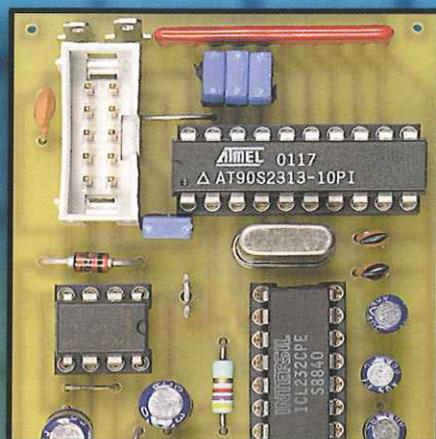


# GÉNÉRATEUR de fonctions

*Production de plusieurs  
formes d'ondes  
sous une fréquence  
de 0,4 Hz à 100 kHz.*



- ▶ *Simulateur  
de carte SIM*
- ▶ *Carte  
alimentation  
intégrée*
- ▶ *Nouveautés  
Cartes 2001*
- ▶ *Programmateur  
24C16*
- etc.*

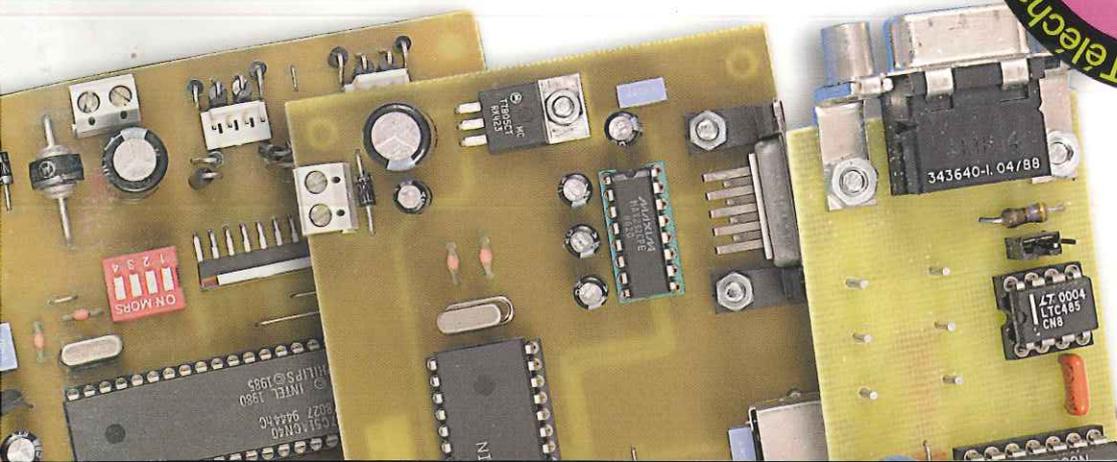
**COMMANDE  
MOTEURS  
PAS À PAS**

**INTERFACE  
RS232  
POUR CLAVIER**

**CHAT EN  
RS 485**



**Téléchargement:**  
TOUS LES  
PROGRAMMES  
ET LES PCB,  
SUR INTERNET  
[www.eprat.com](http://www.eprat.com)



T 3271 - 11 H - 32,80 F - 5,00 € - RD-



FRANCE: 32,80 FF/5€ - DOM: 30FF/4,57€ - BEL: 180FB/4,46€  
CH: 7FS - CAN: 7,50S CAN - ESP: 490 PTA/2,95€  
GR: 1700 GRD/4,99€ - LUX: 180 LUF/4,46€ - MAR: 50 DH  
PORT: 1150 ESC/5,74€

# SOMMAIRE

## ELECTRONIQUE PRATIQUE

HS N°11 - DÉCEMBRE 2001  
I.S.S.N. 0243 4911

PUBLICATIONS GEORGES VENTILLARD

S.A. au capital de 5 160 000 F  
2 à 12, rue Bellevue, 75019 PARIS  
Tél. : 01.44.84.84.84 - Fax : 01.44.84.85.45  
Internet : <http://www.eprat.com>  
Principaux actionnaires :  
M. Jean-Pierre VENTILLARD  
Mme Paule VENTILLARD

Président du conseil d'administration,  
Directeur de la publication : Paule VENTILLARD  
Vice-Président : Jean-Pierre VENTILLARD  
Attaché de Direction : Georges-Antoine VENTILLARD  
Directeur de la rédaction : Bernard FIGHERA (84.65)  
Directeur graphique : Jacques MATON  
Maquette : Dominique DUMAS, Jean-Pierre Rafini

Avec la participation de : U. Bouteville, H. Cadinot,  
X. Fenard, A. Garrigou, P. Gueulle, M. Laury,  
Y. Leidwanger, J.P. Mandon, P. Morin, P. Oguic,  
K. Pham-Dinh, J.P. Pitollet, Ch. Tavernier.

La Rédaction d'Electronique Pratique décline toute responsabilité  
quant aux opinions formulées dans les articles, celles-ci n'enga-  
gent que leurs auteurs.

Directeur de la diffusion et promotion :  
Bertrand DESROCHE  
Responsable ventes :  
Bénédicte MOULET Tél. : 01.44.84.84.54  
N° vert réservé aux diffuseurs et dépositaires de presse :  
0800.06.45.12

PGV - Département Publicité :  
2 à 12 rue de Bellevue, 75019 PARIS  
Tél. : 01.44.84.84.85 - CCP Paris 3793-60  
Directeur commercial : Jean-Pierre REITER (84.87)  
Chef de publicité : Pascal DECLERCK (84.92)  
E Mail : [lehp@e-hp.com](mailto:lehp@e-hp.com)  
Assisté de : Karine JEUFFRAULT (84.57)

Abonnement/VPC: Voir nos tarifs en page intérieure.

Préciser sur l'enveloppe «SERVICE ABONNEMENTS»

**Important** : Ne pas mentionner notre numéro de compte  
pour les paiements par chèque postal. Les règlements en  
espèces par courrier sont strictement interdits.

ATTENTION ! Si vous êtes déjà abonné, vous faciliterez notre  
tâche en joignant à votre règlement soit l'une de vos dernières  
bandes-adresses, soit le relevé des indications qui y figurent.

• Pour tout changement d'adresse, joindre 3, 00 F et la der-  
nière bande.

Aucun règlement en timbre poste.

Forfait photocopies par article : 30 F.

Distribué par : TRANSPORTS PRESSE

Abonnements USA - Canada : Pour vous abonner à  
Electronique Pratique aux USA ou au Canada, commu-  
niquez avec Express Mag par téléphone :

USA : P.O.Box 2769 Plattsburgh, N.Y. 12901-0239  
CANADA : 4011boul.Robert, Montréal, Québec, H1Z4H6  
Téléphone : 1 800 363-1310 ou (514) 374-9811  
Télécopie : (514) 374-9684.

Le tarif d'abonnement annuel (9 numéros) pour les USA  
est de 41 \$US et de 68 \$cnd pour le Canada.

Electronique Pratique, ISSN number 0243 4911, is published  
9 issues per year by Publications Ventillard at P.O. Box 2769  
Plattsburgh, N.Y. 12901-0239 for 41 \$US per year.

POSTMASTER : Send address changes to Electronique Pratique,  
c/o Express Mag, P.O. Box 2769, Plattsburgh, N.Y., 12901-0239.



CPPAP n° 60165  
Imprimerie STIGE  
v. Pescarito 110  
10099 San Mauro - Italie

« Ce numéro  
a été tiré  
à 48 200  
exemplaires »

## INTERFACES PC

- 05 Edito
- 06 Nouveautés Cartes 2001
- 11 Utilisation et commande du CDRom
- 19 Plinius
- 20 Basic SIM : simulateur de carte SIM
- 80 Les cartes mères PC ASUS
- 88 Nouvelle gamme AUDIGY



## Les cartes à réaliser

- 14 Carte alimentation intégrée à un PC
- 24 Télécommande à courants porteurs sur port //
- 32 Interface RS232 pour clavier PC
- 38 Radiocommande 4 canaux simultanés pilotée par PC
- 44 Terminal RISC
- 50 Chat en RS485
- 54 Programmeur de 24C16 portable
- 58 Programmeur en circuit pour µC AVR de ATMEL
- 64 Générateur de fonctions sur port série
- 70 Transformation d'un port série en port //
- 76 Analyseur de port parallèle
- 82 Kit-51 : module programmable à base de 89C51RD+
- 90 Commande de moteurs pas à pas par Bus CAN

## Infos OPPORTUNITÉS

10

La sortie d'Interfaces PC s'avère toujours un événement très attendu de nos fidèles lecteurs. Plus que jamais, votre magazine se doit de vous proposer un contenu riche et varié.

Dans cet esprit, le présent numéro devrait combler les passionnés de développement grâce au simulateur de carte SIM et au kit pour microcontrôleur 89C51RD+. Au chapitre de la domotique, vous trouverez également les télécommandes de votre choix.



Si vous avez maintenant une préférence pour le bus CAN, vous automatiserez votre maison par le biais de la commande de moteurs pas à pas. Mesures et PC font bon ménage, preuve s'il en fallait la description d'un générateur de fonctions aux multiples formes d'ondes sous une large plage de fréquences.

Bien sûr, ce n'est là qu'un bref aperçu du contenu de ce numéro d'Interfaces PC que vous êtes sans doute impatient de découvrir. Aussi, nous laissons maintenant la place à nos auteurs et nous vous souhaitons une bonne lecture.

Mais avant de vous laisser découvrir enfin les pages de ce nouveau numéro d'Interfaces PC, nous profitons de l'occasion qui nous est offerte pour vous souhaiter, au nom de toute la rédaction, de très bonnes fêtes de fin d'année.

**La rédaction**

Nouveautés

## CARTES 2001



Avec près de 350 exposants répartis sur les trois niveaux du CNIT, le salon CARTES est incontestablement le temps fort annuel de toute la profession de la carte à puce. Pour quelque 13500 visiteurs du monde entier, ces 20000m<sup>2</sup> d'exposition constituent une opportunité incontournable pour apprécier des nouveautés qui se mesurent aussi bien en termes d'innovation pure, que d'amélioration de produits existants.

## La BasicCard en vedette !

Annoncée en avant-première dans le n°9, la BasicCard "Professional" était en vedette sur le stand VCT (Versatile Card Technology, distributeur américain de ZeitControl).

Il nous a été particulièrement agréable d'y apprendre, qu'en France, les ventes de kits BasicCard n'ont véritablement démarré qu'à partir de la parution de notre article du n°8 d'INTERFACES PC et sont donc essentiellement le fait de nos fidèles lecteurs ! Sur le CD-ROM de ce numéro, ceux-ci trouveront la mise à jour "version 4" de leur kit, indispensable pour mettre en œuvre la ZC4.1, aujourd'hui disponible au détail chez notre annonceur Hi Tech Tools.

Bénéficiant d'améliorations décisives, à commencer bien sûr par la prise en charge du protocole T=0, ce nouvel environnement de développement (gratuit !) est bien évidemment utilisable avec les BasicCards des générations précédentes (ZC1.1, ZC3.3, ZC3.9, etc.) et avec les actuels lecteurs "CyberMouse" ACR20S. Sur le stand de leur fabricant ACS (Advanced Card Systems), il était toutefois intéressant de suivre l'évolution de

cette gamme de lecteurs

"made in Hong Kong", économiques mais performants.

Le modèle ACR30, par exemple, semble appelé à remplacer progressivement l'ACR20, y compris dans les kits BasicCard, dont la version 4 supporte ce nouveau "best-seller". Il diffère surtout de l'ACR20 par une densité d'intégration supérieure (un seul circuit intégré), simplification obtenue en renonçant à supporter certaines cartes synchrones proches de l'obsolescence.

## L'USB dans tous ses états

Mais la tendance lourde est une montée en puissance des lecteurs à brancher sur un port USB. Il existait déjà des versions USB des ACR20 et ACR30, naturellement compatibles PC/SC, mais la grande nouveauté 2001 est le "SIMTracker" ACT30. Destiné aux cartes au format "SIM micro", il tient entièrement dans le capot d'une fiche USB (dimensions 52x20x10mm) ! Bien qu'il se prête admirablement à une foule d'applications originales comme la confection de dongles avec, pourquoi pas, une BasicCard découpée, il est avant tout

destiné aux cartes SIM

des télé-phones portables.

Fidèle à sa stratégie de mise sur le marché grand public mondial de solutions "clef en main", ACS a d'ailleurs composé un kit "SIMmate Mini" rassemblant un SIMTracker et un logiciel dérivé de SIMmate 2000, que nous avons déjà eu l'occasion de présenter à nos lecteurs.

Quelques-uns d'entre eux ont eu la chance de pouvoir acquérir, sur le salon, ces différents produits avant même leur distribution effective en France.

Il n'est pas douteux que ce concept de lecteur de cartes à puce intégré dans une fiche USB se taillera un franc succès, un autre fabricant s'étant d'ailleurs lui aussi engagé dans cette voie (l'Allemand OMNIKEY, dont le CardMan 6020 a été primé au "Smart Card Awards 2001"). D'une façon générale, quasiment tous les fabricants de lecteurs se sont enfin ralliés au standard PC/SC, ce qui va assurément dans le bon sens. On attend juste de pouvoir s'en procurer



aussi facilement, et à aussi bon compte, que des souris ou des claviers (en Allemagne, c'est le cas depuis longtemps déjà). Reste que des progrès techniques sont encore à faire, un certain consensus semblant se dégager selon lequel aucun modèle de lecteur PC/SC, certifié ou non, ne pourrait être considéré comme parfaitement conforme. Et, de fait, d'un lecteur PC/SC à un autre, un même logiciel ou une même carte se comporte parfois assez différemment. Pire encore, il peut même suffire de mettre à jour un driver (ce qui est censé améliorer les choses !) pour qu'une application ou une carte ne fonctionne plus correctement.

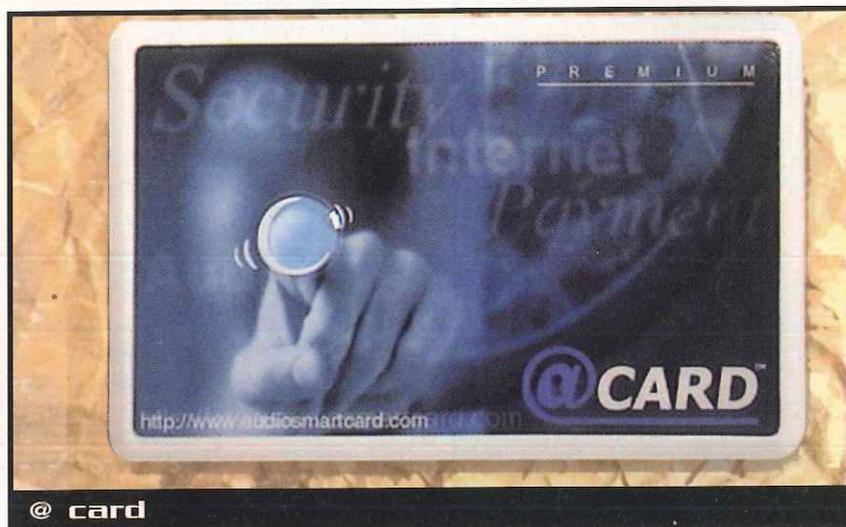
Cela a notamment été remarqué avec les cartes bancaires françaises, dont la plupart des fabricants de lecteurs ignorent superbement certaines particularités, pour ne pas

dire certains archaïsmes.

Une autre source de problèmes, souvent pointée du doigt, est la cohabitation sur un même PC de plusieurs lecteurs PC/SC de marques différentes, leurs drivers respectifs entrant volontiers en conflit.

### Cartes à puce et biométrie

Plusieurs fabricants abordent cette année le domaine des lecteurs de cartes à puce avec capteur biométrique incorporé. La reconnaissance d'empreintes digitales est désormais une technique fiable, rapide et bien acceptée, car ne présentant à l'évidence aucun danger. ACS a ainsi développé sa "BioSIMkey" ADT60 autour du composant "TouchChip" de STMicroelectronics, qui offre une résolution très suffisante de 508 DPI.



L'association des deux techniques sécuritaires que sont la biométrie et la carte à puce permet enfin d'envisager sereinement des applications telles que le vote à distance, sans même parler du commerce électronique et de la banque à domicile. Là encore, OMNIKEY propose un produit comparable, quoique basé sur un lecteur de cartes à puce au plein format ISO. Le composant "Fingertip" de INFINEON (513 DPI) se retrouve d'ailleurs dans un nombre toujours croissant de produits, intégrant ou non un lecteur de cartes à puce.

### Des cartes à puce à couplage acoustique

Il y a environ trois ans, la société ELVA présentait à ce même salon CARTES sa technologie "VocalID" : une carte à puce d'apparence parfaitement normale, mais qui émet un son codé (en FSK, comme un modem) lorsque l'on pose le doigt sur ses contacts. C'était une réelle prouesse que d'intégrer ainsi, dans 0,76 mm d'épaisseur, une pile (l'équivalent "sensitif" d'un bouton-poussoir) et un élément reproducteur de son "VICI" (Vibrating ISO Card Interface). Composée d'un identifiant et d'un cryptogramme, tous deux longs de 64 bits, la séquence audio de moins d'une seconde peut, aussi bien être transmise par téléphone que captée par le micro d'un PC multimédia, s'affranchissant ainsi de tout lecteur de cartes à puce.

A ce jour, ce sont les applications de banque à domicile qui prédominent, peut-être parce que rien ne s'opposerait, techniquement parlant, à équiper de cette fonctionnalité des cartes bancaires par ailleurs tout à fait conventionnelles, avec puce et/ou pistes magnétiques.

La fabrication peut, en effet, se faire sur les lignes de production de cartes à puce ordinaires. Une véritable "convergence" de technologies, donc, qui confère à VocalID tous les attributs d'un futur standard de l'authentification en ligne. Mais à l'occasion de l'édition 2001 du salon CARTES, la société AudioSmartCard opérait le lancement du kit "Secured Sound", combinant un produit fonctionnellement comparable et toute une suite logicielle pour PC : cryptage de fichiers, protection de l'accès à Windows, authentification sur Internet avec ou sans intervention d'un tiers de confiance, etc.



Matériellement parlant, "@CARD" est réalisée à partir

**SIM mate mini**

d'une technologie beaucoup plus traditionnelle que nous pourrions qualifier de "couche épaisse". Il est vrai que, en l'absence de toute contrainte d'insertion dans un quelconque lecteur, autant profiter d'une épaisseur plus généreuse pour intégrer une pile plus forte (17000 utilisations au lieu de 3000), un haut-parleur plus puissant et un bouton-poussoir mécanique. A la limite, rien n'imposait même le format "carte", un porte-clefs ou un médaillon pouvant rendre exactement les mêmes services.

Si ce n'est le codage audio adopté (FSK au lieu de DTMF) et les fonctionnalités cryptographiques, le produit fait penser aux cartes d'appel de certains opérateurs téléphoniques comme Kertel, qui sortent d'ailleurs bel et bien "du même tonneau" (une usine bretonne de câblage électronique).

Par rapport à une carte à puce avec ou sans contact, la grosse différence de protocole est que la communication est unidirectionnelle (carte vers terminal), ce qui exclut d'emblée tout mécanisme "ques-

tion/réponse". Compte tenu de la facilité avec laquelle une séquence audio peut être interceptée, enregistrée et rejouée, toute la sécurité d'un tel système repose sur sa cryptographie. Il est essentiel que chaque cryptogramme soit unique, ne serve qu'une fois et que sa réception "périme" irréversiblement tous ceux qui auraient pu être créés auparavant, qu'ils aient été reconnus ou non. La similitude est frappante avec les systèmes à codage dynamique ou "rolling code", largement éprouvés dans les applications automobiles.

**Un absent de taille**

Avec déjà seize ans d'ancienneté, le salon CARTES jouit d'une telle renommée que l'on se fait peut-être davantage remarquer en cessant d'y exposer qu'en y étant présent chaque année !

Tel est le cas de Microsoft, qui n'avait pas ménagé ses efforts, depuis deux ans, pour promouvoir "Windows for Smart Cards" et son environnement Visual Basic. Considéré un moment comme susceptible de s'imposer comme LE système d'exploitation "ouvert" pour cartes à puce, face à JavaCard et Multos, ce projet pourtant déjà bien avancé semble aujourd'hui purement et simplement enterré.

Mais alors, faut-il craindre un éventuel abandon du support des cartes à puce et, en particulier, des lecteurs PC/SC, dans les prochaines versions de Windows ? Bien au contraire ! La carte à puce est de plus en

plus intimement intégrée au système et, d'ailleurs, Linux n'est plus très loin derrière. Simplement, développer des systèmes d'exploitation pour les PC et pour les cartes à puce sont deux métiers radicalement différents.

Dans le premier cas, on dispose de dizaines de Mo de mémoire et de processeurs tournant à des centaines de MHz, ressources dont le gaspillage bien orchestré dynamise incontestablement les ventes de nouveaux PC ! En matière de cartes à puce, on parle plutôt de ko et de quelques MHz, l'équivalent, en somme, des ordinateurs "familiaux" d'il y a une vingtaine d'années. C'est ce qui fait que le bon vieux Basic est peut-être, finalement, le langage le mieux adapté à la programmation des cartes.

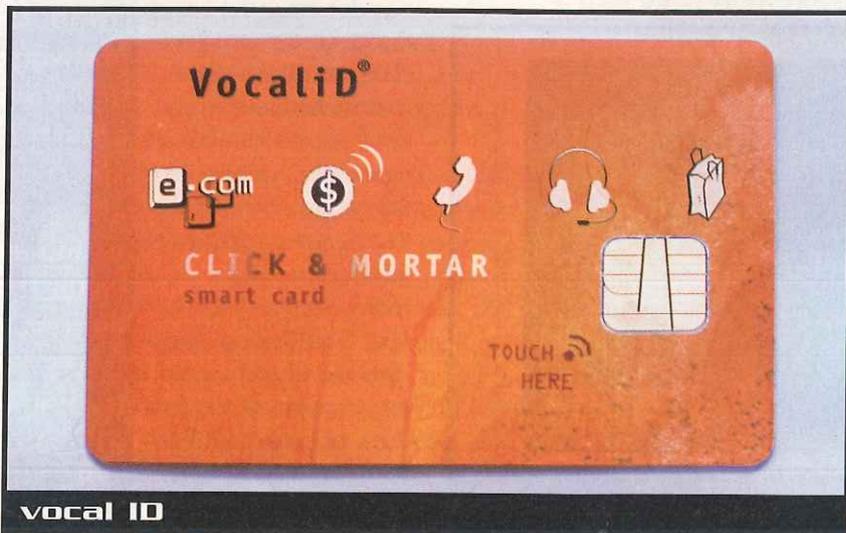
Alors même que les conférenciers de CARTES 2001 commencent à reconnaître que Java est affreusement gourmand en ressources, la BasicCard pourrait bien passer du statut de "première carte à puce programmable en Basic" à celui de "seule carte à puce programmable en Basic" !



**Secured sound**

**Flash et Forth**

Le composant ATMEL (AT90SC3232C) dont est équipée la BasicCard ZC 4.1 appartient, pour sa part, à la toute dernière génération de "puces pour cartes". Celle-ci se caractérise notamment par le remplacement, pour le système d'exploitation, de la traditionnelle ROM masquée par de la mémoire Flash téléchargeable. La disparition des frais de masque et des délais correspondants apporte une extraordinaire souplesse, sans parler d'une indéniable démocratisation. Sans la hantise d'un bug obligeant à refaire un masque, et peut-être même à jeter des milliers de puces, toutes



**vocal ID**



**SIM Trucker**

les audaces sont permises dans la mise au point de systèmes d'exploitation sur mesures.

