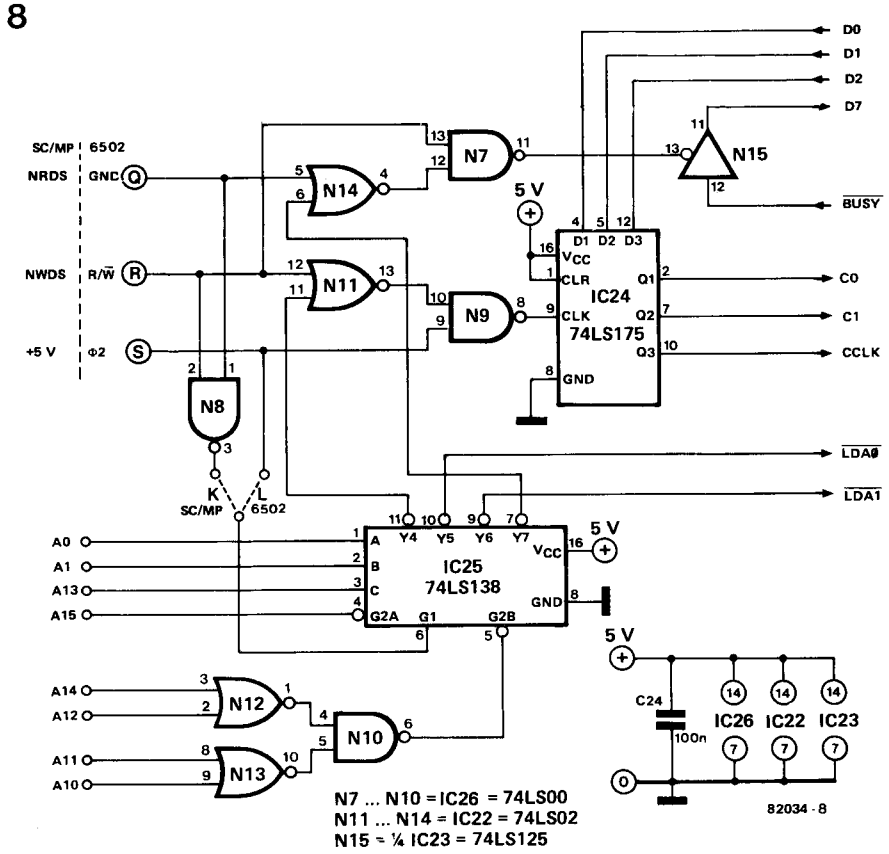

Figure 8. Si l'ordinateur hôte ne possède pas de lignes d'E/S (entrée/sortie) disponibles, il faudra ajouter cette interface simple.

9

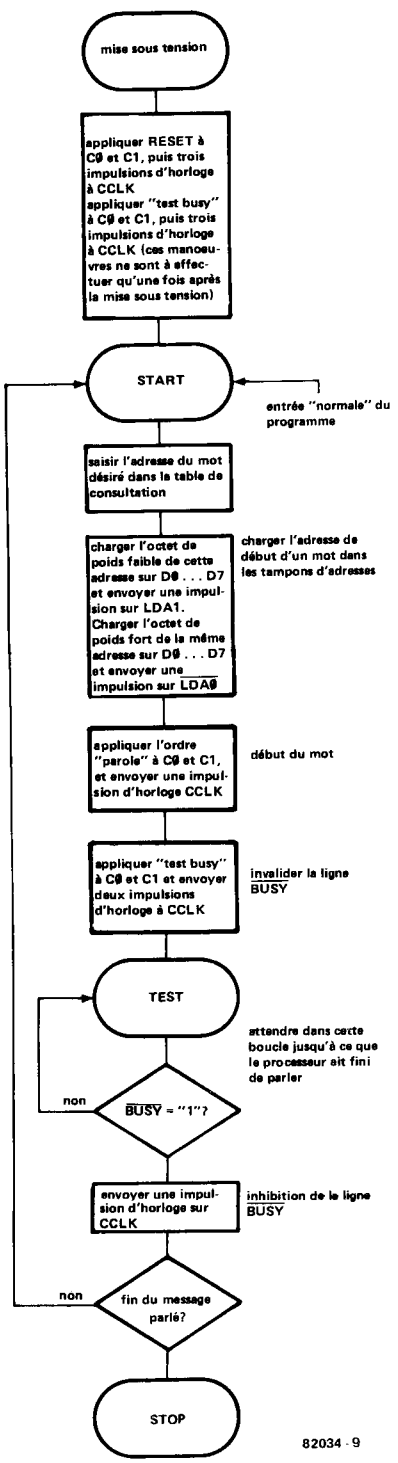

Figure 9. Cet ordinogramme montre quels sont les différents pas que doit effectuer le système pour produire un message parlé.

mots à prononcer afin de constituer un vocabulaire personnel (nous publierons un circuit conçu à cet effet). Il se peut que pour certains systèmes il faille adjoindre le circuit supplémentaire, tel qu'il est donné par la figure 8. On remarquera que ce circuit pourra lui aussi être monté sur la même plaque que le reste, mais ses composants ne sont pas pris en compte dans la liste de ceux du kit de base. Les lignes qui apparaissent à gauche de la figure 8 devront être reliées aux lignes correspondantes du système à microprocesseur concerné. Il faudra bien sûr relier également les lignes DØ . . . D7 au bus de données du microprocesseur.