En matière de téléphonie mobile, par exemple, des cartes SIM extraordinairement innovantes peuvent être produites, même en très faibles quantités, quitte à faire entièrement table rase de ce qui existe déjà. Attendons-nous, par exemple, à entendre bientôt parler de cartes GSM prépayées permettant à des opérateurs étrangers,

peut-être même "virtuels", de rerouter à moindres frais les communications de leurs clients empruntant les réseaux des opérateurs nationaux : nous en avons déjà vu fonctionner un prototype, basé sur une carte équipée d'une puce Flash d'origine Emosyn.

Enfin, qui l'eût cru ? Le Forth refait parler de lui grâce aux cartes à puce !

Exceptionnellement compact, ce langage assez ardu n'est jamais complètement

tombé en désuétude. On lui doit même un certain nombre de développements de pointe en matière de logiciels embarqués. Du côté des cartes à puce, la société Forth Inc. annonçait à CARTES 2001 son outil de développement "SwiftX-SC", version "cartes à puce" de son cross-compileur pour systèmes embarqués.

Côté matériel, il suffit d'un simple lecteur PC/SC et d'un PC pour mettre au point quasiment n'importe quelle application basée sur une carte à composant AT90SC de ATMEL.

Les fonctionnalités de base d'un système d'exploitation de cartes à puce sont disponibles dans un noyau modulaire baptisé SwiftCOS, que vient compléter toute une bibliothèque de fonctions cryptographiques.

Un coup d'œil s'impose aux esprits chagrins qui considèrent déjà la BasicCard comme "un peu trop puissante" à leur goût, et qui devront bien se résoudre à admettre que, décidément, la moindre faute de sécurité ne pardonnera plus.

**P. GUEULLE**

**LECTEUR/ENCODEUR DE CARTE A PUCE**

Le système de développement BasicCard comprend :

- 1 Lecteur/Encodeur CyBermouse (Série ou USB)
- 1 BasicCard 1 Ko EEprom
- 2 BasicCard 8 Ko EEprom
- 1 Lecteur avec afficheur LCD (Balance Reader)
- 1 CD avec logiciel de développement
- 1 Manuel



CYBERMOUSE



CHIPI-INTERNE  
CHIPI-EXTERNE

**PROGRAMMATEUR ET MULTICOPIEUR UNIVERSEL, AUTONOME, PORTABLE**



GALEP-III



ALL-11P2



TOPMAX

**ANALYSEUR LOGIQUE**



LA-2124

**EMULATEUR D'EPROM ET DE MICROCONTROLEUR**



DS-51

**SYSTEME DE DEVELOPPEMENT VHDL**



LP-2900

**LECTEUR/ENCODEUR DE CARTE MAGNETIQUE**

- MCR/MSR : Lecteur simple avec interface Série/TTL/Keyboard
- MSE-6xx :
- Lecteur/encodeur avec interface série



MAGSTRIPE MSE-630

**CARTES D'ÉVALUATION, D'ACQUISITION, BUS I<sup>2</sup>C, BUS PC/104**

**SIMULATION**



B2 SPICE



- 68HC 11/12/16
- 68 332
- 80C 552
- 80C 31/51
- 80C 535

- COMPILATEUR C & ASSEMBLEUR**
- 68HC 11/12/16
- 68/332
- 80C 31/51/552
- MICROCHIP PIC

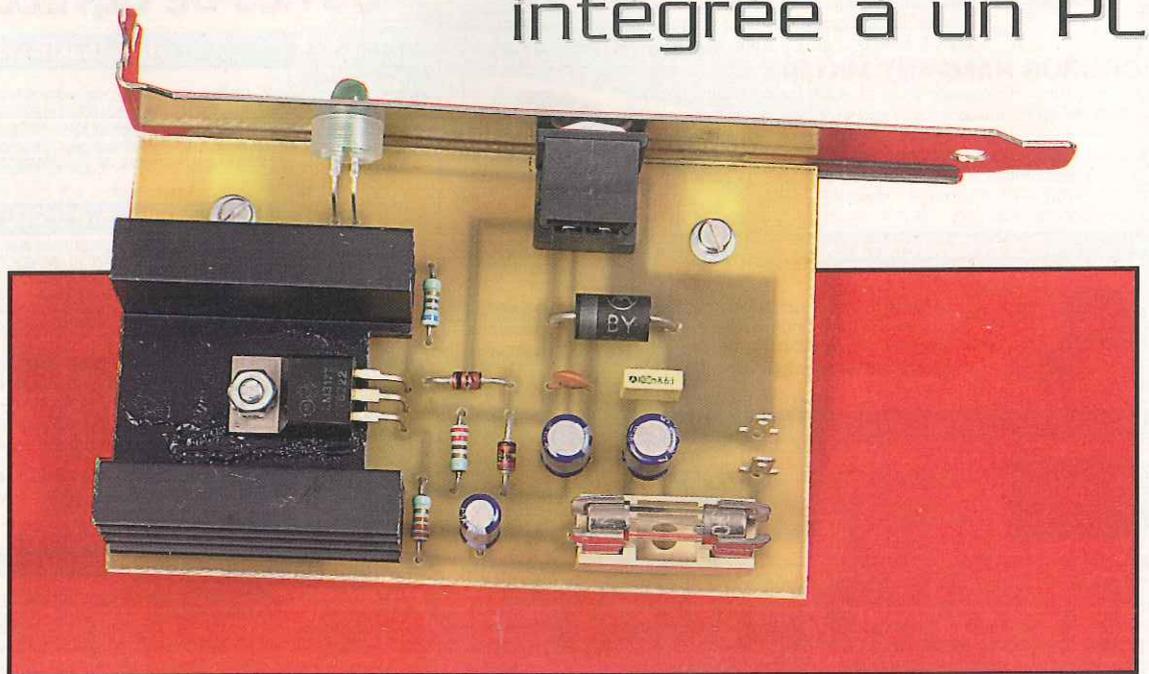
**HI TECH TOOLS (H.T.T.)**

27, rue Voltaire  
72000 LE MANS

Tél : 02 43 28 15 04  
Fax : 02 43 28 59 61

<http://www.hitechtools.com>  
E-mail : [info@hitechtools.com](mailto:info@hitechtools.com)

# Carte alimentation intégrée à un PC



Lorsque l'on veut essayer un montage relié au port série ou parallèle d'un PC et que celui-ci ne prélève pas son alimentation sur ce port, il est nécessaire d'amener à côté du PC une alimentation de laboratoire, de faire appel à des piles ou bien encore de réaliser une alimentation spécifique. Tout cela n'est pas très pratique alors que le PC dispose souvent en interne de tout ce qu'il nous faut, ou presque !

Nous vous proposons donc de réaliser une carte qui prend place dans votre PC mais ne s'enfiche pas sur les connecteurs de la carte mère, qui ne court ainsi aucun danger. Cette carte vous permet ensuite de disposer, depuis l'extérieur de votre PC, de n'importe quelle tension, fixe ou réglable entre 1,5 et 10V, sous un courant pouvant dépasser 1A. Elle est évidemment entièrement protégée contre les courts-circuits et les échauffements excessifs et ne fait donc courir aucun risque à votre précieux (et souvent coûteux) PC.

## Une idée de départ très simple

Comme vous le savez sans doute, l'alimentation d'un PC délivre de multiples tensions parmi lesquelles la valeur la plus élevée qui nous soit accessible est le +12V destiné aux périphériques internes que sont les disques durs et autres lecteurs et graveurs de CD ROM. Avec une alimentation de PC standard, d'une puissance de 250W environ, et sous réserve que vous n'ayez pas une dizaine de périphériques internes dans votre machine, il reste possible

de «tirer» sans difficulté plus d'un ampère sur cette tension.

Il suffit donc de faire appel à un régulateur de tension, réglable depuis l'extérieur du PC, pour disposer d'une alimentation stabilisée parfaitement apte à répondre à la majorité de nos besoins.

Bien sûr, afin de protéger le PC des manipulations malheureuses, il faut utiliser un régulateur convenablement protégé mais, comme nous allons le voir, c'est là chose facile.

Cette idée de départ très simple a reçu une mise en œuvre que nous avons voulu aussi astucieuse et pratique que possible. Vous pouvez donc relier à notre carte de simples «câbles» qui définissent alors chacun une tension de sortie fixe de votre choix, ou bien encore un boîtier de commande externe qui vous permet de sélectionner des tensions fixes prédéterminées ou une tension variable en continu.

## Schéma de notre alimentation

Le schéma utilisé (figure 1) n'a rien de bien original pour qui lit régulièrement les revues d'électronique. Il fait

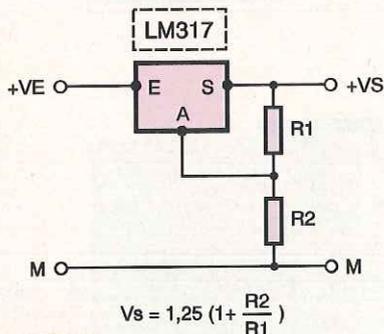
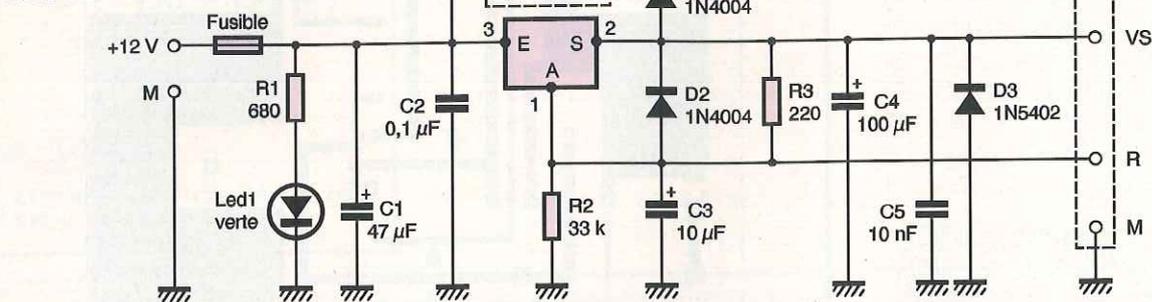
appel, en effet, à un classique régulateur trois pattes ajustable, en l'occurrence un LM317 de NS ou d'une de ses innombrables secondes sources.

La tension positive de 12V provenant du PC passe par un fusible, ultime protection au cas où vous feriez vraiment trop souffrir l'alimentation. Elle arrive ensuite sur l'entrée d'un LM317 monté de façon conventionnelle même si cela ne se voit pas au premier abord sur cette figure.

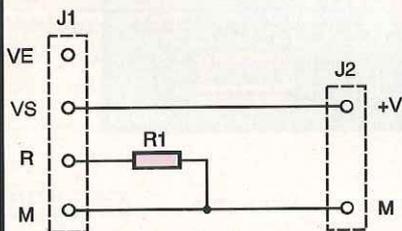
En effet, comme le montre la figure 2, la tension de sortie délivrée par le LM317 ne dépend que du rapport des résistances  $R_1$  et  $R_2$ . Or, si vous calculez ce rapport dans le cas de la figure 1, vous constaterez que l'on dépasse largement les 12V d'entrée. La raison en est très simple. En fait, la tension de sortie de notre carte est déterminée par la résistance (dans le cas d'une tension fixe) ou par le potentiomètre (dans le cas d'une tension variable) connecté entre les bornes R et M de  $J_1$ . Cette résistance ou ce potentiomètre se trouve donc en parallèle sur  $R_2$  et permet de disposer ainsi de n'importe quelle tension comprise entre 1,5 et 10V.

En théorie, nous aurions pu nous

**1**  
Schéma de notre  
carte alimentation



**2**  
Principe d'utilisation  
du LM317

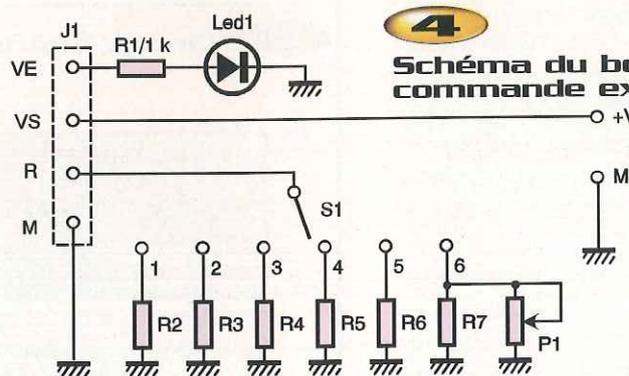


**3**  
Schéma d'un «câble»  
de programmation  
de tension

passer de  $R_2$  mais, en cas d'oubli de connexion de la résistance externe entre R et M, le régulateur aurait fonctionné en mode flottant ce qui n'est pas toujours souhaitable.

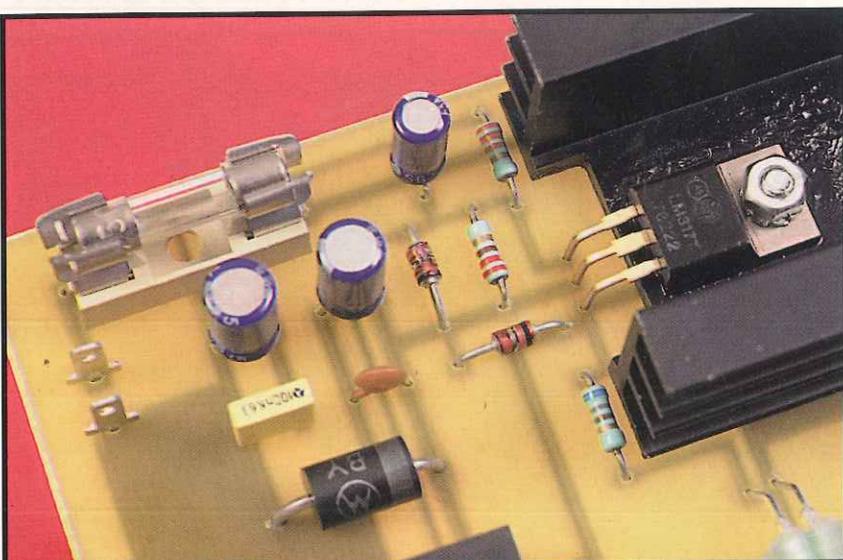
Les diodes  $D_1$  et  $D_2$  assurent la protection du régulateur dans toutes les situations possibles tandis que la diode  $D_3$  protège l'alimentation en cas d'application d'une tension externe négative. Remarquez que la tension d'entrée du régulateur est égale-

**4**  
Schéma du boîtier de  
commande externe



ment ramenée sur le connecteur  $J_1$ . Ce connecteur est en fait une prise mini DIN femelle à 4 broches. Comme le montre la **figure 3**, la prise mâle correspondante peut alors contenir une seule résistance fixe, repérée  $R_1$  sur cette figure ; résistance qui définit alors la tension de sortie de l'alimentation. Il est ainsi possible de réaliser autant de «câbles» de ce type que l'on souhaite disposer de tensions fixes, et le simple fait de brancher l'un d'entre eux dans le connecteur de sortie de la carte lui fera alors générer la tension correspondante.

Il est également possible de relier la carte, toujours au moyen de cette mini DIN à 4 broches, au boîtier de commande externe (**figure 4**). Celui-ci permet de commuter un certain nombre de résistances fixes vous donnant alors un accès immédiat aux tensions correspondantes sans avoir besoin d'un voltmètre de contrôle. Il dispose également d'une position mettant en circuit un potentiomètre qui permet alors d'ajuster la tension de sortie à n'importe quelle valeur comprise entre 1,5 et 10V. La tension d'entrée de la carte, quant à elle, est ramenée via la borne VE sur une LED qui indique, par son allumage, que l'alimentation 12V du PC n'est pas surchargée par le courant débité par notre carte.



utilisation de composants très classiques

## La réalisation

L'approvisionnement des composants ne pose aucun problème car tous sont des classiques disponibles partout. Par contre, notez que nous avons dessiné le circuit imprimé pour un radiateur ML33 de chez SELECTRONIC. Si vous ne voulez pas utiliser ce modèle, veillez à approvisionner un radiateur de taille équivalente.

Pour ce qui est de l'équerre de fixation de la carte en face arrière du PC, vous pouvez utiliser une équerre que vous aurez récupérée sur un emplacement vide, que vous munirez de deux équerres de 10x10mm qui se visseront dans les trous prévus à cet effet sur notre carte. Si ce «bricolage» ne vous inspire pas, sachez que vous pouvez aussi faire appel à une équerre spécifique munie de pattes de fixation. Les trous dessinés sur notre carte ont été prévus pour le modèle de ce type qui porte la référence 115-4568 chez RADIOSPARES.

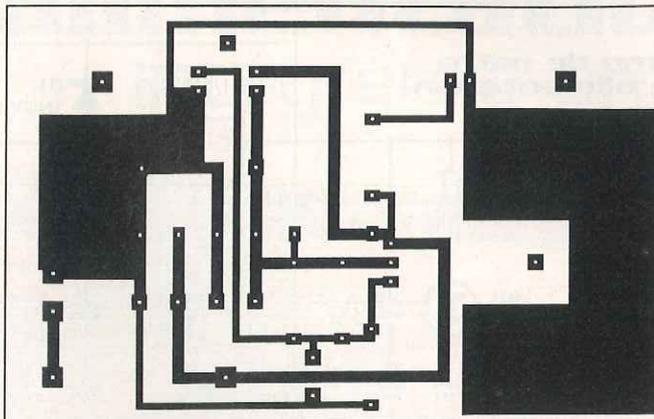
Le circuit imprimé adopte un tracé très simple (**figure 5**). L'implantation des composants est à réaliser en suivant les indications de la **figure 6**. Commencez par les composants passifs pour terminer par les composants actifs en respectant bien le sens des condensateurs chimiques et des diodes.

Le LM317 peut être monté directement sur le radiateur puisque celui-ci ne touche aucune partie métallique du PC. Lors de ce montage, interposez un peu de graisse aux silicones entre le LM317 et son radiateur afin d'améliorer la conduction thermique. L'équerre métallique de la face arrière du PC est à percer de deux trous. L'un sert à faire dépasser la LED placée en haut de la carte et l'autre à donner accès à la prise DIN à 4 broches.

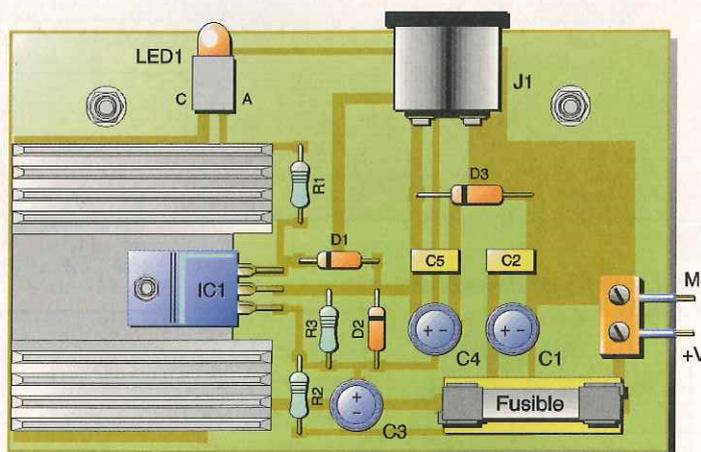
Une fois ces trous réalisés, vous pouvez alors assembler le tout et vous préparer à la mise en place dans votre PC, non sans avoir, au préalable, réalisé un cordon de sélection de tension.

Utilisez pour cela le schéma de la figure 3 et le brochage de la prise DIN présenté **figure 7** pour réaliser ce câble. La valeur de la résistance vous est indiquée par le **tableau 1** dans lequel nous avons fait figurer toutes les tensions standards dont vous pourriez avoir besoin.

Vous pouvez aussi, si vous le désirez, réaliser le boîtier de commande de la figure 4.



### 5 Circuit imprimé



### 6 Implantation des composants

Nous n'avons pas dessiné de circuit imprimé pour cela car les résistances peuvent se câbler directement sur les cosses du com-

mutateur. Un petit boîtier plastique à façade métallique convient à merveille pour recevoir le peu de composants nécessaires.



perçage de l'équerre métallique



tous les PC de la création. Si vous n'avez plus de connecteur d'alimentation libre, sachez que l'on trouve des câbles de dérivation en Y chez tous les bons revendeurs de matériel informatique pour environ 10 F (1,5 €). Enfichez le câble de sélection de tension de votre choix dans le connecteur de la carte et vérifiez que vous avez bien, en sortie de celle-ci, c'est à dire à l'extrémité du câble correspondant, la tension prévue par la résistance dont il est équipé.

Tension	R théorique	R réelle la plus proche
1,5V	44 Ω	43 Ω
3V	311 Ω	309 Ω
5V	678 Ω	665 Ω
6V	857 Ω	866 Ω
9V	1422 Ω	1430 Ω
1,25 à 10V	0 à 1540 Ω	2,2 kΩ en // sur 5100 Ω

**Choix des résistances (R<sub>1</sub> figure 3 ou R<sub>2</sub> à R<sub>7</sub> figure 4) en fonction de la tension de sortie désirée.**

Ici encore, le tableau 1 indique les valeurs des résistances et du potentiomètre à utiliser. Notez que toutes les résistances que nous avons indiquées sont disponibles, soit dans la série E24, soit dans la série E96 des résistances à 1%.

### Utilisation

Insérez la carte dans votre PC après l'avoir raccordée à la tension +12V et à la masse que vous prélèverez sur un connecteur libre destiné à un périphérique interne. La masse est repérée en noir et le +12V en jaune sur

Pour ce qui est des protections, le LM317 dispose d'une limitation de son courant de court-circuit qui varie de 1,5 à 3,4A selon les modèles. Il est également protégé contre les échauffements excessifs. Notre alimentation est donc virtuellement indestructible. La seule manipulation à ne pas faire, mais elle est très improbable, consiste à appliquer sur la sortie de l'alimentation une tension de même polarité provenant d'une alimentation externe.

C. TAVERNIER

## Nomenclature

- IC<sub>1</sub> : LM317 en boîtier T0220
- D<sub>1</sub>, D<sub>2</sub> : 1N4004
- D<sub>3</sub> : 1N5402, BY252
- LED<sub>1</sub> : LED verte
- R<sub>1</sub> : 680 Ω 1/4W 5% (bleu, gris, marron)
- R<sub>2</sub> : 33 kΩ 1/4W 5% (orange, orange, orange)
- R<sub>3</sub> : 220 Ω 1/4W 5% (rouge, rouge, marron)
- C<sub>1</sub> : 47 µF/25V chimique radial
- C<sub>2</sub> : 0,1 µF Mylar
- C<sub>3</sub> : 10 µF/25V chimique radial
- C<sub>4</sub> : 100 µF/25V chimique radial
- C<sub>5</sub> : 10 nF céramique
- J<sub>1</sub> : prise mini DIN 4 contacts femelle pour circuit imprimé

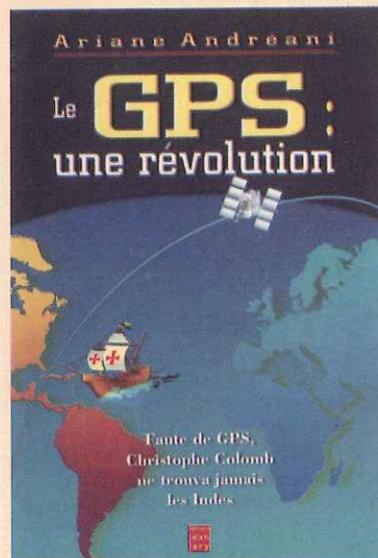
- Radiateur type ML33 (SELECTRONIC) ou équivalent pour IC<sub>1</sub>
- Porte fusible pour CI et fusible T20 de 2A temporisé
- Équerre métallique standard pour carte d'extension PC (voir texte)

#### Boîtier de commande

- LED<sub>1</sub> : LED verte
- R<sub>1</sub> : 1 kΩ 1/4W 5% (marron, noir, rouge)
- R<sub>2</sub> à R<sub>7</sub> : voir tableau
- P<sub>1</sub> : potentiomètre linéaire rotatif de 2,2 kΩ
- J<sub>1</sub> : prise mini DIN mâle 4 points à monter sur câble
- S<sub>1</sub> : commutateur rotatif 2 circuits, 6 positions

## Le GPS : une révolution

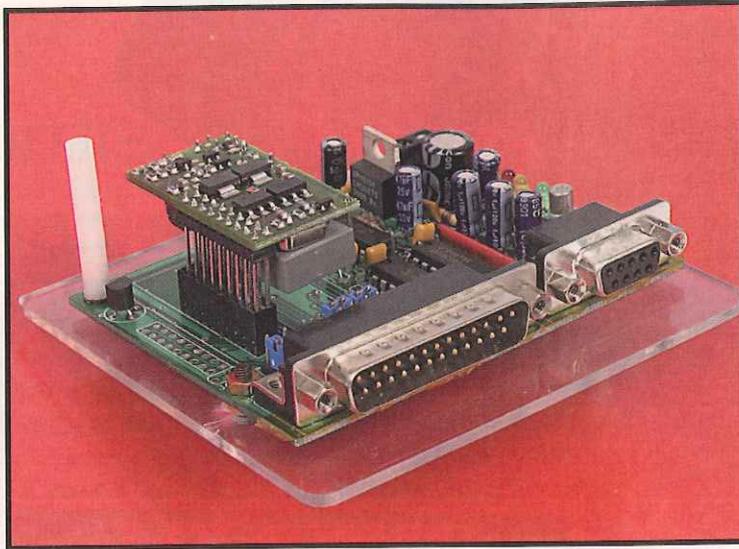
**D'une précision et d'une simplicité d'utilisation sans égales, le GPS a révolutionné la navigation et les transports.**



*Pour preuve, rares sont les navires et les véhicules en général à n'être pas équipés d'une antenne GPS. Son succès ne s'arrête pourtant pas là. Que ce soit pour construire un pont, pour prédire les changements climatiques, pour anticiper les secousses sismiques ou pour synchroniser les horloges sur l'ensemble du globe, le système GPS s'est rendu indispensable. Les nombreuses interrogations des utilisateurs sur le fonctionnement, la précision et les applications du GPS ainsi que sur les systèmes de référence en vigueur ont motivé la rédaction de cet ouvrage. Son autre objectif est de fournir au lecteur une source d'informations rare sur l'ensemble des données, services et produits disponibles. Ce livre s'adresse aussi bien à des amateurs de loisirs qu'à des professionnels désirant approfondir leurs connaissances.*

Ariane ANDREANI - Éditions Jean Jary  
160 pages - 20 € (131,19 F)

# PLINIUS, modems par courant porteur pour ordinateurs



**La société LEXTRONIC commercialise un nouveau produit innovant qui permet la communication entre deux ordinateurs. Ce système consiste en deux modems connectés sur les lignes du secteur et aux deux ordinateurs, système qui permet l'échange de données entre ces derniers par l'intermédiaire du réseau 220V. Cet échange sera évidemment "domestique" et restreint sur un même réseau, celui d'une habitation.**

**P**linius est un système complet de modems conçus pour les applications nécessitant des communications de données sur un réseau privé (bande CENELEC 95 - 148,5 kHz). Une grande intégration du système garantit un prix de revient très abordable et une grande fiabilité. Les principales applications de Plinius sont :

- la domotique,
  - les systèmes d'alarme et de sécurité,
  - le contrôle de systèmes industriels.
- La platine se compose d'un circuit d'interface avec l'ordinateur sur laquelle vient s'enficher le module Plinius qui permet la jonction aux lignes du secteur. Les principales caractéristiques de Plinius sont :
- circuit intégré de connexion synchrone/asynchrone pour jonction sur lignes secteur,
  - communications half-duplex en mode FSK 2400 BPS,
  - porteuse synthétisée par une horloge interne de 110592 kHz,
  - faible distorsion du signal TX et basse consommation de puissance,
  - haute sensibilité,
  - deux versions disponibles dépendant de la classe CENELEC (AV=12V/24V),
  - détection de porteuse,

- générateur programmable de RESET,
- sorties disponibles de la tension de référence et de l'horloge,
- compatible avec les normes CENELEC 500065-1 et FCC,
- protection contre les erreurs de transmissions.

Le kit d'évaluation est commercialisé sous la forme de deux platines mères sur lesquelles sont enfichés les modules à courant porteur Plinius ainsi qu'une disquette contenant le programme de communication entre celles-ci. Il conviendra de respecter les précautions d'usage comme pour tout montage relié directement au secteur. Un capot en matière plastique protège l'utilisateur du module Plinius relié au 220V. La mise en œuvre des modules est extrêmement simple.

Le contenu de la disquette, fournie avec le kit, devra être copié sur le disque dur des deux ordinateurs devant communiquer. Chaque carte mère devra être reliée par un cordon série à un port libre des ordinateurs (COM1 ou COM2). Ces câbles seront connectés aux prises DB9 (à droite), le connecteur à 25 broches (celui de gauche) ne devant pas être utilisé. De même, la position des cavaliers de configuration ne devra en

aucun cas être modifiée.

On connectera ensuite un cordon secteur au bornier prévu à cet effet, puis on fixera le capot protecteur sur l'entretoise prévue à cet effet. Une tension de +12V sera alors appliquée sur le bornier de droite (voir figure 2). Les LED rouge et orange devront alors s'allumer.

On reliera ensuite le cordon secteur au 220V et on lancera le programme "CHATLIN.EXE". Deux fenêtres devront alors s'ouvrir ("TX Frame" et "RX Frame"). Le port de communication COM1 ou COM2 devra alors être sélectionné et il suffira de cliquer sur la case "Start Cession". On effectuera les mêmes opérations sur le second ordinateur PC.

En cliquant sur la case "Connect" des deux PC, les deux cartes se connectent et la fenêtre "RX Frame" affiche le message "Hello".

Il ne restera plus qu'à taper un texte sur l'un des ordinateurs et d'appuyer sur la touche "Enter" afin que ce dernier apparaisse dans la fenêtre "RX Frame" du second ordinateur. On effectuera exactement la même opération avec ce dernier afin de vérifier que la transmission s'effectue correctement dans les deux sens. Il conviendra ensuite de frapper la touche "END" afin de sortir du programme.

En conclusion, nous dirons que ce kit permettra la mise en œuvre de nombreuses applications de domotique et de surveillance dans une habitation tout en évitant la mise en place de câbles peu esthétiques. Il conviendra de respecter scrupuleusement les précautions d'usage puisque le secteur est utilisé. Cependant, les platines sont efficacement isolées et cela ne constitue pas un problème pour peu que l'on respecte les recommandations du fabricant.

**P. OGUIC.**

# BasicSIM



## Un simulateur de carte SIM

**Jusqu'à présent, nous avons mis à contribution le kit BasicCard pour développer des applications "terminal". Voici venu le moment de lui faire donner sa pleine mesure, en faisant entrer en scène l'étonnante carte à puce à "système d'exploitation ouvert" qui en constitue, finalement, la raison d'être. Nous allons ainsi montrer comment une BasicCard peut simuler la carte SIM d'un téléphone portable, de façon aussi bien matérielle que "virtuelle", d'ailleurs !**

### Simulateur ou émulateur ?

Comme son nom l'indique, un simulateur de carte SIM a pour rôle de simuler logiquement le fonctionnement de la carte SIM d'un téléphone portable. Un émulateur serait plutôt destiné à prendre la place de la carte SIM dans un véritable mobile GSM, s'apparentant finalement à ce qu'il est convenu d'appeler une "SIM de test". Les BasicCard "enhanced" ZC3.3 ou ZC3.9 fournies dans les kits de développement fonctionnant exclusivement en protocole "T=1", ne peuvent servir que de simulateur et, en aucun cas, d'émulateur.

La spécification GSM 11.11 de l'ETSI imposant l'usage du protocole T=0, la réalisation d'un émulateur ne pourrait s'envisager qu'avec la toute nouvelle BasicCard "professional" (ZC 4.1). Et on y pense... Reste qu'un simple simulateur peut déjà imiter assez fidèlement une véritable SIM lorsqu'elle est insérée dans un lecteur de cartes à puce supportant le protocole T=1. Tel est, bien évidemment, le cas des lecteurs CyberMouse livrés dans les kits BasicCard et de la plupart des lecteurs PC/SC susceptibles d'être utilisés avec les logiciels de la

"boîte à outils" décrite dans notre n°10. Mais la simulation de cartes SIM peut aussi s'envisager sans carte ni lecteur !

En effet, le kit BasicCard dispose d'un puissant simulateur logiciel, capable de faire "tourner" simultanément sur un simple PC une application "terminal" et une application "carte". Dans ces conditions, la version "virtuelle" de notre carte SIM constituera un puissant outil de mise au point d'applications inspirées, peut-être, de notre "boîte à outils". Cela d'autant plus que nous avons imaginé une fonction "espion", permettant à la carte (réelle ou virtuelle) d'enregistrer "au fil de l'eau" toutes les commandes qu'elle reçoit.

L'analyse ultérieure du fichier "log" ainsi créé constituera une aide précieuse pour comprendre en profondeur le détail des échanges intervenus entre la carte et son "terminal". Bien entendu, les contenus de tous les fichiers de notre carte simulée pourront être modifiés en toute liberté, ce qui permettra de se placer, "pour voir", dans des situations que l'on ne pourrait pas reproduire avec une carte SIM émise par un opérateur de téléphonie mobile, dont maints fichiers

sont protégés en écriture.

D'un point de vue pratique, la BasicSIM se présente sous la forme d'un fichier ".IMG" produit avec la toute récente version 4 du kit de développement, offerte d'ailleurs sur notre CD-ROM (sous-répertoire INSTBC4). Indispensable pour travailler avec la BasicCard "professional" ZC4.1, cet outil résolument professionnel bénéficie d'améliorations décisives par rapport à la version 3 et mérite, par conséquent, d'être adoptée même si l'on n'utilise encore que des ZC1.x et 3.x.

Pour que nos lecteurs ne soient pas contraints de procéder immédiatement à cette migration, qui suppose tout de même une période d'adaptation, nous avons extrait du kit "version 4" les éléments indispensables à la mise en œuvre de la BasicSIM, qui peuvent fort bien être utilisés séparément. On se servira, par exemple, de l'utilitaire de téléchargement BCLOAD à la place de l'ancien WBCLOAD, pour programmer une ZC3.3 avec BS33.IMG, ou bien une ZC3.9 avec BS39.IMG.

Pour faire de la simulation pure, c'est le contenu du sous-répertoire SIMU que l'on utilisera, après l'avoir recopié

sur le disque dur du PC. Il s'y trouve, en effet, non seulement un fichier BS.IMG représentant la carte, mais des versions ".IMG" de tous les programmes de notre "boîte à outils" du n°10.

Grâce à un fichier batch SIMU.BAT (dont il n'est pas interdit d'examiner le contenu), on tapera simplement, par exemple, SIMU DIAGSIM si l'on souhaite simuler notre programme DIAGSIM.EXE avec la BasicSIM. Il convient de noter que le fichier BS.IMG est automatiquement mis à jour à la fin de chaque session de simulation, enregistrant ainsi tout changement intervenu dans la zone EEPROM de la BasicSIM. Des modifications apportées, par exemple, avec NAVSIM, seront ainsi prises en compte par le prochain programme dont on viendra simuler le fonctionnement. L'état d'origine de la BasicSIM pourra, bien entendu, être restauré aussi souvent qu'on le souhaitera, en "écrasant" tout simplement BS.IMG avec son original, repris sur le CD-ROM. Une autre méthode, plus proche de la réalité matérielle, consisterait à réaliser une BasicSIM à partir d'une ZC3.3 ou d'une ZC3.9 et à exécuter directement les programmes de notre "boîte à outils" sur un lecteur PC/SC (voir n°10) ou bien sur un CyberMouse resté dans son mode série "natif" utilisé pour télécharger la carte. Il conviendrait alors d'utiliser les variantes de notre "boîte à outils" fournies à cet effet (sous-répertoire EXE1 si le lecteur est branché sur COM1: ou EXE2 s'il est connecté à COM2:).

Mais le fin du fin consistera évidemment à essayer la BasicSIM avec des logiciels du commerce tels que SIMSurf Profi (<http://www.towitoko.de>) ou SIMmate 2000 (<http://www.simmate2000.com>). C'est là que le mode "espion" de la BasicSIM pourra se déchaîner, révélant commande par commande comment fonctionnent ces logiciels, ce qui est une source d'inspiration inestimable...

## Mise en œuvre pratique

Le lecteur CyberMouse d'un kit BasicCard étant supposé branché sur le port série COM1: du PC (seulement après le démarrage réussi de Windows si un driver PC/SC a déjà été installé), on ouvrira une fenêtre MS-DOS avant de se placer dans le répertoire où l'on aura copié les fichiers du CD-

ROM. Il suffira alors de taper :

```
BCLoad -D -P1 BS33
```

pour programmer une ZC3.3.

Si le lecteur était branché sur COM2: et que l'on veuille programmer une ZC3.9 (version PHILIPS de la ZC3.3 qui, elle, est produite par INFINEON), on taperait naturellement :

```
BCLoad -D -P2 BS39
```

Cela étant fait, il faut encore initialiser la BasicSIM au moyen de l'utilitaire BSUTIL.EXE, lui aussi fourni dans trois versions différentes (PC/SC, COM1: et COM2:) couvrant toutes les situations possibles. C'est ce même utilitaire qui permet d'activer ou désactiver le mode "espion" de la BasicSIM et aussi de vider le fichier "log" dès qu'il a été récupéré sur le PC (faute de quoi la mémoire de la carte pourrait finir par "déborder"). Cette opération de récupération est prise en charge par un utilitaire séparé, RDSPY.EXE, lui aussi fourni dans les trois versions.

Le fichier CARD.LOG obtenu sur le disque dur du PC est un fichier ASCII, consultable avec n'importe quel éditeur de texte et composé d'une ligne de valeurs hexa pour chaque commande ayant été reçue par la carte. Voici un court extrait d'un tel enregistrement (la suite se trouve avec les autres programmes sur le CDROM et sur notre site) :

```
A0 A4 00 00 02 7F 20
A0 C0 00 00 16
A0 A4 00 00 02 6F AE
A0 C0 00 00 0F
A0 B0 00 00 01
A0 A4 00 00 02 3F 00
A0 C0 00 00 16
A0 A4 00 00 02 3F 00
A0 C0 00 00 16
A0 A4 00 00 02 2F E2
A0 C0 00 00 0F
A0 B0 00 00 0A
A0 A4 00 00 02 3F 00
.....
```

Notons qu'une ligne vide en cours de listing indiquerait qu'un RESET est intervenu à cet instant (carte retirée et réinsérée, par exemple). L'analyse fine du contenu de ce fichier suppose une certaine connaissance de la spécification GSM 11.11 et de la norme ISO 7816 et donc, par exemple, la lecture préalable de nos ouvrages "Téléphones portables et PC" ainsi que "PC et cartes à puce", parus dans la collection ETSF de DUNOD.

## Détaillons un peu !

Il est bien évident qu'avec 8 Ko de mémoire EEPROM, dont le programme Basic occupe déjà plus de la moitié, une ZC3.3 ne peut accueillir la totalité des fichiers d'une carte SIM perfectionnée (16 à 64 Ko). Nous avons donc simulé, en priorité absolue, tous les fichiers dont la présence est indispensable pour que la carte soit reconnue comme une véritable SIM. Dans la limite de la mémoire disponible (et en laissant tout de même assez de place pour le fichier "log"), nous avons ensuite choisi, parmi les fichiers normalisés GSM 11.11, ceux pouvant donner lieu aux manipulations les plus intéressantes. Voici donc le résultat de ces choix, qui ouvrent déjà de vastes possibilités :

### Répertoire racine :

```
3F00:2FE2 (ICCID)
```

### Répertoire TELECOM :

```
7F10:6F3A (ADN, Abbreviated Dialling Numbers)
7F10:6F3B (FDN, Fixed Dialling Numbers)
7F10:6F3C (SMS, Short Messages)
7F10:6F40 (MSISDN, Own Numbers)
7F10:6F42 (SMSP, Short Messages Service Parameters)
7F10:6F43 (SMSS, SMS Status)
7F10:6F44 (LND, Last Number Dialed)
```

### Répertoire GSM :

```
7F20:6F05 (LP, Language preference)
7F20:6F07 (IMSI, International Mobile Subscriber Identity)
7F20:6F20 (Kc, Cipherring Key)
7F20:6F30 (PLMN, Preferred PLMNs)
7F20:6F31 (HPLMN search period)
7F20:6F38 (SST, SIM Service Table)
7F20:6F3E (GID1, Group Identifier level 1)
7F20:6F3F (GID2, Group Identifier level 2)
7F20:6F74 (BCCH, Broadcast Control Channels)
7F20:6F78 (ACC, Access Control Class)
7F20:6F7B (FPLMN, Forbidden PLMNs)
7F20:6F7E (LOCI, Location Information)
7F20:6FAD (AD, Administrative Data)
7F20:6FAE (Phase)
```

### Répertoire DCS :

```
7F21:6F05 (LP, Language preference)
7F21:6F07 (IMSI, International Mobile Subscriber Identity)
7F21:6F20 (Kc, Cipherring Key)
```

7F21:6F30 (PLMN, Preferred PLMNs)  
 7F21:6F31 (HPLMN search period)  
 7F21:6F38 (SST, SIM Service Table)  
 7F21:6F3E (GID1, Group Identifier level 1)  
 7F21:6F3F (GID2, Group Identifier level 2)  
 7F21:6F74 (BCCH, Broadcast Control Channels)  
 7F21:6F78 (ACC, Access Control Class)  
 7F21:6F7B (FPLMN, Forbidden PLMNs)  
 7F21:6F7E (LOCI, Location Information)  
 7F21:6FAD (AD, Administrative Data)  
 7F21:6FAE (Phase)

Le fichier "SST" (SIM Service Table) reflète fidèlement les compromis qu'il a ainsi fallu trouver, son contenu (DF 30 C3 F3) signalant par autant de bits à 1 les services normalisés qu'il a été possible de rendre existants, et/ou actifs.

même demandé un, n'importe quel code choisi au hasard serait accepté ! Il sera alors intéressant d'étudier les réactions des différents logiciels, de toutes provenances, que l'on aura la possibilité d'expérimenter. SIMSurf profi, par exemple, rejettera les cartes dont l'ICCID (numéro de série) lui paraît suspect.

PhoneFile, de son côté, rejette systématiquement les cartes "T=1" comme le ferait un véritable téléphone portable.

SIMmate 2000, enfin, accepte les cartes "T=1" mais éprouve des difficultés pour y lire les longs fichiers (en l'occurrence les SMS).

Un mot, enfin, au sujet de l'algorithme cryptographique "A3A8", qu'il n'était évidemment pas question de simuler fidèlement, tandis qu'il fallait tout de même assurer le fonctionnement vraisemblable

Octet	Actif	Existe	SERVICES GSM
1	b2	b1	Service 1: CHV (PIN)1 disable fonction
	b4	b3	Service 2: Abbreviated Dialling Numbers (ADN)
	b6	b5	Service 3: Fixed Dialling Numbers (FDN)
	b8	b7	Service 4: Short Message Storage (SMS)
2	b2	b1	Service 5: Advice of Charge (AoC)
	b4	b3	Service 6: Capability Config. Parameters (CCP)
	b6	b5	Service 7: PLMN Selector
	b8	b7	Service 8: Party Subaddress
3	b2	b1	Service 9: MSISDN
	b4	b3	Service 10: Extension 1
	b6	b5	Service 11: Extension 2
	b8	b7	Service 12: SMS Parameters
4	b2	b1	Service 13: Last Number Dialed (LND)
	b4	b3	Service 14: Cell Broadcast Message Identifier
	b6	b5	Service 15: Group Identifier Level 1
	b8	b7	Service 16: Group Identifier Level 2

Le contenu, par défaut, de chacun de ces fichiers a été soigneusement étudié de façon à être le plus vraisemblable possible, faute de quoi la carte risquerait d'être rejetée par tel ou tel logiciel de gestion de cartes SIM.

Chacun pourra toutefois modifier tout ou partie de ces données, car aucun code confidentiel ne lui sera normalement réclamé, y compris pour écrire dans des fichiers normalement protégés par un code "administrateur". Et s'il lui en était tout de

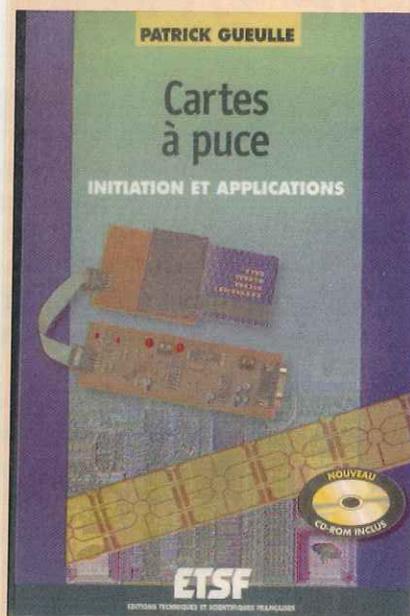
de la commande "RUN GSM ALGORITHM". On pourra constater, en exécutant (ou simulant) notre programme GSM.EXE, que la réponse (SPRES et Kc) est tout bonnement obtenue en "découpant" la valeur (normalement aléatoire) soumise au calcul.

Difficile de faire plus simple...

P. GUEULLE

## Cartes à puces Initiation et applications

**Cartes à puces va tout simplement vous permettre d'apprendre à lire et à écrire dans la plupart des cartes à puces, avec l'aide d'un micro-ordinateur compatible PC.**

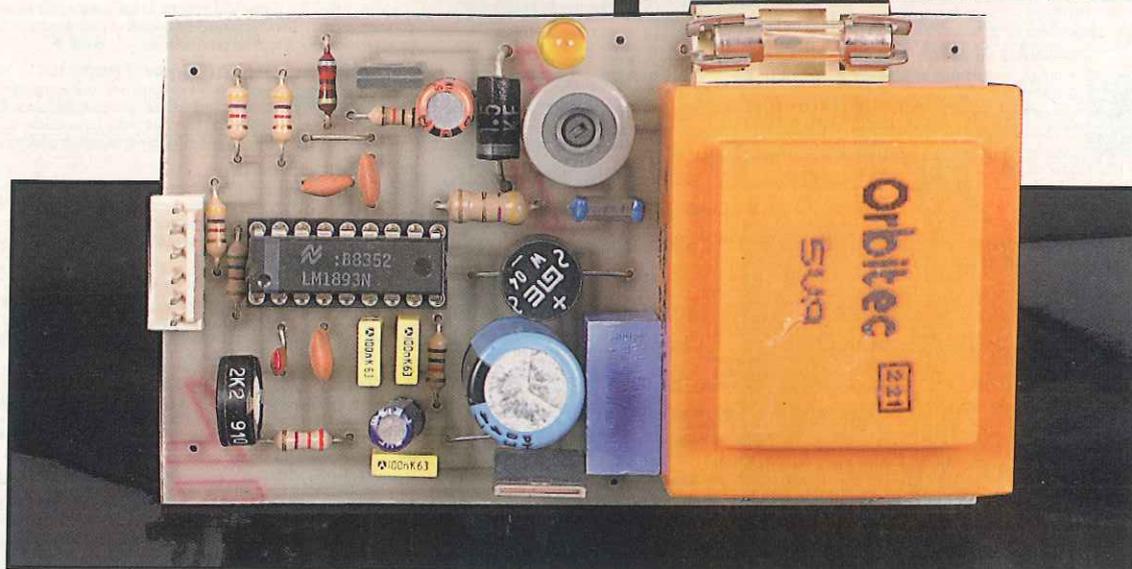


Le cédérom inclus dans le livre contient tous les logiciels et autres fichiers nécessaires. Vous commencerez par réaliser une "boîte à outils" complète : lecteurs, encodeurs, connecteurs, "fausses cartes" en circuit imprimé, alimentations spéciales, etc. Après une découverte progressive des possibilités de ces composants fascinants et d'une bonne partie de leurs petits secrets, vous pourrez monter vos propres applications pratiques : testeur de poche pour télécartes, serrures de sûreté à carte et même destructeur de cartes "compromettantes".  
 Public : Amateurs d'électronique, étudiants et professionnels qui désirent s'initier aux techniques liées à cette technologie.