Le décodage d'adresse n'est pas très sophistiqué; le circuit occupe complètement le bloc de mémoire adressable qui s'étend de 2000 à 23FF, pour quatre adresses seulement! En cas de nécessité, ce décodage pourra être modifié afin de réduire cet espace utilisé au strict minimum. Il suffit en fait de disposer des quatre adresses suivantes:

- données pour CØ, et C1 et CCLK: ici c'est l'adresse 2000 qui est utilisée. Bit Ø, 1 = CØ,C1; bit 2 = CCLK
- signal $\overline{\text{LDA 1}}$: adresse 2002 (donnée = adresse/octet de poids faible)
- signal $\overline{\text{LDA Ø}}$: adresse 2001 (donnée = adresse/octet de poids fort)
- signal $\overline{\text{BUSY}}$: adresse 2003, bit 7 -bit de poids fort-

L'entrée G1 d'IC25 sera connectée différemment selon le système à microprocesseur utilisé. Pour le Junior Computer, il faudra la relier à Ø2 (strap L); pour le SC/MP, c'est à une combinaison de NRDS et NWDS (strap K). En règle générale ce signal indique le moment où les adresses et les données sont valides.

Lorsque l'on dispose d'une interface convenable, il n'est pas bien difficile d'obtenir un son parlé sur la sortie audio du circuit. L'organigramme du programme utilisé à cette fin est donné par la figure 9. Après la mise sous tension, il faut initialiser le processeur vocal. C'est ce que l'on fait en chargeant successivement les données Ø7 - Ø3 - Ø7 - Ø3 - Ø7 - Ø3 - &7, à l'adresse 2000. Le résultat ainsi obtenu est que CØ et C1 sont mis au niveau logique haut, tandis que CCLK voit son niveau basculer trois fois. Nous avons déjà indiqué que la durée de l'impulsion (lorsque le niveau logique est inversé) sur CCLK (bit 2 dans la séquence) doit être de 6,25 µs au moins. L'adjonction d'une boucle de temporisation peut donc se justifier. L'étape suivante consiste à charger alternativement ØØ et Ø4 à l'adresse 2000, et ceci trois fois de suite.

C'est ainsi que nous arrivons au label "START"; c'est maintenant que commence le programme de lancement d'une séquence de synthèse proprement dite, le reste n'étant que préparatifs. C'est d'abord l'octet de poids faible de l'adresse du mot à prononcer que l'on charge à l'adresse 2002 (cette opération provoque automatiquement l'indispensable impulsion sur $\overline{\text{LDA 1}}$); puis c'est l'octet de poids fort qui est chargé à l'adresse 2001. A présent, l'ordre "parole" peut être donné (Ø2 - Ø6 à l'adresse 2000). Pour finir, la séquence de données ØØ - Ø4 - ØØ - Ø4 est appliquée à l'adresse 2000, à raison de 6,25 µs au moins par donnée. Cette dernière manoeuvre correspond à l'application du "test busy" et de l'inversion (deux fois de suite) du niveau logique de CCLK.