Patrick Gueulle - DUNOD

176 pages - 225 FRF - 34,3 €

# Télécommande à courants porteurs sur port parallèle



Si l'automatisation de nombreuses fonctions domestiques comme le chauffage ou bien encore la commande de volets roulants semble aujourd'hui très facile avec un micro-ordinateur, cette facilité n'est qu'apparente. En effet, il ne suffit pas de relier à son PC une carte interface à relais sur le port série ou parallèle pour y parvenir car il reste à résoudre le délicat problème des liaisons aux organes commandés.

Deux solutions à ce problème existent aujourd'hui : les liaisons radio faisant appel aux célèbres modules hybrides de chez AUREL, MIPOT ou TELECONTROLLI, que nous avons déjà utilisés à de multiples reprises, et les courants porteurs que nous préconisons depuis des années et qui, curieusement, sont étrangement délaissés par les industriels français. Leur mise en œuvre est pourtant fort simple, peu coûteuse et la qualité de la liaison ainsi réalisée est excellente. Qui plus est, cette dernière est insensible aux perturbations radioélectriques, ce qui n'est pas le cas des modules radio précités.

Nous vous proposons donc de réaliser un système de télécommande à courants porteurs piloté par le port parallèle de n'importe quel compatible PC qui, comme vous allez pouvoir le constater dans un instant, présente tous ces avantages.

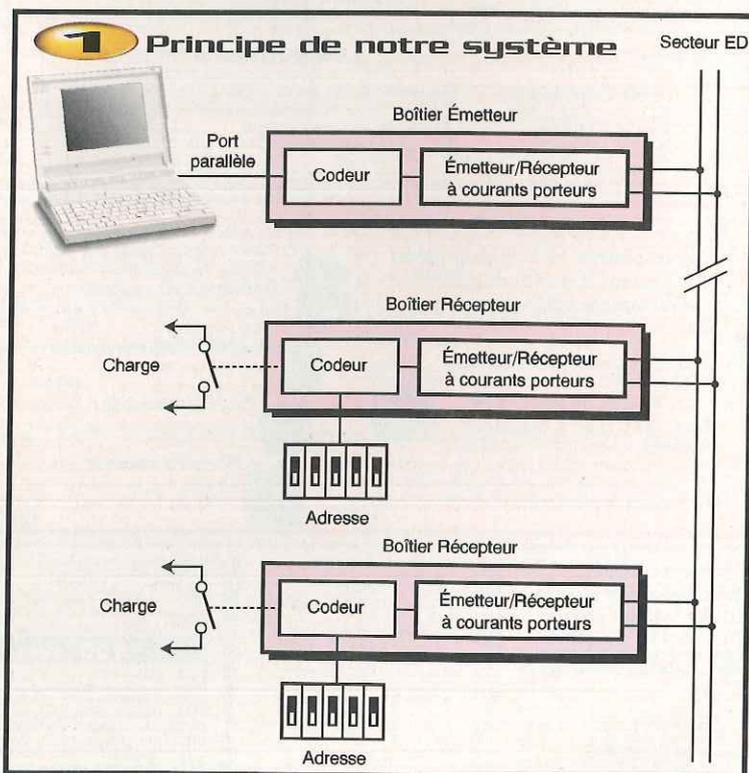
## Présentation

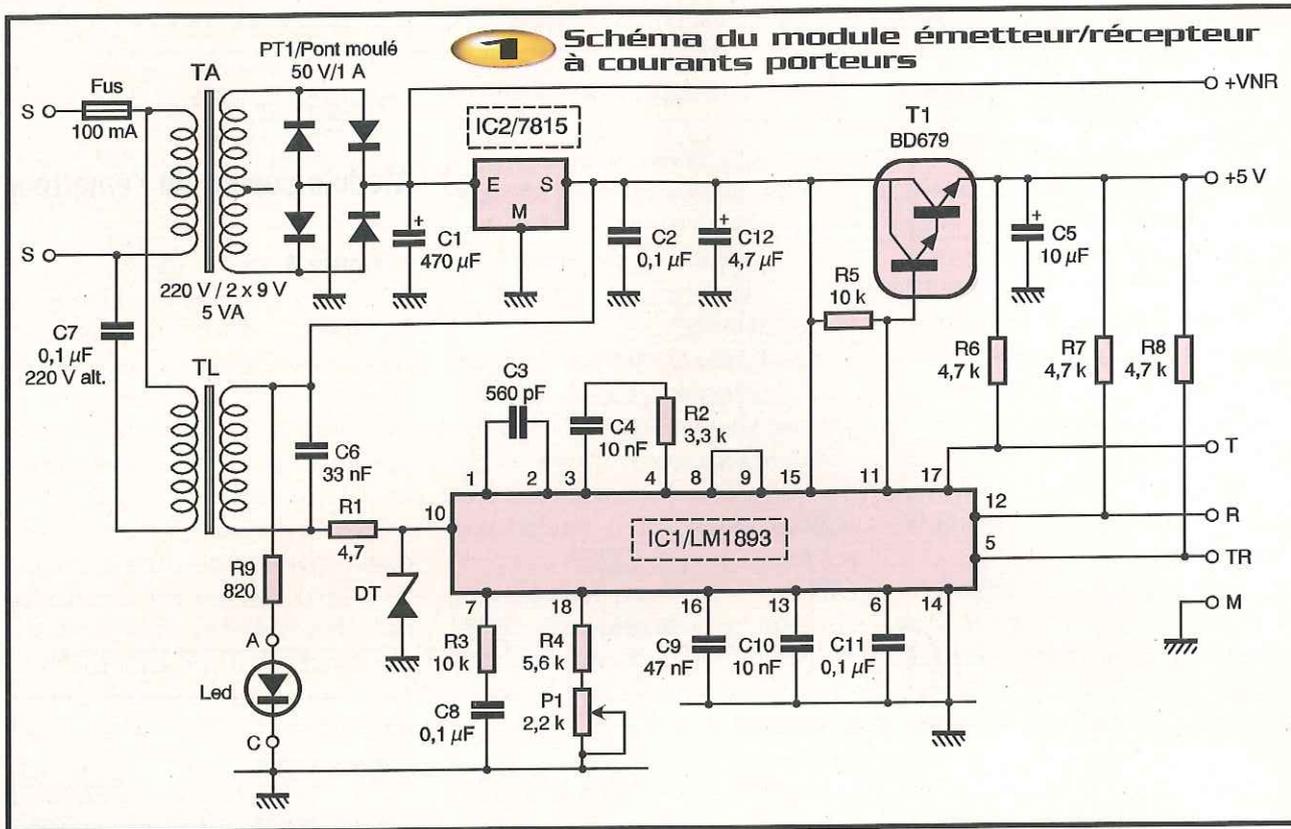
Comme le montre la **figure 1**, notre système est composé de deux éléments distincts : un boîtier émetteur réalisé en un seul exemplaire, placé à proximité immédiate

du micro-ordinateur, et autant de boîtiers récepteurs qu'il y a de charges à commander. Chaque boîtier récepteur est placé à proximité de la charge qu'il commande et il n'y a évidemment aucune liaison entre le boîtier émetteur et les boîtiers récepteurs : celle-ci ayant lieu grâce

aux courants porteurs véhiculés par le réseau EDF.

Rappelons que ce que l'on nomme courants porteurs consiste en fait à superposer au 50 Hz du secteur un signal haute fréquence, généralement aux environs de 150 kHz, que l'on peut moduler en fréquence : on est





alors en présence d'une émission de type FSK - ou en amplitude : on est alors en présence d'une émission de type ASK. En ce qui nous concerne et compte tenu du circuit d'émission/réception à courants porteurs choisi, nous utiliserons une émission de type FSK, plus résistante aux parasites violents que l'on rencontre sur le secteur que son homologue ASK. Une telle émission code les niveaux logiques hauts et bas au moyen de deux fréquences distinctes voisines de la fréquence centrale de fonctionnement.

Ceci étant précisé et comme le montre la figure 1, le boîtier émetteur contient deux modules distincts : un module émetteur/récepteur à courants porteurs, utilisé ici exclusivement en émission, et un module codeur chargé de coder les ordres à destination des boîtiers récepteurs. Ce module codeur se raccorde directement au port parallèle de n'importe quel compatible PC.

Au niveau de chacune des charges commandées, le boîtier récepteur comporte, lui aussi, deux modules : un module émetteur/récepteur à courants porteurs, identique à celui de l'émetteur mais utilisé ici exclusivement en réception, et un module décodeur chargé de décoder les

ordres émanant de l'émetteur et de faire coller ou décoller un relais.

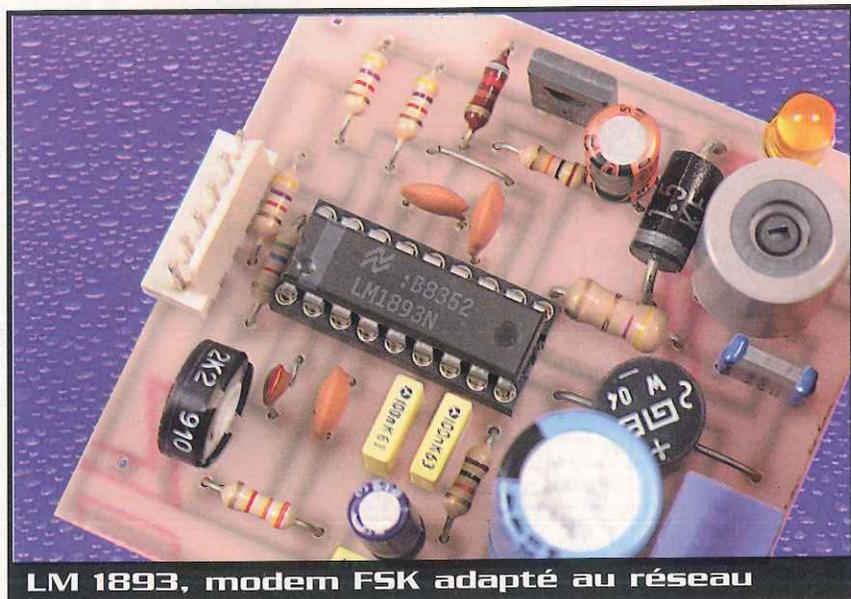
Chaque boîtier récepteur se voit affecter une adresse choisie au moyen de mini-interrupteurs DIL et c'est grâce à cette adresse que le boîtier émetteur peut sélectionner tel ou tel récepteur.

Bien sûr, il est possible d'affecter la même adresse à plusieurs boîtiers récepteurs qui réagiront alors en même temps aux mêmes ordres. C'est très utile pour commander par

exemple tout un groupe de convecteurs électriques.

## L'émetteur/récepteur à courants porteurs

Nos fidèles lecteurs reconnaîtront dans le schéma de la **figure 2** un montage que nous avons présenté à de nombreuses reprises depuis des années. Nous n'avons en effet pas jugé opportun de le modifier



car sa fiabilité est excellente comme le prouvent les nombreuses années d'utilisation qu'il a à son actif en de multiples exemplaires.

Il fait appel à un «vieux» LM1893 qui est un modem FSK complet spécialement adapté au fonctionnement sur le réseau EDF. Ce circuit peut travailler sur toute fréquence comprise entre 50 kHz et plus de 300 kHz et, en ce qui nous concerne, nous avons retenu la valeur de 125 kHz qui présente l'avantage de permettre de disposer de transformateurs HF prêts à l'emploi dans le commerce de détail. Malgré son âge, le LM1893 n'a toujours pas été détrôné, pour un tel usage, par des circuits plus récents.

L'alimentation fait appel à un simple transformateur 2 x 9V dont le point milieu n'est pas utilisé, ce qui permet de disposer après redressement et filtrage de 24V environ. Cette tension est régulée à 15V par le régulateur IC<sub>2</sub>.

Le couplage haute fréquence au réseau EDF fait appel au transformateur TL

accordé sur 125 kHz par C<sub>6</sub>. Le condensateur C<sub>7</sub> isole, quant à lui, le secondaire de ce transfo du 220V tout en laissant passer facilement le 125 kHz. La sortie de l'étage de puissance du LM1893 attaque le primaire de ce transformateur via la patte 10 du circuit. La diode DT est une diode Transil interdisant aux plus violents des parasites d'atteindre et de risquer de détruire le LM1893.

Le condensateur C<sub>3</sub> placé entre les pattes 1 et 2 fixe la fréquence centrale de l'oscillateur interne du LM1893 à 125 kHz environ. Cette fréquence peut être ajustée exactement par action sur P<sub>1</sub> afin de l'amener dans la plage prévue par le transformateur accordé TL.

La patte 11 est une sortie stabilisée à 5,6V qui, en alimentant la base de T<sub>1</sub>, permet de disposer ainsi d'une alimentation stabilisée 5V utilisée par les modules codeur et décodeur des boîtiers émetteur et récepteur(s).

La sortie des signaux reçus se fait sur la patte 12 tandis que les signaux à émettre

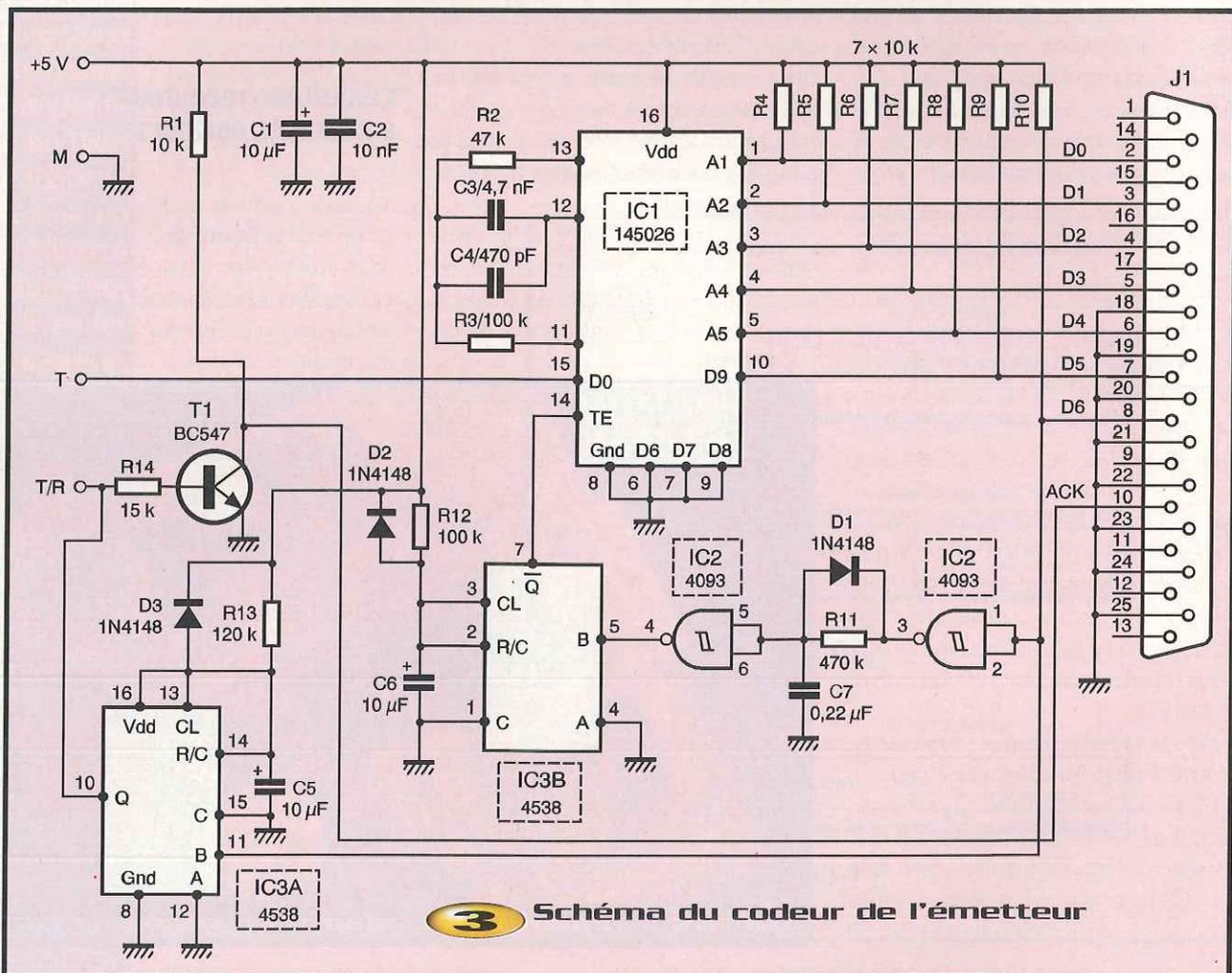
sont appliqués à la patte 17. La sélection émission/réception, enfin, se fait au moyen de la patte 5 qui place le circuit en réception lorsqu'elle est au niveau bas.

### Module codeur de l'émetteur

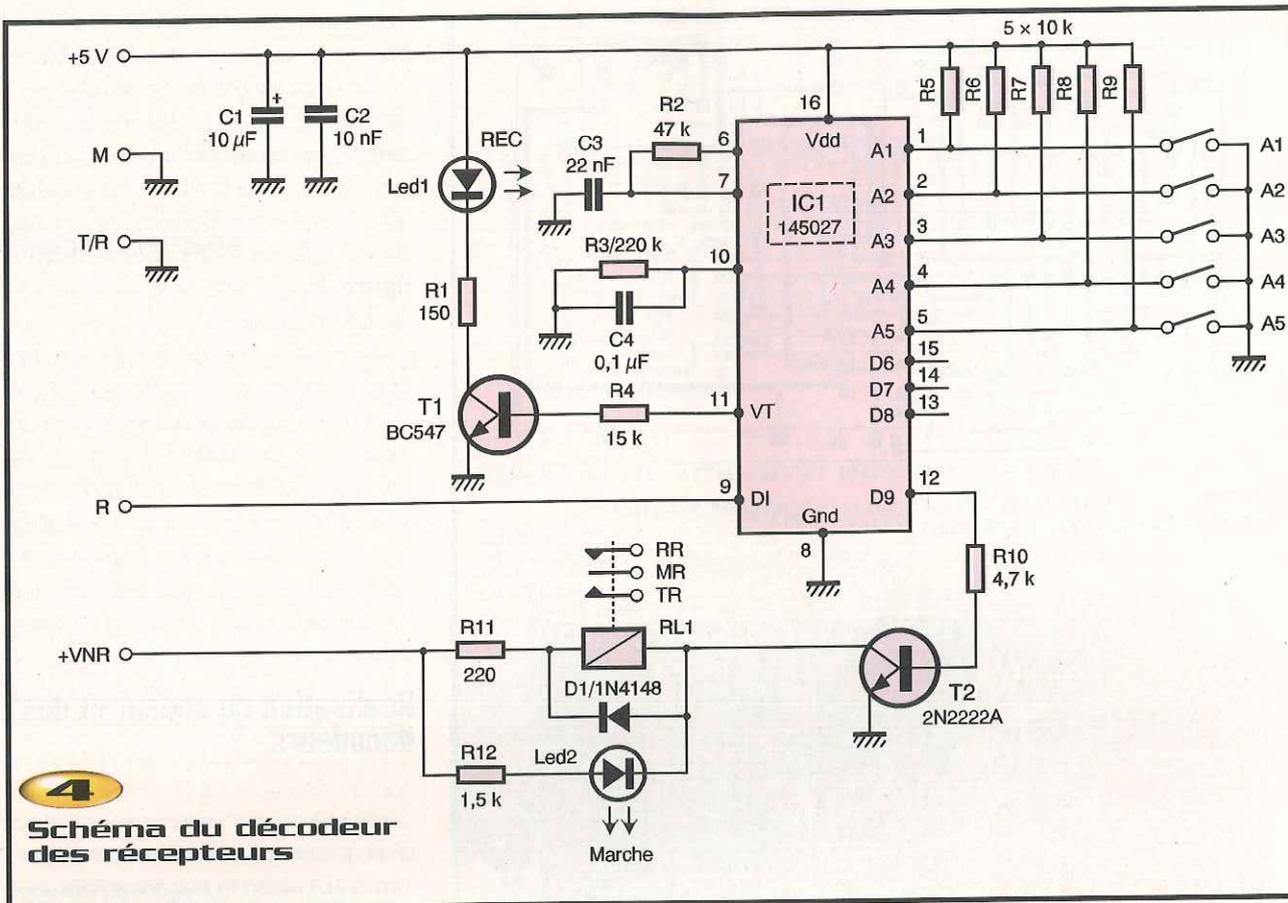
La **figure 3** présente son schéma complet. Il utilise un 145026, codeur numérique sur lequel nous ne reviendrons pas car il a déjà été utilisé à de nombreuses reprises dans Interfaces PC.

Les entrées A1 à A5 de IC<sub>1</sub> sélectionnent l'adresse du récepteur destinataire, tandis que l'entrée de donnée D9 permet de définir l'ordre à envoyer. L'information codée correspondante est disponible en patte DO de IC<sub>1</sub> pour peu qu'il soit validé par un niveau bas appliqué à sa patte TE. C'est le rôle du reste de la logique qui permet de n'envoyer un ordre que pendant quelques secondes afin de ne pas saturer le réseau EDF de transmission HF.

Pour cela, le micro-ordinateur doit tester la patte ACK de son interface parallèle



**3** Schéma du codeur de l'émetteur



**4**  
Schéma du décodeur des récepteurs

afin de s'assurer qu'aucun ordre n'est en cours d'émission. Il peut alors faire passer la patte D6 au niveau bas, ce qui déclenche immédiatement le monostable IC<sub>3a</sub> et place en émission le module émetteur/récepteur via sa ligne T/R. Simultanément, cela fait passer ACK au niveau bas indiquant que le module est occupé.

Quelques millisecondes ensuite, à cause de IC<sub>2</sub>, le monostable IC<sub>3b</sub> est déclenché à son tour et valide alors IC<sub>1</sub> en émission pendant environ une seconde. IC<sub>3b</sub>, puis IC<sub>3a</sub>, reviennent ensuite au repos libérant ACK.

Le module est alors prêt pour un nouvel envoi d'ordre.

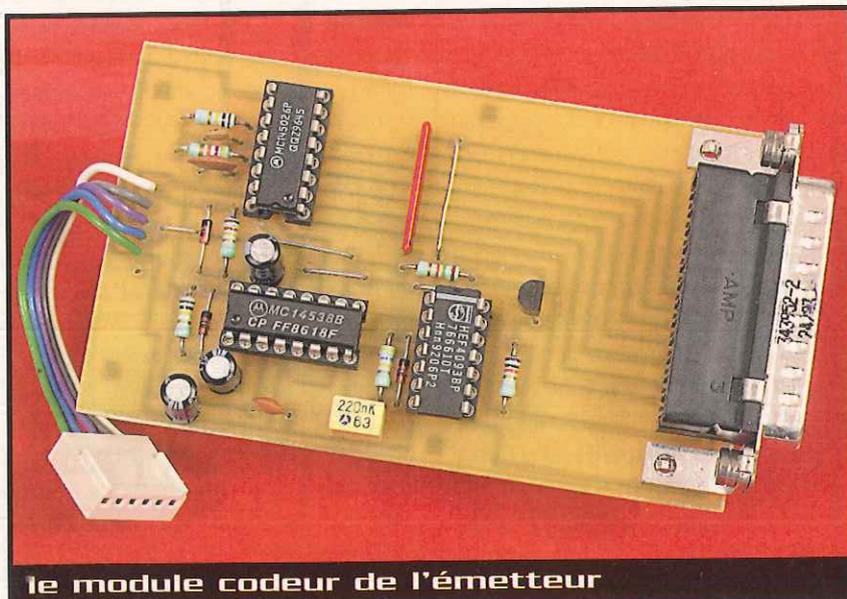
### Module décodeur des récepteurs

Son schéma vous est présenté **figure 4** et fait appel, comme il se doit, à un 145027, décodeur associé au 145026. Il reçoit les informations codées provenant du module émetteur/récepteur via sa patte DI. Si l'adresse programmée sur ses pattes A1 à A5 correspond à celle choisie sur l'émetteur, il fait passer VT au niveau haut tant que dure la réception de l'information et délivre cette dernière sur D9.

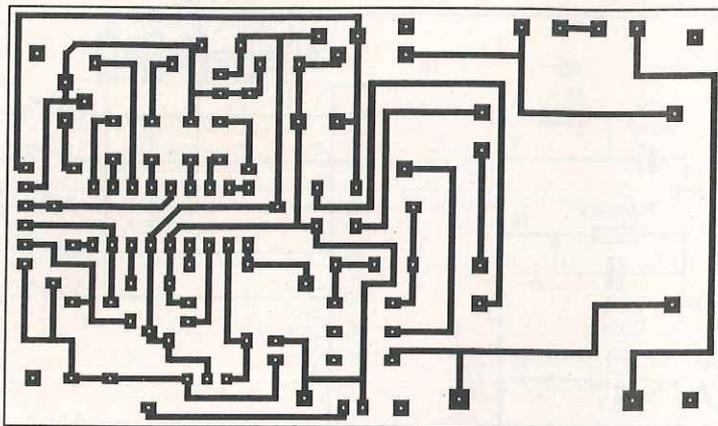
Selon que l'on a envoyé un 1 ou un 0, on fait donc coller le relais commandé par T<sub>2</sub>. La LED<sub>2</sub> correspondante permet de visualiser son état tandis que l'allumage de la LED<sub>1</sub>, commandée par T<sub>1</sub>, permet de constater que le module reçoit bien un ordre qui lui est destiné.

### Réalisation des émetteurs/récepteurs

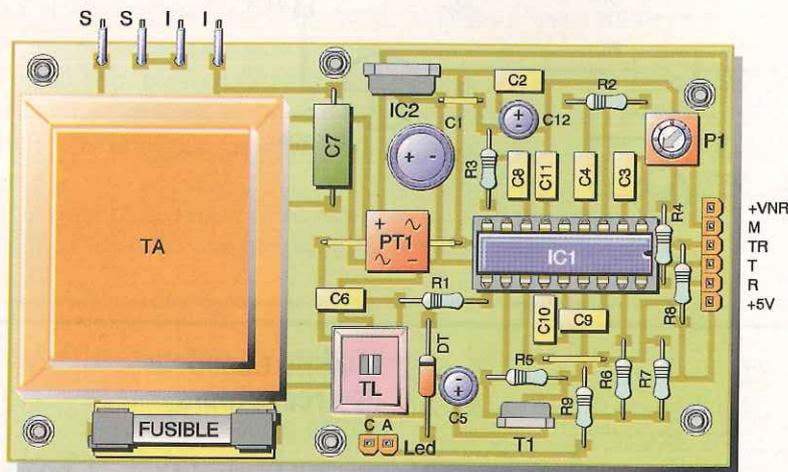
Comme vous avez pu le constater à la lecture du principe de notre système, il faut évidemment réaliser autant de module émetteurs/récepteurs que vous avez de



le module codeur de l'émetteur

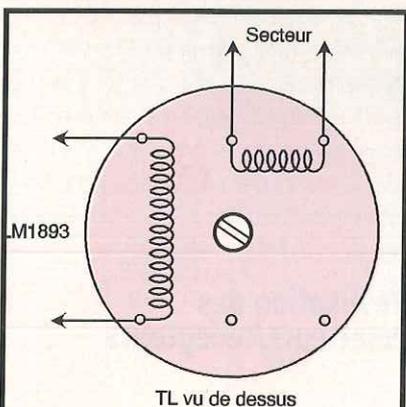


**5** Circuit imprimé du module émetteur/récepteur



**6** Implantation des composants du module émetteur/récepteur

charges distinctes à commander plus 1, puisque ce module est commun au boîtier émetteur et aux boîtiers récepteurs. Dans



**7** Brochage du transformateur TL

un premier temps, et pour que vous puissiez au moins essayer notre système, il vous faut en réaliser au minimum 2 !

Les composants utilisés sont classiques et aucun problème de disponibilité n'est à craindre. Le transformateur TL, malgré sa référence inhabituelle, est généralement disponible chez tous les revendeurs qui stockent le LM1893 car c'est son complément attitré.

Le module émetteur/récepteur prend place, transformateur d'alimentation compris, sur un circuit imprimé dont le tracé vous est présenté **figure 5**. Le montage des composants est à faire en suivant les indications de la **figure 6**, dans l'ordre classique : composants passifs puis composants actifs. Veillez à mettre en place, en premier lieu, les straps dont un passe sous le pont de redressement PT<sub>1</sub>. De ce

fait, il sera impérativement réalisé en fil isolé pour de simples raisons de sécurité.

Le transformateur TL est parfaitement symétrique mécaniquement mais ne l'est pas électriquement. Aucun symbole de détrompage n'étant visible sur son boîtier, utilisez un ohmmètre et les indications de la **figure 7** pour déterminer le sens d'implantation correct.

Le transistor T<sub>1</sub> n'a pas besoin de radiateur ; par contre, il est prudent de munir le régulateur IC<sub>2</sub> d'un petit carré de Dural en faisant office. Si le boîtier dans lequel vous intégrez le montage est métallique, vous pouvez utiliser une face de ce dernier comme radiateur pour IC<sub>2</sub>. Aucun accessoire d'isolement n'est nécessaire puisque sa languette métallique est reliée à la masse.

### Réalisation du codeur et des décodeurs

L'approvisionnement des composants ne présente aucune difficulté. Les mini-interrupteurs DIL ne sont utiles que si vous souhaitez pouvoir changer souvent et facilement les adresses de vos divers modules récepteurs. Le réseau SIL 5 x 10 kΩ des décodeurs n'étant pas courant, un espace suffisant est prévu sur le circuit imprimé pour recevoir un 7 ou 8 x 10 kΩ dont il suffit de couper les pattes sumuméraires.

Le dessin du circuit imprimé du codeur vous est proposé **figure 8** et celui du décodeur **figure 10**. Les implantations, quant à elle, vous sont indiquées en **figures 9** et **11**. Elles ne présentent aucune difficulté et sont à réaliser dans l'ordre classique : composants passifs puis composants actifs. Attention au sens des réseaux SIL dont le commun est repéré par un trait ou un point parfois assez discret.

Le raccordement entre les modules codeur, décodeurs et les modules émetteurs/récepteurs à courants porteurs peut être fait par un petit morceau de câble en nappe muni ou non de connecteurs. Notez que les différents plots de raccordement entre ces modules se font face dans le bon ordre lorsque l'on superpose les circuits imprimés. Ceci a été prévu afin de faciliter la mise en boîtier du montage si nécessaire. Le codeur est muni d'une prise DB25 mâle

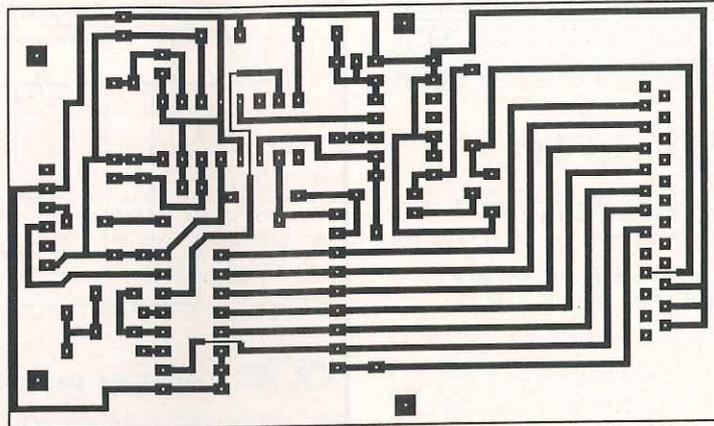
dont le brochage est conforme à celui rencontré sur les ports parallèles des compatibles PC. Vous utiliserez donc un cordon DB25 standard droit, c'est à dire encore câblé fil à fil, pour le relier à votre micro-ordinateur.

### Réglages initiaux

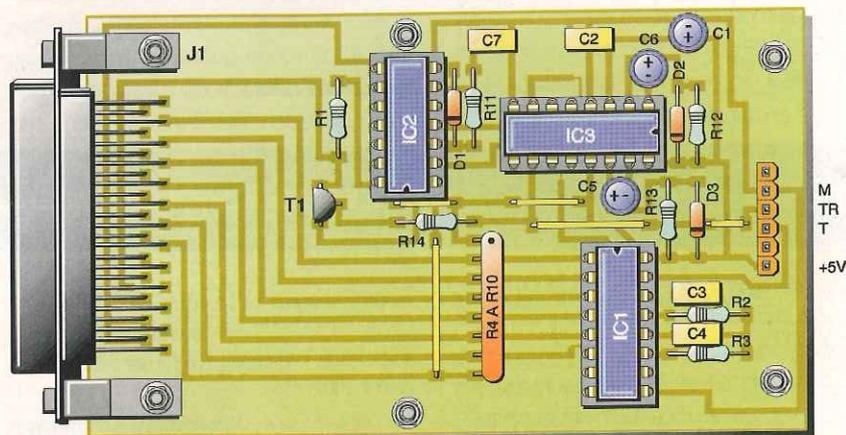
Il est nécessaire de régler, une fois pour toutes rassurez-vous, la fréquence de fonctionnement des VCO des modules émetteurs/récepteurs à courants porteurs. Ce réglage peut être fait très rapidement à l'oscilloscope, même avec un modèle très simple, mais il peut aussi être réalisé sans aucun appareil de mesure. Il faut alors faire preuve d'un peu plus de patience !

Ce réglage se déroule de la façon suivante si vous possédez un oscilloscope :

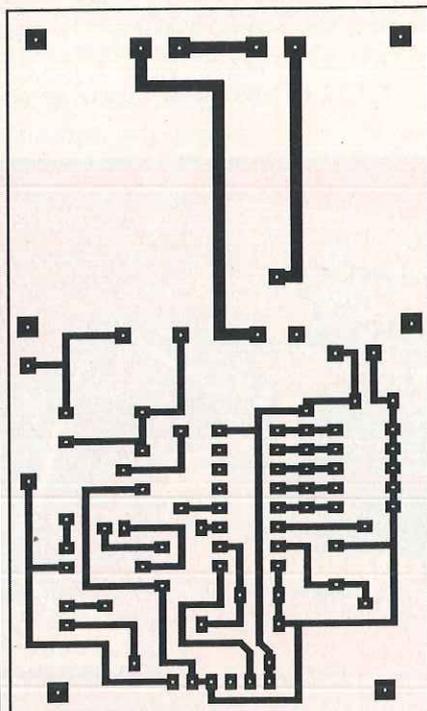
- connectez l'entrée verticale de l'oscilloscope entre le point commun de R<sub>1</sub> et C<sub>6</sub> et la masse du module émetteur/récepteur à régler,
- mettez le montage sous tension ce qui doit vous faire observer une sinusoïde,
- ajustez alors P1 pour obtenir la plus grande amplitude possible et repérez sa position,



**8** Circuit imprimé du codeur de l'émetteur

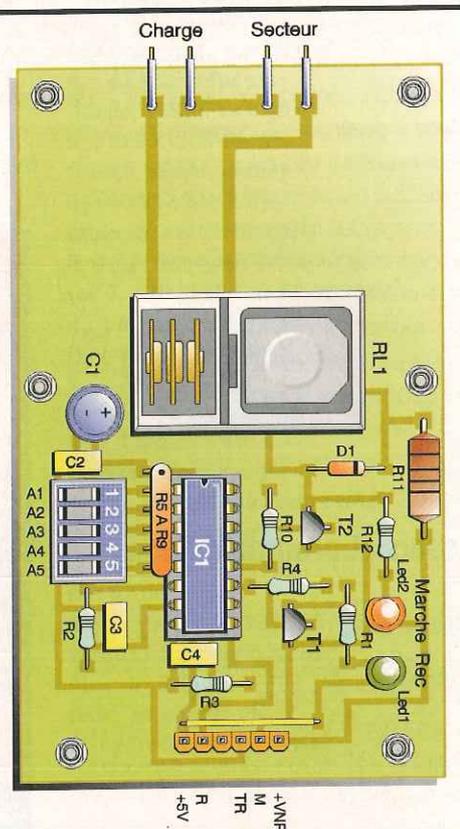


**9** Implantation des composants du codeur de l'émetteur



**10** Circuit imprimé du décodeur des récepteurs

**11** Implantation des composants du décodeur des récepteurs



- reliez ensuite T à la masse et retouchez P1 pour obtenir à nouveau la plus grande amplitude possible et repérez aussi sa position,

- placez alors P1 au milieu de la zone ainsi déterminée.

En l'absence d'oscilloscope, il faut disposer d'au moins deux modules pour procéder au réglage qui se déroule de la façon suivante :

- connectez deux modules au secteur sur des prises éloignées mais raccordées sur la même phase si votre habitation est câblée en triphasé,

- placez les deux potentiomètres P<sub>1</sub> de chaque module à mi-course,

- mettez TR d'un module à la masse afin de le placer en réception et laissez TR de l'autre en l'air afin qu'il soit en émission,

- connectez un voltmètre ou une sonde logique sur la sortie R du module placé en réception et mettez les montages sous tension,

- si vos modules arrivent à dialoguer, le voltmètre ou la sonde logique doit indiquer un niveau haut lorsque T du module placé en émission reste en l'air et un niveau bas lorsque vous reliez ce même T à la masse. Si tel n'est pas le cas, retouchez doucement, dans un sens ou dans l'autre, le potentiomètre P<sub>1</sub> d'un seul module pour y parvenir,

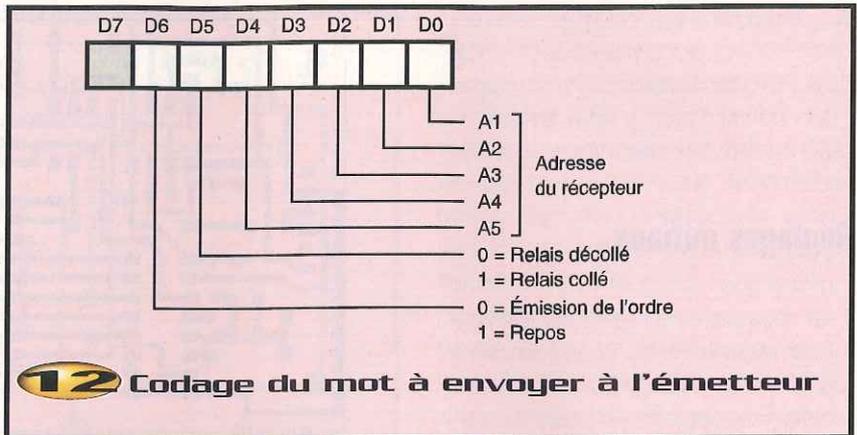
- lorsque vous avez réussi à établir la communication, continuez à manœuvrer P<sub>1</sub> de façon à déterminer la plage de sa course pour laquelle la liaison fonctionne,

- placez-le ensuite au centre de cette plage, - échangez les rôles des deux modules et procédez de la même façon pour régler le potentiomètre de l'autre module.

Ces réglages sont à réaliser une fois pour toutes puisqu'ils visent à accorder le VCO du LM1893 sur le transformateur de ligne TL. Ils doivent évidemment être réalisés sur tous les modules utilisés dans votre installation.

## Essais et programmation

Les modules codeur et décodeur ne nécessitant aucun réglage, il est possible de les raccorder aux modules émetteurs/récepteurs une fois que ces derniers ont été réglés comme expliqué ci-dessus afin de passer aux essais en vraie grandeur.



Comme nous l'avons vu ci-dessus, le codeur de l'émetteur est relié au port parallèle de n'importe quel compatible PC au moyen d'un câble droit fil à fil standard muni d'un connecteur DB25 femelle.

La programmation de l'ensemble ne présente aucune difficulté et peut être réalisée dans le langage de votre choix compte tenu de son extrême simplicité. Il suffit juste en effet de pouvoir accéder aux registres qui contrôlent le port parallèle, ce qui est possible avec n'importe quel langage sur tout compatible PC.

Le principe que doit respecter votre programme de commande est alors le suivant :

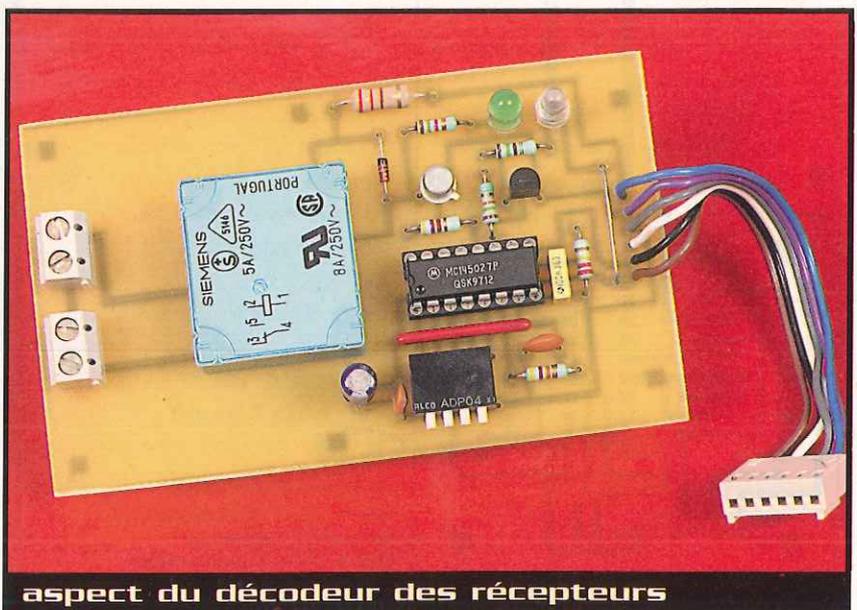
- lecture de l'état de ACK afin de s'assurer que le module n'est pas en cours d'émission. Si ACK est au niveau bas, le module est occupé et il faut alors attendre que ACK revienne au niveau haut. Si ACK est au niveau haut, un ordre peut être

envoyé,

- envoi de l'ordre par écriture dans le registre de données du port parallèle, du mot dont le principe de codage est indiqué **figure 12**. Les cinq bits de poids faibles constituent l'adresse du ou des récepteurs destinataires. Le bit D5 est à mettre à 1 pour faire coller le relais et à 0 pour le faire décoller et le bit D6 est à mettre à 0 pour envoyer l'ordre,

- écriture du même mot dans le même registre mais avec D6 à 1 pour replacer le module au repos.

A ce stade de la réalisation, votre télécommande est parfaitement opérationnelle et peut être mise en boîtier que vous choisirez librement car aucune contrainte particulière n'est à respecter. Nous vous conseillons juste, pour les boîtiers récepteurs, des modèles solidaires d'une prise de courant ce qui permet ainsi de les enfi-cher entre la prise murale et la fiche de l'ap-



pareil à commander, comme l'on fait avec les prises téléphoniques gigognes par exemple.

Si votre domicile est alimenté en triphasé, toutes les prises ne sont évidemment pas reliées à la même phase et il se peut que la communication soit impossible à établir, si l'émetteur se trouve sur une phase et des récepteurs sur une autre.

La solution à ce problème est fort simple. Il suffit de ponter vos phases en haute fréquence, après votre disjoncteur EDF, en les reliant deux à deux par des condensateurs de 47 nF de 1500V de tension de service.

Si vous ne trouvez pas de tels condensateurs chez votre revendeur habituel voyez un dépanneur télé qui dispose

généralement de telles valeurs dans ses tiroirs !

Pour finir, sachez que tous les interphones «secteur» utilisent la modulation de fréquence, parfois sur des fréquences voisines de celles de notre système. Il se peut donc qu'il soit impossible de transmettre des ordres lorsqu'un tel interphone est en marche en position émission et sur la même phase que le récepteur concerné lors de la transmission de l'ordre. Ce n'est pas une limitation bien gênante dès lors que l'on a pris conscience de son existence et de sa raison d'être.