Nous en arrivons à une boucle d'attente dont on ne sort que lorsque la sortie

Tableau 5.

```
                                JUNIOR'S ASSEMBLER          PAGE 01
0010: 0200                         ORG   $0200
0020:
0030:                     DATE :  29-9-'81
0040:
0050:
0060:                     SPEECH SYNTHESISER  TMS 5100
0070:
0080:                     INTERFACE ADDRESSING
0090:
0100: 0200                CMND   *     $2000  COMMAND ADDRESS
0110: 0200                LDZERO *     $2001  LDA0 STROBE ADDRESS
0120: 0200                LDONE  *     $2002  LDA1 STROBE ADDRESS
0130: 0200                BUSY   *     $2003  BUSY READ OUT
0140:
0150:                     JUNIOR MONITOR START ADDRESS
0160:
0170: 0200                RESET  *     $1C1D
0180:
0190:                     SPEECH ADDRESS LOOK UP TABLE
0200:
0210: 0200                TABLE  *     $0400  LOWER ORDER ADDRESS BYTE FIRST WORD
0220:                                  $0401  HIGHER  "      "    "    "    "
0230:                                  $0402  LOWER ORDER ADDRESS BYTE SECOND WORD
0240:                                  $0403  HIGHER  "      "    "    "    "
0250:                                  .....
0260:                                  $04FF  HIGHER ORDER ADDRESS BYTE LAST WORD
0270:
0280:                     ***********
0290:                     MAINPROGRAM
0300:                     ***********
0310:
0320: 0200 A9 07          POWUP  LDAIM $07
0330: 0202 8D 00 20              STA   CMND   SET 'RESET' COMMAND ON C0 AND C1
0340: 0205 20 45 02              JSR   TOGGLE TOGGLE CCLK THREE TIMES
0350: 0208 20 45 02              JSR   TOGGLE
0360: 020B 20 45 02              JSR   TOGGLE
0370: 020E A9 00                 LDAIM $00
0380: 0210 20 45 02              JSR   TOGGLE SET 'TEST BUSY' COMMAND ON C0 AND C1, AND
0390: 0213 20 45 02              JSR   TOGGLE TOGGLE CCLK THREE TIMES
0400: 0216 20 45 02              JSR   TOGGLE
0410:
0420: 0219 A2 00          START  LDXIM $00    CLEAR X-REGISTER
0430:
0440: 021B BD 00 04       STRT   LDAX  TABLE  LOWER ORDER SPEECH START ADDRESS TO ACCU
0450: 021E 8D 02 20              STA   LDONE  SET D0 TO D7 TO THIS BYTE AND STROBE LDA1
0460: 0221 E8                    INX
0470: 0222 BD 00 04              LDAX  TABLE  HIGHER ORDER SPEECH START ADDRESS TO ACCU
0480: 0225 8D 01 20              STA   LDZERO SET D0 TO D7 TO THIS BYTE AND STROBE LDA0
0490: 0228 A9 02                 LDAIM $02
0500: 022A 20 45 02              JSR   TOGGLE SET 'TALK' COMMAND ON C0 AND C1, AND
0510:                                         TOGGLE CCLK ONCE
0520: 022D A9 00                 LDAIM $00
0530: 022F 20 45 02              JSR   TOGGLE SET 'TEST BUSY' COMMAND ON C0 AND C1, AND
0540: 0232 20 45 02              JSR   TOGGLE TOGGLE CCLK TWICE
0550:
0560:
          JUNIOR'S ASSEMBLER            PAGE 02


0570: 0235 AD 03 20       TEST   LDA   BUSY   READ BUSY LINE
0580: 0238 10 FB                 BPL   TEST   BUSY LINE NOT HIGH?
0590: 023A 20 45 02              JSR   TOGGLE TOGGLE CCLK ONCE
0600: 023D E8                    INX
0610: 023E E0 08                 CPXIM $08    COMPARE X WITH THE NUMBER OF WORDS MULTIPLIED
0620:                                         BY TWO ( IN THIS CASE THE NUMBER
0630:                                         OF WORDS IS FOUR )
0640: 0240 D0 D9                 BNE   STRT   WORD SEQUENCE NOT COMPLETED?
0650: 0242 4C 1D 1C              JMP   RESET  RETURN TO JUNIOR MONITOR
0660:
0670:                     **********
0680:                     SUBROUTINE
0690:                     **********
0700:
0710: 0245 29 03          TOGGLE ANDIM $03    SET CCLK TO ZERO AND
0720: 0247 8D 00 20              STA   CMND   TRANSFER ACCUMULATOR B0 AND B1 TO C0 AND C1 LINE
0730: 024A 09 04                 ORAIM $04
0740: 024C EA                    NOP          DELAY TWO MICRO SECONDS ( CCLK 'LOW' TIME MUST
0750:                                         BE AT LEAST 6.25 MICRO SEC. )
0760: 024D 8D 00 20              STA   CMND   SET CCLK TO ONE
0770: 0250 60                    RTS
ID=
```

Tableau 5. Ce programme donne une langue au Junior Computer!

"busy" passe au niveau logique haut ("busy" = MSB à l'adresse 2003). Une nouvelle séquence de données 00 - 04 est alors chargée à l'adresse 2000 afin d'inhiber la sortie "busy". Si à présent il faut "parler" un autre mot, la séquence est reprise depuis start. Le tableau 5 reproduit un programme complet qui a été écrit pour le Junior Computer.

Disponibilité des composants

Les conditions dans lesquelles on pourra trouver tout le matériel nécessaire à la réalisation du "moulin à paroles" sont particulières. Comme nous ne doutons pas de l'intérêt plutôt vif que ne manquera pas de susciter cette innovation, nous demandons à nos lecteurs de se renseigner soigneusement auprès de leur revendeur habituel et d'accorder une attention exceptionnelle à leurs annonces.

Dans l'immédiat, nous proposons un circuit de synthèse vocale aux performances remarquables; mais notez-bien que notre intention réelle va plus loin: notre prétention est de vous offrir un outil de travail, qui vous permette de constituer vous-même votre propre vocabulaire. Ceci sera facilement réalisable dès que vous disposerez de l'interface que nous publierons prochainement, associée à un système à microprocesseur, tel le SC/MP ou le Junior Computer . . . ou tout autre!