C. TAVERNIER

## Contact

### ELECTRONIQUE PRATIQUE

est sur  
**INTERNET:**

composez  
<http://www.eprat.com>.

vos remarques etc.  
[redac@eprat.com](mailto:redac@eprat.com)

## Nomenclature

### Module émetteur/récepteur

IC<sub>1</sub> : LM1893  
IC<sub>2</sub> : 7815 (régulateur +15V boîtier T0220)  
T<sub>1</sub> : BD679, TIP110, TIP120  
PT<sub>1</sub> : pont moulé 50V 1A ou plus  
DT : diode Transil 43V (par ex. 1N6286A ou 1,5 KE 43 A)  
LED : LED quelconque  
R<sub>1</sub> : 4,7 Ω 1/2W (jaune, violet, or)  
R<sub>2</sub> : 3,3 kΩ 1/4W 5% (orange, orange, rouge)  
R<sub>3</sub>, R<sub>5</sub> : 10 kΩ 1/4W 5% (marron, noir, orange)  
R<sub>4</sub> : 5,6 kΩ 1/4W 5% (bleu, vert, rouge)  
R<sub>6</sub> à R<sub>8</sub> : 4,7 kΩ 1/4W 5% (jaune, violet, rouge)  
R<sub>9</sub> : 820 Ω 1/4W 5% (gris, rouge, marron)  
C<sub>1</sub> : 470 µF/40V chimique radial  
C<sub>2</sub>, C<sub>8</sub>, C<sub>11</sub> : 0,1 µF mylar  
C<sub>3</sub> : 560 pF céramique  
C<sub>4</sub>, C<sub>10</sub> : 10 nF mylar  
C<sub>5</sub> : 10 µF/25V chimique radial  
C<sub>6</sub> : 33 nF céramique ou mylar  
C<sub>7</sub> : 0,1 µF/220V alternatifs classe X2  
C<sub>9</sub> : 47 nF mylar  
C<sub>12</sub> : 4,7 µF/25V chimique radial  
TA : transformateur moulé 220V/2x9V/5 VA  
TL : transformateur TOKO 707 VX A042  
P<sub>1</sub> : potentiomètre ajustable vertical de 2,2 kΩ pour circuit imprimé

Porte fusible pour CI et fusible T20 de 100mA temporisé  
1 support de CI 18 pattes

### Codeur de l'émetteur

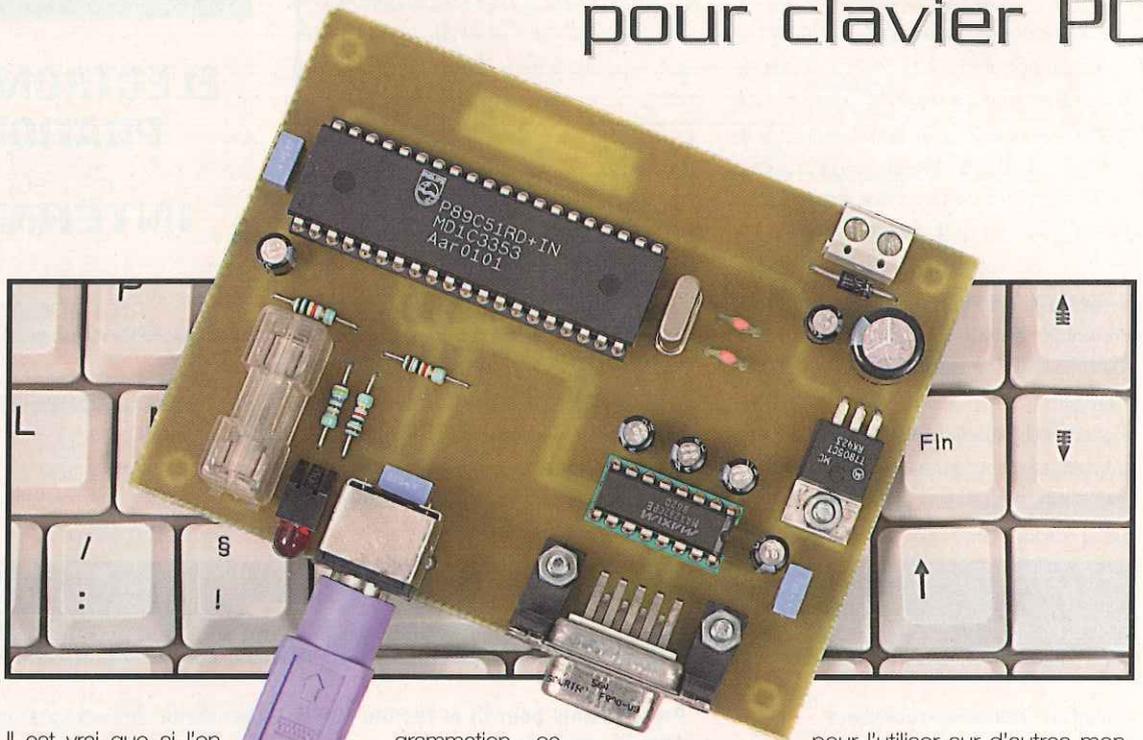
IC<sub>1</sub> : 145026 CMOS  
IC<sub>2</sub> : 4093 CMOS  
IC<sub>3</sub> : 4538 CMOS  
T<sub>1</sub> : BC547  
D<sub>1</sub> à D<sub>3</sub> : 1N914 ou 1N4148  
R<sub>1</sub> : 10 kΩ 1/4W 5% (marron, noir, orange)  
R<sub>2</sub> : 47 kΩ 1/4W 5% (jaune, violet, orange)  
R<sub>3</sub>, R<sub>12</sub> : 100 kΩ 1/4W 5% (marron, noir, jaune)  
R<sub>4</sub> à R<sub>10</sub> : Réseau SIL 7 x 10 kΩ 1 point commun  
R<sub>11</sub> : 470 kΩ 1/4W 5% (jaune, violet, jaune)  
R<sub>13</sub> : 120 kΩ 1/4W 5% (marron, rouge, jaune)  
R<sub>14</sub> : 15 kΩ 1/4W 5% (marron, vert, orange)  
C<sub>1</sub>, C<sub>5</sub>, C<sub>6</sub> : 10 µF/25V chimique radial  
C<sub>2</sub> : 10 nF céramique  
C<sub>3</sub> : 4,7 nF céramique  
C<sub>4</sub> : 470 pF céramique  
C<sub>7</sub> : 0,22 µF mylar  
1 support de CI 14 pattes  
2 supports de CI 16 pattes  
Connecteur DB25 mâle soudé pour circuit imprimé

### Décodeur des récepteurs

IC<sub>1</sub> : 145027 CMOS  
T<sub>1</sub> : BC547 CMOS  
T<sub>2</sub> : 2N2222A  
D<sub>1</sub> : 1N914 ou 1N4148  
LED<sub>1</sub> : LED verte  
LED<sub>2</sub> : LED rouge  
R<sub>1</sub> : 150 Ω 1/4W 5% (marron, vert, marron)  
R<sub>2</sub> : 47 kΩ 1/4W 5% (jaune, violet, orange)  
R<sub>3</sub> : 220 kΩ 1/4W 5% (rouge, rouge, jaune)  
R<sub>4</sub> : 15 kΩ 1/4W 5% (marron, vert, orange)  
R<sub>5</sub> à R<sub>9</sub> : Réseau SIL 5 x 10 kΩ (ou 7 x 10 kΩ) 1 point commun  
R<sub>10</sub> : 4,7 kΩ 1/4W 5% (jaune, violet, rouge)  
R<sub>11</sub> : 220 Ω 1/2W (rouge, rouge, marron)  
R<sub>12</sub> : 1,5 kΩ 1/4W 5% (marron, vert, rouge)  
C<sub>1</sub> : 10 µF/25V chimique radial  
C<sub>2</sub> : 10 nF céramique  
C<sub>3</sub> : 22 nF céramique  
C<sub>4</sub> : 0,1 µF mylar  
RL<sub>1</sub> : relais plat 12V/1RT/8A, Siemens V 23057 A ou équivalent  
Bloc de 5 (ou 4 + 1) mini-interrupteurs en boîtier DIL  
1 support de CI 16 pattes

# Interface RS232

## pour clavier PC



Aujourd'hui on peut acheter un clavier de PC pour à peine plus de 10 € (66 Fr. environ). Cela en fait donc un périphérique de saisie extrêmement bon marché. Pour s'en convaincre, il suffit de regarder le prix d'un clavier de 16 touches dans les catalogues des annonceurs. Afin de vous permettre de profiter de toute la souplesse et du confort d'un vrai clavier dans vos applications (domotiques ou autres), nous vous proposons de réaliser une petite interface RS232 pour clavier PC.

Il est vrai que si l'on ajoute le prix des composants de ce petit montage au prix du clavier, le coût de la solution globale est un peu différent de ce que nous avons annoncé, et le comparatif de prix que nous avons mentionné dans l'introduction est déjà moins favorable. Malgré tout, la solution que nous vous proposons pour disposer d'un clavier PC à votre convenance dans vos applications est très intéressante.

### Schéma

Le schéma de notre montage est reproduit en **figure 1**. Au premier coup d'œil, on constate que le cœur de ce montage est articulé autour du microcontrôleur P89C51RD+, l'un des derniers-nés de la grande famille 8051 chez PHILIPS. L'intérêt principal d'utiliser ce microcontrôleur dans notre montage réside dans sa mémoire programme de technologie FLASH et au programme de "boot" qu'il intègre en standard. Grâce à cela, les outils nécessaires à la programmation ce microcontrôleur sont extrêmement simples et peu coûteux. Le programme nécessaire à la pro-

grammation ce microcontrôleur est d'ailleurs fourni gratuitement par PHILIPS sur son site Internet. Les lecteurs qui ne connaîtraient pas déjà ce microcontrôleur sont invités à lire l'excellent article publié à ce sujet dans le n° 9 d'Interfaces PC (numéro hors série d'Électronique pratique).

C'est uniquement la simplicité de programmation qui nous a guidés dans le choix de ce microcontrôleur, car, pour le reste, ce microcontrôleur est utilisé ici comme un simple 87C52. Aucune des ressources particulières du P89C51RD+ n'a été mise en œuvre par le logiciel associé à ce montage. Il sera donc possible de remplacer le P89C51RD+ par un 87C52 sur votre maquette (si des fois vous en avez dans vos fonds de tiroirs). Ceci dit, sur un plan économique, la différence de prix entre un P89C51RD+ et un 87C52 n'est pas toujours significative. Cela dépend beaucoup du distributeur chez qui vous vous approvisionnez. Si l'écart de prix que vous constatez chez votre revendeur n'est pas trop important, préférez un P89C51RD+. Vous pourrez effacer et reprogrammer le microcontrôleur à volonté

pour l'utiliser sur d'autres montages, ce qui vaut bien un ou deux euros de plus.

Pour les impatientes qui se demandent bien à quoi peut ressembler le schéma électronique de l'interface avec un clavier de type PC, nous abordons maintenant le vif du sujet. A bien regarder la figure 1, on s'aperçoit vite il n'y a pas de quoi "fouetter un chat". En effet, un clavier de PC utilise seulement deux lignes bidirectionnelles pour dialoguer selon un protocole série synchrone.

En raison de la structure à drain ouvert des ports du microcontrôleur, il suffit simplement de relier les broches 1 et 5 du connecteur CN<sub>2</sub> à un port de U<sub>1</sub>. Les résistances R<sub>1</sub> et R<sub>2</sub> viennent en parallèle des résistances internes du microcontrôleur, pour accélérer le temps de montée des signaux. Enfin, notre montage devra fournir 5VDC pour alimenter le clavier. Plutôt que de relier directement le potentiel VCC à la broche 4 de CN<sub>2</sub>, nous avons préféré ajouter un fusible pour protéger le montage. Nous avons pensé que certains d'entre vous profiteront de l'occasion pour ressortir des vieux claviers de leurs greniers et que l'état de fonc-



Les pastilles seront percées à l'aide d'un foret de 0,8 mm de diamètre, pour la plupart. En ce qui concerne CN<sub>1</sub>, D<sub>1</sub> et REG<sub>1</sub>, il faudra percer les pastilles avec un foret de 1mm de diamètre. En ce qui concerne CN<sub>2</sub>, il faudra percer les pastilles de passage des ergots avec un foret de 2,5 mm de diamètre. Enfin, n'oubliez pas de percer le trou de passage des vis du connecteur CN<sub>2</sub> et du régulateur REG<sub>1</sub> avec un foret de 3,5 mm de diamètre.

Comme d'habitude, procurez-vous les composants avant de réaliser les circuits imprimés, pour vérifier que l'implantation est possible. Cette remarque concerne essentiellement le connecteur mini-DIN CN<sub>2</sub>.

Notez que vous pouvez éventuellement faire l'économie du connecteur CN<sub>2</sub> si vous décidez de couper le câble du clavier pour souder les fils directement sur le circuit imprimé de notre montage.

Dans ce cas de figure, il peut être utile d'immobiliser le câble à l'aide d'une attache rapide pour éviter que les fils ne se cassent au fil de temps, à cause des manipulations du clavier et des tensions mécaniques que cela générerait sur les fils.

Pour le reste, il n'y a pas de difficulté particulière pour l'implantation. Soyez tout de même attentifs au sens des condensateurs et des circuits intégrés et respectez bien la nomenclature. Si vous avez choisi d'utiliser un microcontrôleur P89C51RD+, il sera utile de le monter sur un support pour vous permettre de le démonter facilement en vue de l'utiliser sur d'autres montages par la suite.

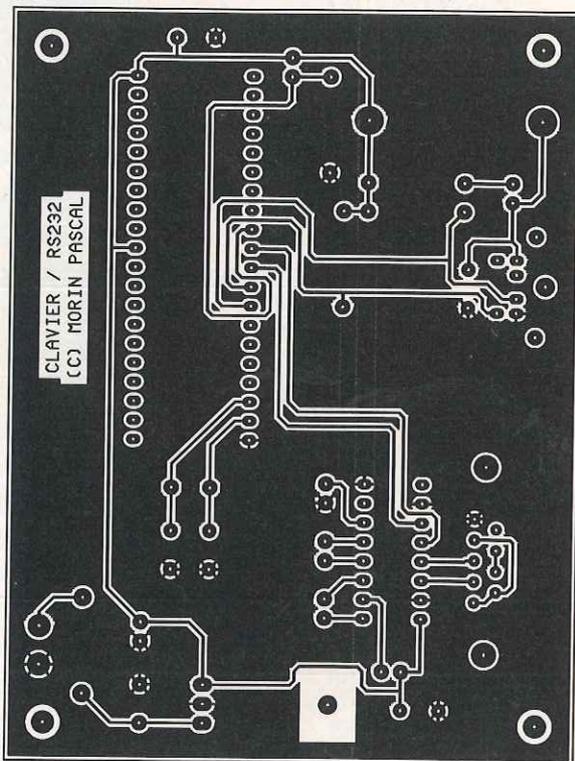
Veillez bien à choisir un connecteur femelle pour CN<sub>3</sub>. Car un modèle mâle s'implante parfaitement, mais les points de connexions se retrouvent inversés par symétrie par rapport à l'axe vertical. Dans ce cas, il n'y a aucune chance pour que le montage dialogue avec un PC, à moins de fabriquer un câble spécial qui rétablirait l'ordre voulu. En ce qui concerne le câble nécessaire pour relier notre montage au port série d'un PC, il sera important d'utiliser un câble de bonne qualité et d'une longueur inférieure à 1,5m car le montage dialogue à la vitesse de 38400 Bauds. Cette vitesse de communication a été retenue en raison du temps de réaction nécessaire au microcontrôleur pour vider son buffer avant saturation (qui ne s'est jamais amusé

à appuyer plusieurs touches du clavier en même temps ?).

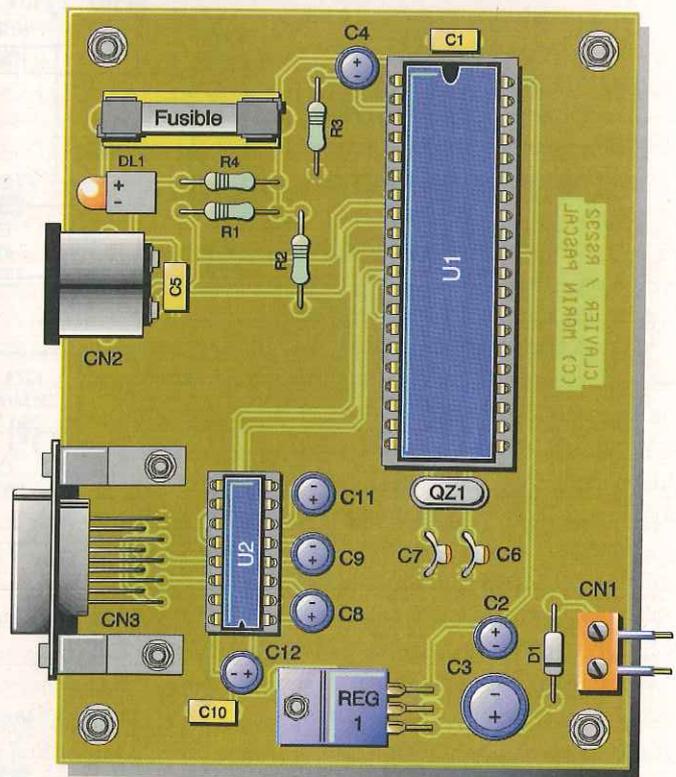
Le microcontrôleur sera programmé avec le contenu d'un fichier que vous pourrez vous procurer par téléchargement sur le serveur Internet de la revue (eprat.com).

Le fichier KDBRS232.ROM est le reflet binaire du programme à implanter dans le microcontrôleur tandis que le fichier KDBRS232.HEX correspond au format HEXA INTEL. Selon le modèle de programmeur dont vous disposez, vous utiliserez l'un ou l'autre des fichiers (vous utiliserez le fichier KDBRS232.HEX avec le programme WinISP de PHILIPS). Si vous n'avez pas la possibilité de télécharger les fichiers programmes, vous pourrez envoyer une demande à la rédaction en joignant une disquette formatée accompagnée d'une enveloppe self-adressée convenablement affranchie (tenir compte du poids de la disquette).

L'utilisation du montage est quasiment instantanée. Il vous suffit de brancher un clavier de type PC dans le connecteur CN<sub>2</sub> et de relier le connecteur SUBD9 à un port série de votre PC. Les paramètres de communication de la liaison série sont les



**2** Tracé du circuit imprimé



**3** Implantation des éléments

suivants : 38400 bauds, 8 bits de données, 1 bit de stop et pas de parité ni de contrôle de flux. A la mise sous tension, le montage émet un message d'accueil sur la liaison série, puis il envoie le code ASCII de toutes les touches qui seront appuyées sur le clavier branché à la maquette.

Notre programme gère l'essentiel des touches. Cependant, quelques combinaisons ne sont pas traitées. Il faut savoir qu'un clavier de type PC est un périphérique totalement dénué de tout sens pratique.

Seul, il ne sait rien faire. Par exemple, lorsque vous appuyez sur la touche SHIFT d'un clavier, ce dernier se moque complètement de la signification de la touche. Il transmet (habituellement au PC, appelé le "HOST" dans la documentation des claviers) un code qui indique seulement l'emplacement physique de la touche appuyée dans la matrice. Puisque notre montage remplit ici le rôle du "HOST", c'est notre programme qui doit interpréter toutes les actions particulières que l'on demande habituellement à un clavier sans même y prêter attention (utilisation de la touche ALT-GR par exemple). Bien entendu, dans le programme de notre montage, nous avons pris en charge les combinaisons de touches indispensables (SHIFT, CAPS LOCK et ALT-GR). Cependant, nous nous sommes cantonnés à gérer uniquement les combinaisons qui correspondent à un code ASCII (presque dans leur totalité).

Il faut savoir que pour la gestion des états du clavier contrôlé par le montage, le programme doit manipuler de nombreuses tables de correspondances. L'auteur s'est donc contenté de constituer les tables qu'il a jugées indispensables afin que l'utilisation de ce montage ne soit pas trop restrictive. Les exceptions correspondent à l'utilisation des voyelles avec un chapeau ou un tréma. Cela reste acceptable.

Les autres touches, qui n'ont pas un équivalent ASCII, ne sont pas gérées par notre programme. C'est le cas par exemple des touches de direction ou de contrôle (pause, impression écran, etc.). En revanche les touches de fonctions F1 à F12 sont tout de même traitées par le programme. Bien entendu, il a fallu déci-

Fonction	Code à envoyer au montage
Allumer la LED 'Arrêt Défilement'	'0'
Éteindre la LED 'Arrêt Défilement'	'1'
Allumer la LED 'Verr Num'	'2'
Éteindre la LED 'Verr Num'	'3'
Allumer la LED 'Caps Lock'	'4'
Éteindre la LED 'Caps Lock'	'5'
Remise à zéro du clavier	'*'
Passer en mode 'Debug'	'!'
Sortir du mode 'Debug'	'\$'

**4 Liste des commandes du montage**

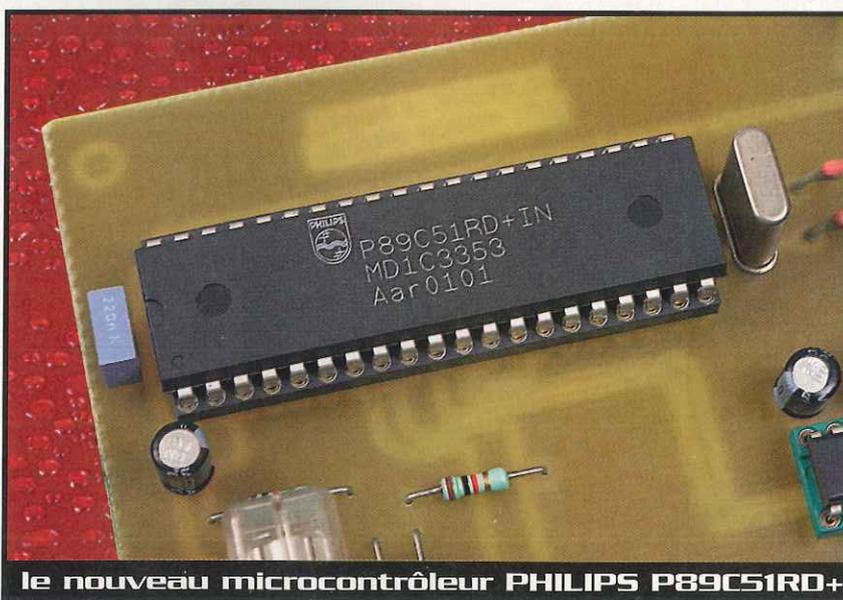
der arbitrairement du code ASCII qui leur est associé. Nous avons simplement choisi d'associer aux touches de fonction le code ASCII qui correspond au numéro de chacune des touches. Par exemple, à la touche F1 correspond le code 01H et à la touche F10 correspond le code 0AH. Cette simplicité pose cependant un petit problème puisque, du coup, la touche F7 (qui retourne le code 07H) est confondue avec la fonction 'Ring Bell' (comprenez : faire retentir un coup de sonnette) et que la touche F9 est confondue avec la touche 'TAB'.

De même, la touche F10 est confondue avec la touche 'RETOUR'. Mais nous pensons que cette confusion n'est pas trop gênante dans la mesure où elle ne concerne que peu de fonction et que la gêne n'est réelle que si vous avez besoin d'utiliser toutes les combinaisons des touches d'un clavier PC. Ajoutons à cela que l'utilisation de la touche ALT suivie

d'un code numérique (en base décimale) pour générer un code ASCII n'est pas supportée par le programme de notre montage.

La liaison avec un clavier de PC est bidirectionnelle car le "HOST" peut également transmettre des commandes au clavier. Notre petit montage gère certaines de ces fonctions. Si le montage reçoit le caractère "\*", via la liaison série, il enverra au clavier une commande de remise à zéro. Le programme de notre montage gère automatiquement l'état du clavier en conséquence. Par exemple, si le clavier est verrouillé en majuscule (via la touche CAPS LOCK) la remise à zéro provoquera un retour à une gestion en minuscule.

En réalité, c'est notre montage qui mémorise les états de fonctionnement demandés par l'utilisateur, car le clavier ne sait rien faire tout seul. Pour le clavier, les LED et les touches n'ont aucune signification particulière. Fort heureuse-



**le nouveau microcontrôleur PHILIPS P89C51RD+**

ment, le programme de notre petit montage se charge de cette gestion, afin que vous retrouviez le comportement habituel d'un clavier de PC (en fait, le programme de notre montage réalise à peu près le même travail que le "HOST" de clavier qui est implanté sur la carte mère de votre PC).

Toutes ces explications ont pour but de vous préparer à accepter l'idée que les témoins lumineux 'Arrêt du défilement' et 'Verr num' n'ont aucun sens particulier (ni pour le clavier, ni pour notre montage).

Si vous envoyez le caractère '0' via la liaison, le témoin lumineux 'Arrêt du défilement' va s'allumer, tandis que si vous envoyez le caractère '1' via la liaison série, le témoin lumineux 'Arrêt du défilement' va s'éteindre.

Comme le montre le tableau de la **figure 4**, vous pouvez utiliser les témoins lumineux comme bon vous semble, mais cela ne modifiera pas le comportement du clavier.

C'est ainsi que vous découvrirez (peut être avec stupéfaction) qu'avec notre montage les touches du pavé numérique retournent toujours le code ASCII des chiffres associés, que le témoin lumineux 'Verr num' soit allumé ou non (souvenez-vous que notre montage ne gère pas les touches de déplacement).

Les curieux pourront observer les codes bruts que retourne vraiment le clavier connecté au montage en activant la fonction 'Debug'.

Pour cela, il faut envoyer le caractère '!' au montage, via la liaison série. Pour désactiver ce mode, il suffit d'envoyer le caractère '\$' au montage, via la liaison série, ou d'utiliser la commande de remise à zéro.

Malgré les quelques limitations que nous venons de mentionner, nous sommes persuadés que vous trouverez ce montage très utile et que, très vite, vous mettrez au panier vos vieux claviers 16 touches, tellement vous préférerez le confort de taper sur un vrai clavier.

P. MORIN

## 5 Vue d'écran



**l'incontournable driver de lignes MAX232**

## Nomenclature

**CN<sub>1</sub>** : bornier à vis 2 contacts (pas 5,08 mm)

**CN<sub>2</sub>** : connecteur mini-DIN 6 points (connecteur pour clavier PC)

**CN<sub>3</sub>** : connecteur SubD 9 points, femelle, sorties coudées, à souder sur circuit imprimé (ex. : réf. HARTING 09 66 112 7601)

**C<sub>1</sub>, C<sub>5</sub>, C<sub>10</sub>** : 220 nF

**C<sub>2</sub>, C<sub>4</sub>, C<sub>6</sub>, C<sub>9</sub>, C<sub>11</sub>, C<sub>12</sub>** : 10 µF/25V sorties radiales

**C<sub>3</sub>** : 470 µF/25V sorties radiales

**C<sub>8</sub>, C<sub>7</sub>** : 33 pF céramique au pas de 5,08 mm

**DL<sub>1</sub>** : diode LED rouge 3 mm

**D<sub>1</sub>** : 1N4001 (diode de redressement 1A/100V)

**F<sub>1</sub>** : fusible 0,1A (5x20 mm) + port fusible à souder sur circuit imprimé

**QZ<sub>1</sub>** : quartz 12 MHz en boîtier HC49/U

**REG** : régulateur LM7805 (5V) en boîtier TO220

**R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>** : 10 kΩ 1/4W 5% (marron, noir, orange)

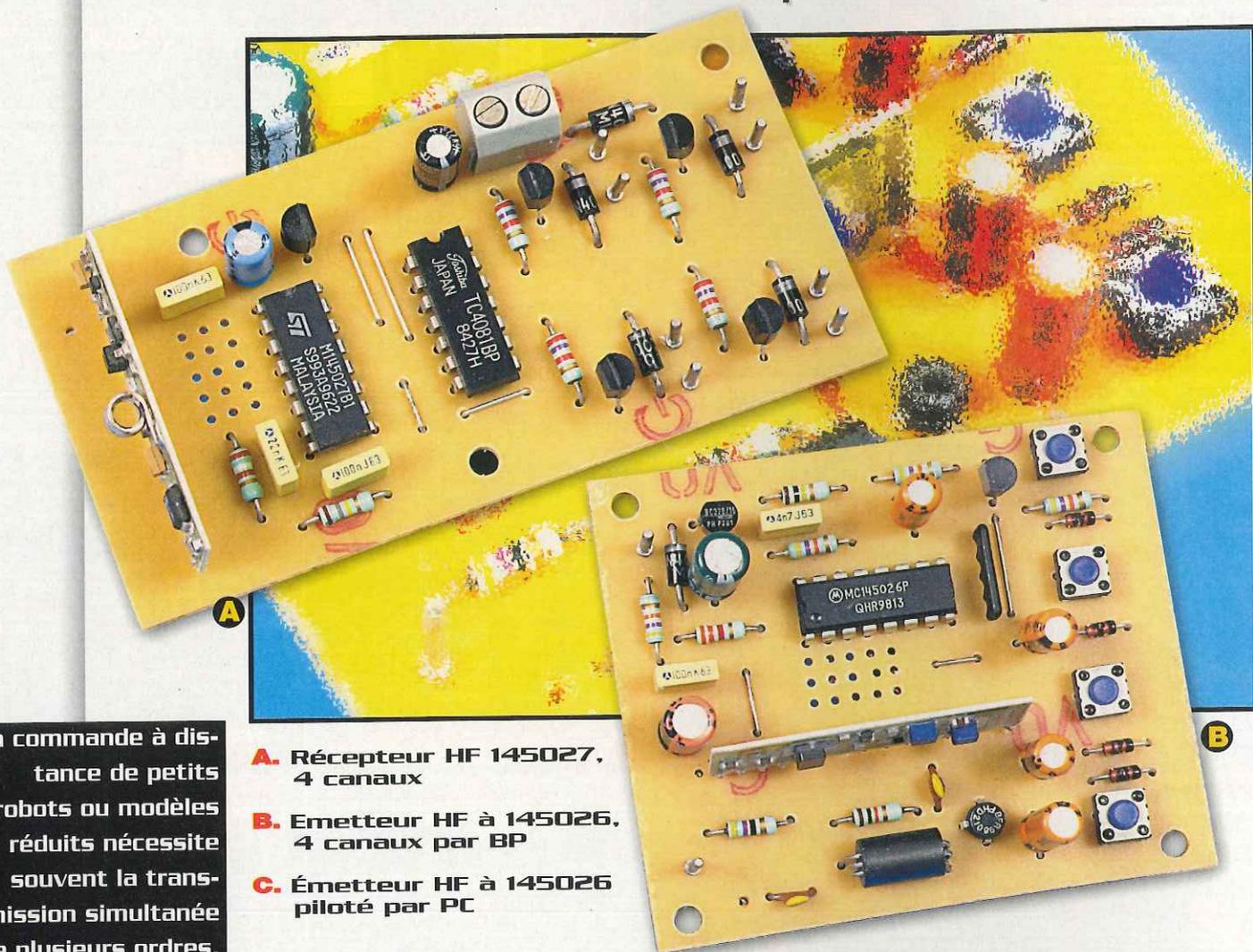
**R<sub>4</sub>** : 470 Ω 1/4W 5% (jaune, violet, marron)

**U<sub>1</sub>** : P89C51RD+ (voir le texte)

**U<sub>2</sub>** : driver de lignes MAX232

# Radiocommande 4 canaux

simultanés pilotée par PC



A

**A. Récepteur HF 145027, 4 canaux**

**B. Émetteur HF à 145026, 4 canaux par BP**

**C. Émetteur HF à 145026 piloté par PC**

B

La commande à distance de petits robots ou modèles réduits nécessite souvent la transmission simultanée de plusieurs ordres.

Le récepteur HF proposé rassemble 4 canaux radiocommandés soit à partir d'un compatible PC, soit par une petite télécommande d'appoint. Par ailleurs, l'émetteur PC est capable de radiocommander 16 récepteurs indépendants, tel que celui décrit dans notre revue n°247.

Le décodeur 145027 de MOTOROLA ou THOMSON (préfixe MC ou M) présente la particularité de pouvoir extraire un mot binaire de 4 bits à partir d'un code délivré par l'encodeur 145026 qui lui est généralement associé. Les 4 bits du mot binaire identifié par le 145027 dépendent du paramétrage des quatre entrées de codage E6 à E9 de l'encodeur 145026. Cette particularité permet d'obtenir facilement 4 canaux de télécommande, chacun des canaux correspondant à l'un des bits du mot binaire en question (en théorie, 24=16 canaux sont réali-

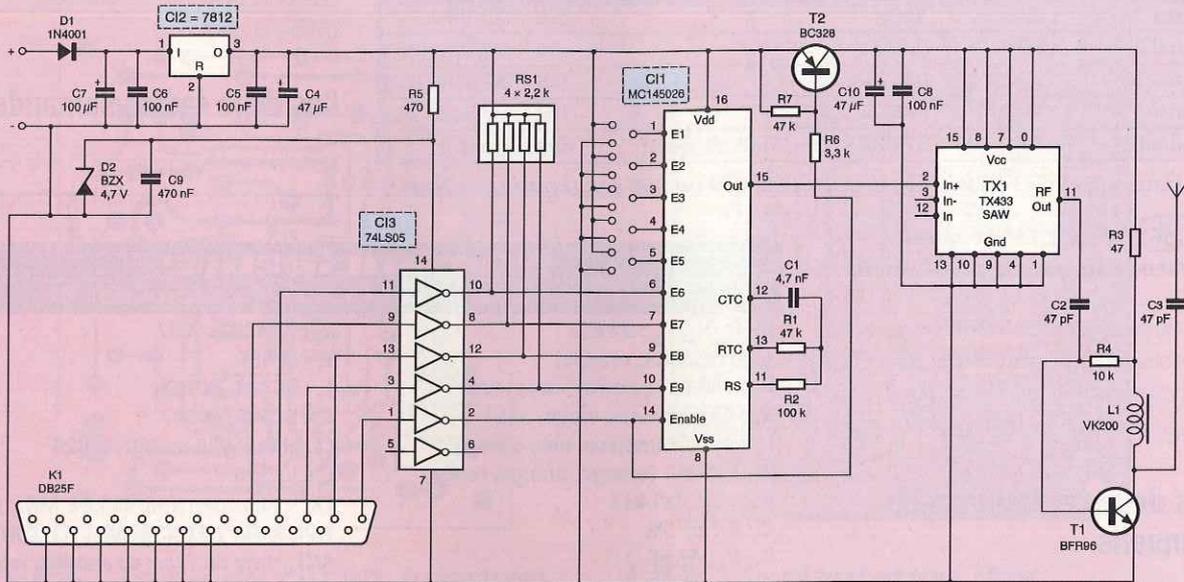
sables sans simultanéité). Associé à des modules HF fiables et économiques, ce dispositif de codage permet la réalisation d'une radiocommande simple et très bon marché de 4 canaux simultanés. Deux émetteurs de télécommandes sont proposés : l'un est une radiocommande manuelle à boutons poussoirs, l'autre est un émetteur HF contrôlé par un compatible PC via son interface parallèle.

Bien que ces radiocommandes aient été conçues pour fonctionner avec la carte récepteur 4 canaux équipée d'un 145027, présentée dans cet article, elles

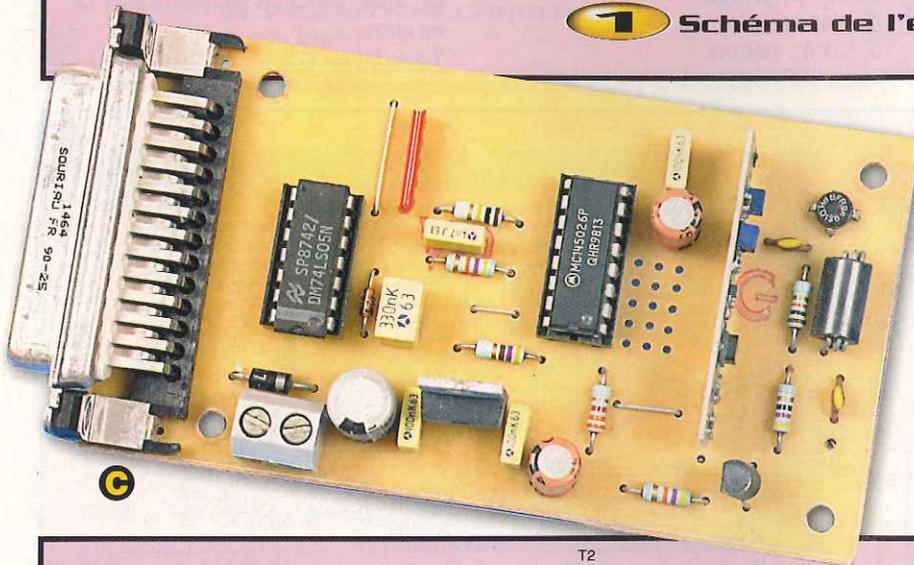
peuvent également commander un récepteur mono-canal à base de 145028, tel que le relais HF décrit dans le n°247 p60. Seize récepteurs indépendants pourront alors être commandés à distance par votre PC.

## Les émetteurs de télécommande

Bien entendu, un MC145026 ou M145026 est au cœur de la radiocommande pilotée par PC (**figure 1**) et de la télécommande HF manuelle (**figure 2**). Les entrées E1 à E9 de cet encodeur 145026

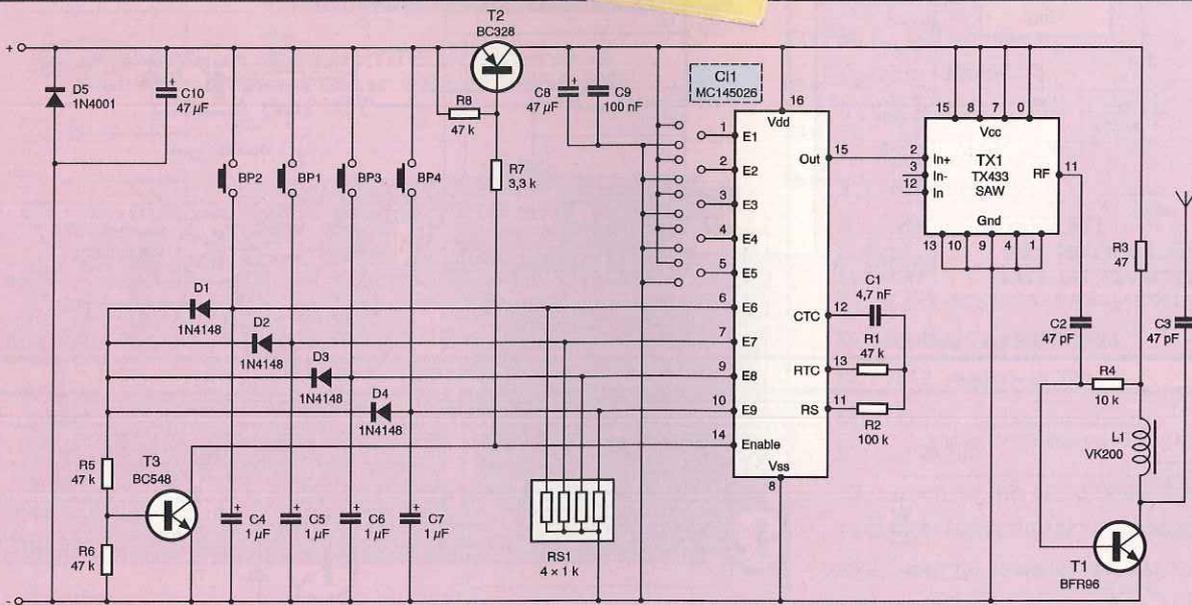


1 Schéma de l'émetteur piloté par PC



sont à trois états, selon qu'elles sont à 0, 1 ou laissées en l'air, ce qui permet 39=19683 combinaisons possibles, dont 35=243 sont utilisables pour différencier plusieurs télécommandes. Les composants R<sub>1</sub>, C<sub>1</sub> et R<sub>2</sub> réalisent avec des inverseurs logiques internes à CI<sub>1</sub> un oscillateur astable. Afin d'obtenir une meilleure portée, un étage HF complémentaire, construit autour du transistor T<sub>1</sub>, augmente la puissance d'émission. En contre partie et du fait de la polarisation du transistor UHF, le courant consommé

C



2 Schéma de principe 4 canaux par BP

Fonction	DB25	Désignation	Mise à 0 par	Mise à 1 par
Canal 1	N°5	DATA4	-	Out ad_data,247
Canal 2	N°4	DATA3	-	Out ad_data,251
Canal 3	N°2	DATA1	-	Out ad_data,254
Canal 4	N°3	DATA2	-	Out ad_data,253
Validation	N°1	STROBE	Out ad_out,1	Out ad_out,0

Ad\_data = 888, 632 ou 956 et ad\_out = 890, 634 ou 958 selon port parallèle

### 3 Fonction des lignes du port parallèle

au repos n'est plus négligeable et un transistor PNP, T<sub>2</sub>, assure la commutation de l'alimentation des étages HF lorsqu'une émission est demandée.

### Cas de la télécommande manuelle

Pour la télécommande à boutons poussoirs, le fait d'appuyer sur un BP entraîne la conduction du transistor T<sub>3</sub>. Dès lors, la broche 14 de CI<sub>1</sub> est à un niveau logique

bas et la sortie 15 délivre alors un signal binaire codé.

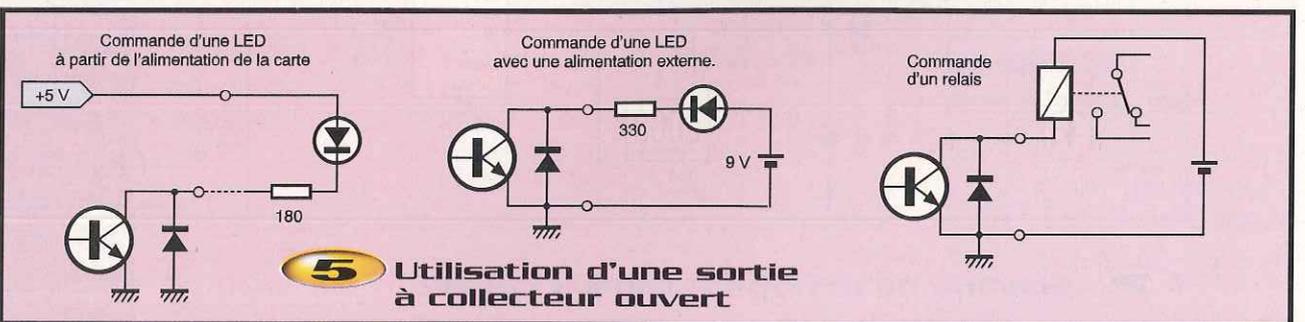
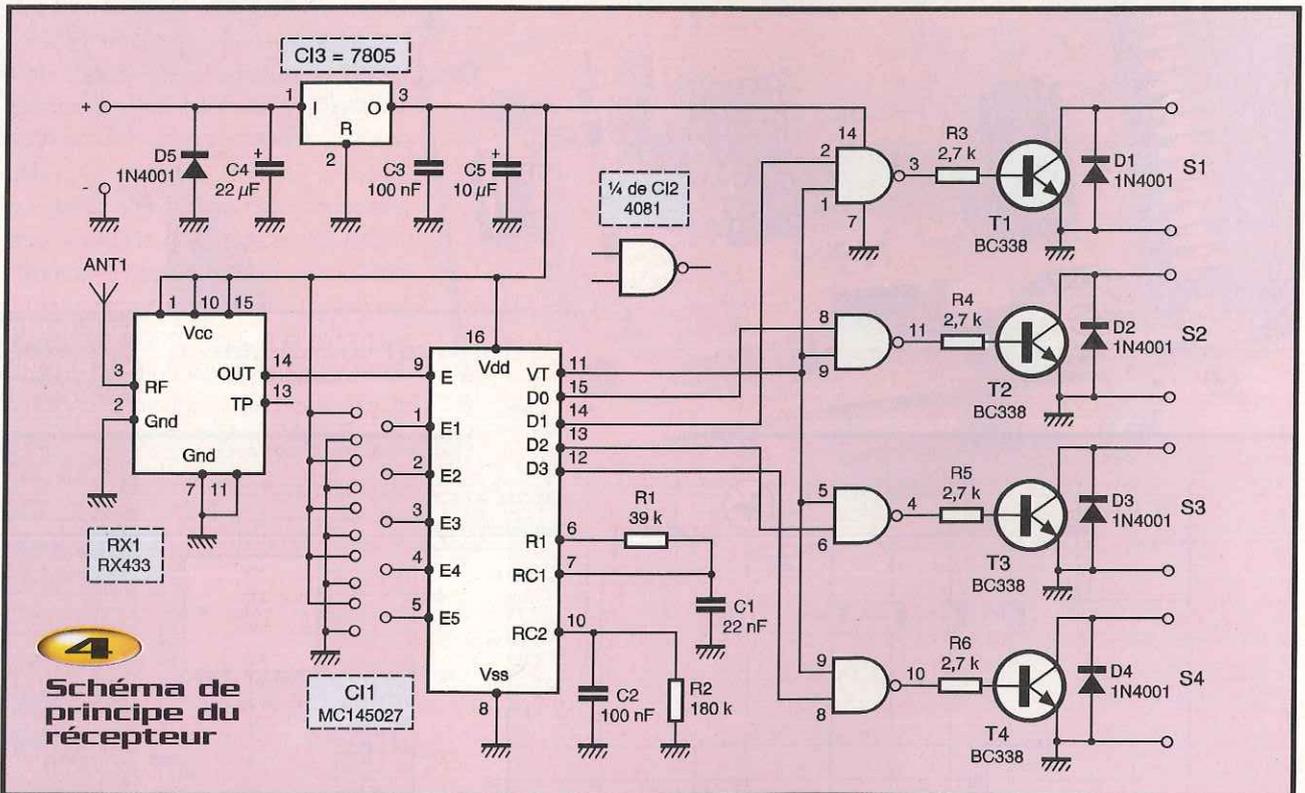
### Cas de la radiocommande PC

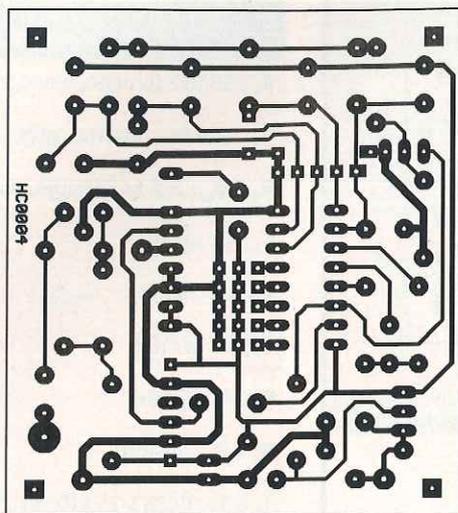
Pour la radiocommande PC, les six inverseurs à collecteur ouvert contenus dans

## Nomenclature

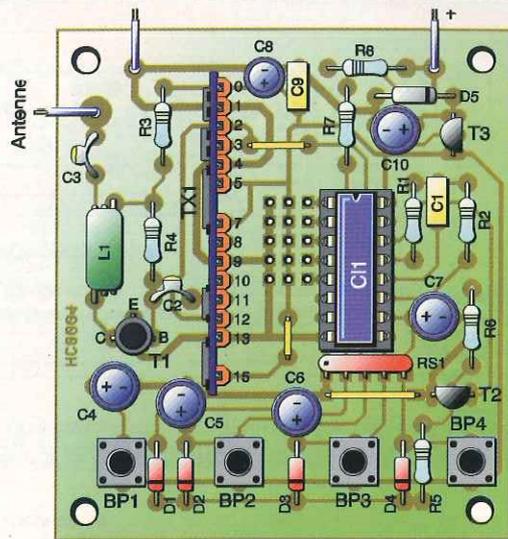
de l'émetteur HF à 145026, 4 canaux par BP

R <sub>1</sub> , R <sub>5</sub> , R <sub>6</sub> , R <sub>8</sub> : 47 kΩ (jaune, violet, orange)	D <sub>5</sub> : 1N4001..4007
R <sub>2</sub> : 100 kΩ (marron, noir, jaune)	T <sub>1</sub> : BFR96
R <sub>3</sub> : 47 Ω (jaune, violet, noir)	T <sub>2</sub> : BC328, BC327
R <sub>4</sub> : 10 kΩ (marron, noir, orange)	T <sub>3</sub> : BC548, BC547
R <sub>7</sub> : 3,3 kΩ (orange, orange, rouge)	CI <sub>1</sub> : M145026 ou MC145026
RS <sub>1</sub> : SIL 4x1 kΩ	L <sub>1</sub> : VK200
C <sub>1</sub> : 4,7 nF 5%	TX : émetteur AM 433,92 MHz, AUREL TX433SAW (SELECTRONIC) TELECONTROLLI
C <sub>2</sub> , C <sub>3</sub> : 47 pF	ANT <sub>1</sub> : brin de 17cm ou antenne fouet 1/4 d'onde
C <sub>4</sub> à C <sub>7</sub> : 1 μF	BP <sub>1</sub> à BP <sub>4</sub> : boutons poussoirs pour CI ou autre
C <sub>8</sub> , C <sub>10</sub> : 47 μF/16V	1 pile 9V ou 12V
C <sub>9</sub> : 100 nF	
D <sub>1</sub> à D <sub>4</sub> : 1N4148	

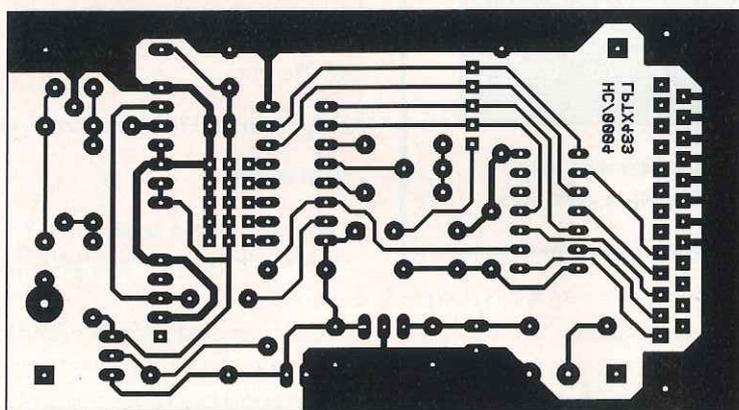




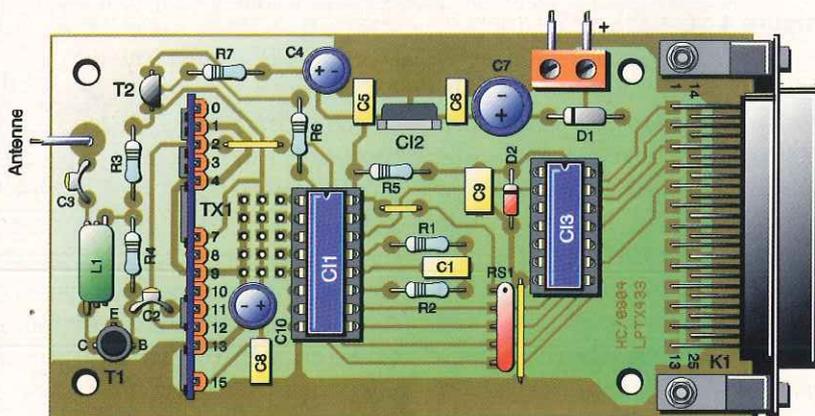
**6** Tracé du circuit imprimé de l'émetteur HF à 145026, 4 canaux par BP



**9** Implantation des éléments de l'émetteur HF à 145026, 4 canaux par BP



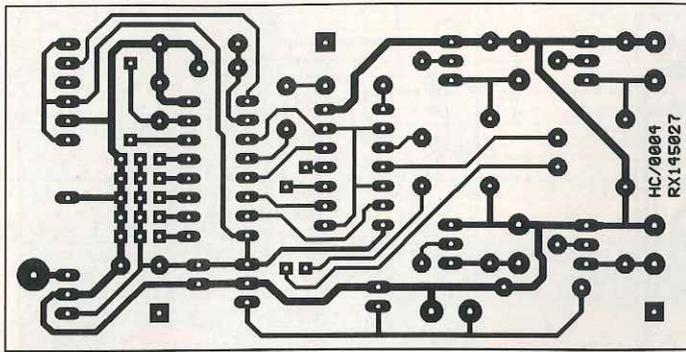
**7** Tracé du circuit imprimé de l'émetteur HF à 145026 piloté par PC



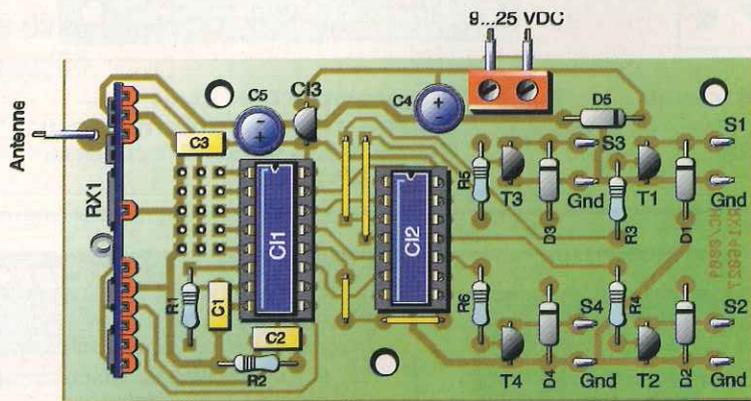
**10** Implantation des éléments de l'émetteur HF à 145026 piloté par PC

## Nomenclature de l'émetteur HF à 145026 piloté par PC

- R<sub>1</sub>, R<sub>7</sub> : 47 kΩ (jaune, violet, orange)
- R<sub>2</sub> : 100 kΩ (marron, noir, jaune)
- R<sub>3</sub> : 47 Ω (jaune, violet, noir)
- R<sub>4</sub> : 10 kΩ (marron, noir, orange)
- R<sub>5</sub> : 470 Ω (jaune, violet, marron)
- R<sub>6</sub> : 3,3 kΩ (orange, orange, rouge)
- RS<sub>1</sub> : SIL 1+4x4,7 kΩ
- C<sub>1</sub> : 4,7 nF 5%
- C<sub>2</sub>, C<sub>3</sub> : 47 pF
- C<sub>4</sub>, C<sub>10</sub> : 47 μF/16V
- C<sub>5</sub>, C<sub>6</sub>, C<sub>8</sub> : 100 nF
- C<sub>7</sub> : 100 μF/25V
- C<sub>9</sub> : 330 à 470 nF
- D<sub>1</sub> : 1N4001..4007
- D<sub>2</sub> : zéner 4,7V, BZX55C4V7
- T<sub>1</sub> : BFR96
- T<sub>2</sub> : BC328, BC327
- Cl<sub>1</sub> : M145026 ou MC145026
- Cl<sub>2</sub> : 7812, régulateur 12V
- Cl<sub>3</sub> : 74LS05
- L<sub>1</sub> : self VK200
- TX<sub>1</sub> : émetteur AM 433,92 MHz, AUREL TX433SAW (SELECTRONIC) TELECONTROLLI
- ANT<sub>1</sub> : brin de 17cm ou antenne fouet 1/4 d'onde
- K<sub>1</sub> : embase DB25 femelle
- 1 bornier 2 plots à souder



**8** Tracé du circuit imprimé du récepteur HF 145027, 4 canaux



**11** Implantation des éléments du récepteur HF 145027, 4 canaux

un 74LS05 sont utilisés comme tampon entre le port parallèle du PC et l'encodeur 145026. Cette interface logique permet, outre une certaine isolation entre les éléments actifs de la radiocommande et le PC, la remise en forme des signaux issus du PC et l'adaptation au circuit logique de la radiocommande. Le réseau de résistance  $RS_1$  force, à un niveau logique haut d'environ 12V, les entrées E6 à E9 de l'encodeur 145026 lorsqu'un étage de sortie à collecteur ouvert du 74LS05 est bloqué, ce qui est le cas lorsqu'un niveau bas est appliqué sur l'entrée d'un inverseur. Le tableau de la **figure 3** détaille la fonction des lignes du port parallèle destinées au contrôle de la radiocommande PC. La tension d'alimentation du 74LS05 est régulée à 5V par la diode zéner  $D_2$ , dont le courant inverse est limité par la résistance  $R_5$ . Le découplage de cette alimentation 5V est assuré par le condensateur  $C_6$ . L'alimentation générale de 12V de cette interface PC, devant

délivrer au moins 100mA, est fournie par un régulateur de tension fixe 7812. Les condensateurs  $C_4$  à  $C_{10}$  assurent le filtrage et le découplage de l'alimentation des différents circuits.

### Le schéma du récepteur

La **figure 4** présente le schéma de principe du récepteur quatre canaux. Le module  $RX_1$ , récepteur AM 433,92 MHz, démodule le signal modulé en amplitude par l'émetteur AM et délivre sur sa broche 2, un code sériel binaire. Le circuit intégré 145027 décode ce mot binaire en le comparant au paramétrage de ses cinq entrées E1 à E5. Comme pour le 145026 et le 145028, ces entrées sont à trois états et peuvent être à un 0 ou à un 1 logique, ou encore laissées en l'air. Le code reçu est de neuf bits. Les cinq premiers bits correspondent au paramétrage du 145027. S'ils sont corrects, la sortie VT de  $C1$  passe à 1 pour indiquer que la

## Nomenclature

Récepteur HF 145027,  
4 canaux

$R_1$  : 39 k $\Omega$  (orange, blanc, orange)

$R_2$  : 180 k $\Omega$  (marron, gris, jaune)

$R_3$  à  $R_6$  : 2,7 k $\Omega$  (rouge, violet, rouge)

$C_1$  : 22 nF 5%

$C_2, C_3$  : 100 nF

$C_4$  : 22  $\mu$ F/10V

$C_5$  : 10  $\mu$ F/50V

$D_1$  à  $D_5$  : 1N4001..7

$T_1$  à  $T_4$  : BC337-25, BC338-25

$CI_1$  : M145027 ou MC145027

$CI_2$  : 4081

$CI_3$  : 78L05, régulateur 5V

$RX_1$  : récepteur AM 433,92 MHz, AUREL

TÉLÉCONTROLLI

NB-1M, etc.

$ANT_1$  : brin de 17cm ou antenne fouet  
quart d'onde

1 bornier 2 plots à souder

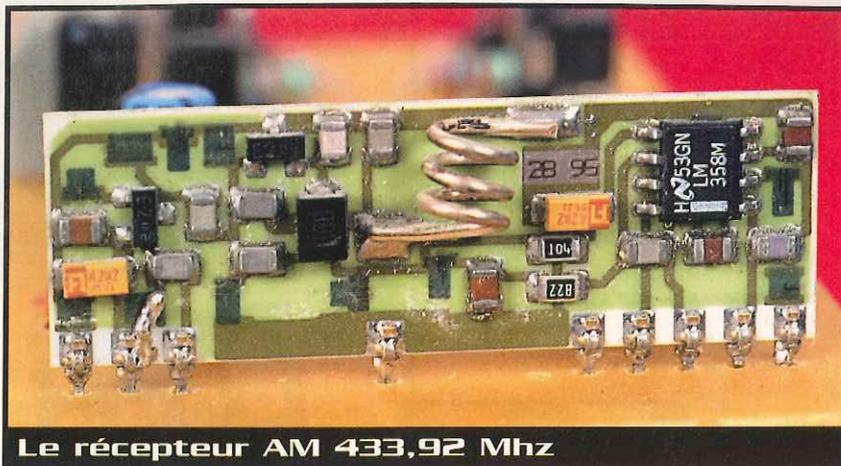
base du code est bonne. Les quatre autres bits émis sont reportés sur les sorties D0, D1, D2, D3 de  $CI_1$  et caractérisent le ou les boutons poussoirs sollicités ou la donnée établie par le port parallèle du PC. L'état logique des sorties D0 à D3 reste inchangé entre deux impulsions VT. Chacune de ses sorties est, par conséquent, associée au signal VT par une porte ET (AND). Ainsi, la sortie d'une porte AND restera à 1, seulement tant que le bouton poussoir qui lui correspond est appuyé ou tant que la sortie "STROBE" du port parallèle du PC est à un niveau logique haut. En effet, une sortie est au niveau logique haut si, au niveau de l'émetteur, l'entrée correspondante est à 1 ou laissée en l'air. Dans le cas contraire (entrée du 145026 de l'émetteur à 0), la sortie correspondante du 145027 est à 0.

A chaque canal est associé un transistor à collecteur ouvert, lequel pourra être utilisé en guise d'interrupteur commandé

basse tension (35Vmax/100mA). La **figure 5** présente quelques utilisations d'un transistor à collecteur ouvert.

## La réalisation

Le tracé des pistes des trois cartes proposées apparaît respectivement en **figure 6** pour la télécommande HF manuelle, en **figure 7** pour la radio-commande pilotée par PC et en **figure 8** pour le récepteur quatre canaux simultanés. De même, les **figures 9, 10 et 11** présentent l'implantation des



Le récepteur AM 433,92 Mhz

DEBUT:

REM initialisation du port

out 888,255:out 890,1:N=0:A=0:B=0:C=0:D=0

cls

locate 1,30:color 0,7,0:print "ELECTRONIQUE PRATIQUE":color 7,0,0

locate 3,21:print "Radiocommande PC de 4 canaux simultanés"

locate 24,18:print "Pour quitter le programme, appuyez sur [Q].";

locate 12,1:print "Entrez une séquence de 1 à 4 chiffres, par exemple 123"

locate 13,1:print "pour sélectionner les 3 premiers canaux simultanément."

locate 10,1:input "Quels canaux voulez-vous activer (1,2,3 et/ou 4)";CANAUX\$

if CANAUX\$="q" or CANAUX\$="Q" then goto FIN

locate 10,1:print string\$(80, " ");

locate 12,1:print string\$(80, " ");

locate 13,1:print string\$(80, " ");

locate 5,28:print "Canaux sélectionnés : ";CANAUX\$

for i=1 to len(CANAUX\$)

A\$=mid\$(CANAUX\$,i,1)

if A\$="1" then A=8

if A\$="2" then B=4

if A\$="3" then C=1

if A\$="4" then D=2

next i

N=255-A-B-C-D:out 888,N

color 16,7,0

locate 10,13:print "Appuyez sur [Entrée/Enter] pour valider l'émission HF."

color 7,0,0

locate 15,10:print "Pour sélectionner de nouveaux canaux, appuyez sur [Echap/ESC]."

locate 16,10:print "Pour stopper l'émission HF, appuyez sur [A]."

locate 17,10:print "Pour reprendre l'émission HF, appuyez sur [C]."

locate 24,4:

print "Pour quitter le programme, appuyez sur [Q], [End/Fin] ou [Suppr/Delete].";

K\$="":k=0

DO

K\$=inkey\$

if k\$<>" " then k=asc(right\$(K\$,1))

if k=13 then

out 890,0

locate 10,1

print string\$(80, " ")

end if

if k=27 then exit loop

if k=65 or k=97 then out 890,1

if k=67 or k=99 then out 890,0

if k=79 or k=83 or k=81 or k=113 then goto FIN

k=0

loop

goto DEBUT:

FIN:

cls:print "Au revoir":end

composants de ces trois platines. Vous débuterez par les straps et poursuivrez avec les résistances, puis par des éléments successivement plus épais. Le tracé des pistes des cartes d'émission a été prévu pour recevoir différents modules HF, pour le cas où le booster UHF construit autour du transistor BFR96 ne serait pas utile. En effet, ce booster présente un fonctionnement optimal avec un gain d'environ 10dB et une portée limite d'environ 300m, s'il est associé avec un module émetteur AUREL TX433-SAW. Si ce booster n'est pas utilisé, vous pourrez utiliser le module TX433 de LC433 proposé par la société LEXTRONIC entre les pastilles désignées 7 et 12 ou un quelconque émetteur par un connecteur trois broches, connecté à l'emplacement [012] de TX<sub>1</sub>.

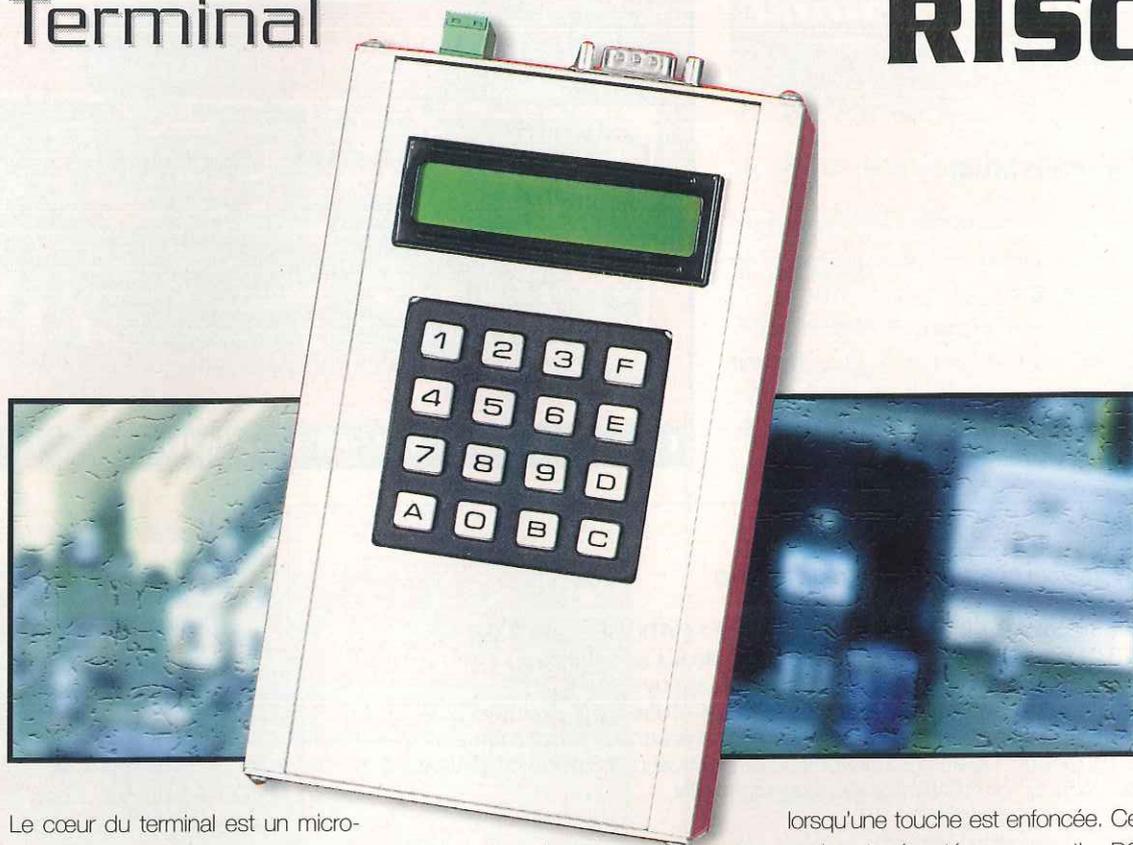
Pour le cas où la portée n'est pas critique et si vous souhaitez préserver le plus longtemps possible une alimentation par pile de la carte d'émission HF, le booster UHF ne doit pas être utilisé. Dans ce cas, les composants C<sub>2</sub>, R<sub>4</sub>, R<sub>3</sub>, L<sub>1</sub>, L<sub>2</sub>, C<sub>3</sub> et T<sub>1</sub> ne sont pas implantés sur la carte et l'antenne est alors connectée en sortie 11 du module émetteur TX<sub>1</sub>, à l'emplacement de l'une des deux bornes du condensateur C<sub>2</sub>. La **figure 12** présente le listing d'un petit programme écrit en TURBO BASIC, facilement transposable en un autre langage. Ce programme succinct, que vous pourrez facilement personnaliser, indique précisément les principales commandes de contrôle de la carte d'émission HF.

**12** Listing du programme

H.CADINOT

## Terminal

## RISC



**Ce mini-terminal peut être d'une grande utilité dans plusieurs domaines : présentation de messages explicites pour des systèmes qui en sont dépourvus, gestion d'accès à code, débogage de code par envoi sériel d'un contenu de registre, module de commande distant, etc. On pourra aussi s'en servir à des fins d'apprentissage de programmation pour microcontrôleurs AVR, utilisant les fonctions TIMER, UART, pilotage LCD, interruption externe.**

Le cœur du terminal est un microcontrôleur RISC de type AT90S2313, de l'écurie ATMEL. Il gère les trois périphériques essentiels du système : liaison RS232, afficheur LCD et clavier 16 touches. Le quartz  $Q_1$  fait fonctionner  $U_3$  à la fréquence de 7,3728 MHz, ce qui permet de générer facilement (et avec précision) les fréquences de communication du port RS232 : 2400, 4800, 9600 et 19200 bauds. Les instructions du microcontrôleur seront exécutées en

135 ns (une instruction est exécutée en un cycle machine).

Le clavier  $CL_1$ , de type matricé, est organisé en 4 colonnes de 4 touches chacune. Elles sont représentées par les chiffres 0 à 9 et les lettres A à F. Le clavier fonctionne de pair avec  $U_2$ , un circuit de décodage spécialisé. Il adresse continuellement les rangées et les colonnes de  $CL_1$  et fournit le code hexadécimal correspondant,

lorsqu'une touche est enfoncée. Ce code est présenté sur ses sorties D0 à D3.

La sortie DA (Data Available) indique au microcontrôleur l'activation d'une touche, et un code valide, par un niveau haut. Le front montant ainsi créé sera pris en compte de manière interruptive par  $U_3$ , ce qui évite une interrogation constante de  $U_2$ . L'entrée /OE (Output Enable) est forcée à l'état bas : les sorties du 74C922 sont donc toujours activées. Le condensateur  $C_5$  définit la fréquence de scrutation du clavier et le condensateur  $C_4$  la constante de temps d'anti-rebond. L'interface sérielle est bâtie autour de  $U_5$ , un convertisseur TTL/RS232 classique (MAX232).

Les condensateurs  $C_8$  à  $C_{11}$  font fonctionner les pompes de charge internes pour disposer des niveaux de tensions requis par la norme RS232, à partir de l'alimentation unique +5Vdc.

On n'utilise que les lignes TxD et RxD, ce qui suffit à bon nombre de systèmes pour communiquer sans problème.

La connexion RS232 se fait au travers du connecteur  $JP_2$ , une DB9 mâle soudée. Le brochage est standard :

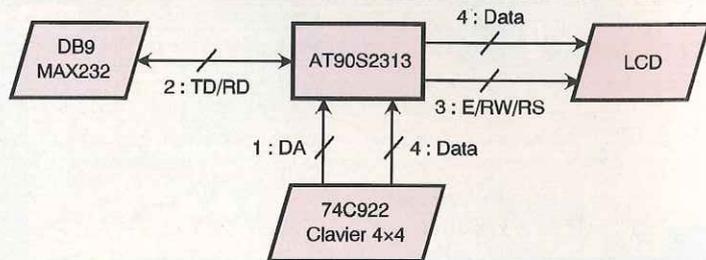
$C_5$	Fréquence de scrutation
10 nF	6 kHz
$\approx 60$ nF	1 kHz
100 nF	600 Hz
$\approx 600$ nF	100 Hz
1 $\mu$ F	60 Hz

$C_4$	Période d'anti-rebond
100 nF	1 ms
470 nF	4,7 ms
1 $\mu$ F	10 ms
4,7 $\mu$ F	47 ms
10 $\mu$ F	100 ms
47 $\mu$ F	470 ms
100 $\mu$ F	1 s



Caractéristiques du 74C922

Signal	Broche	Fonction
TxD	3	Transmission de donnée
RxD	2	Réception de donnée
GND	5	Masse



## Synoptique général

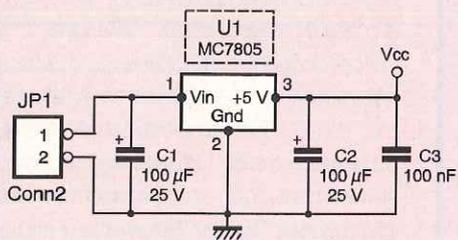
chaque utilisation et à chaque mise hors tension).

L'afficheur LCD est un modèle courant à deux lignes de 16 caractères, rétro-éclairé. La commande de cette fonction incombe au cavalier I<sub>1</sub>, situé sur le circuit imprimé : l'éclairage est actif en position fermée. Il convient donc de le placer à sa convenance, en fonction de l'éclairage ambiant ou de la consommation électrique (environ 400mA pour le rétro-éclairage).

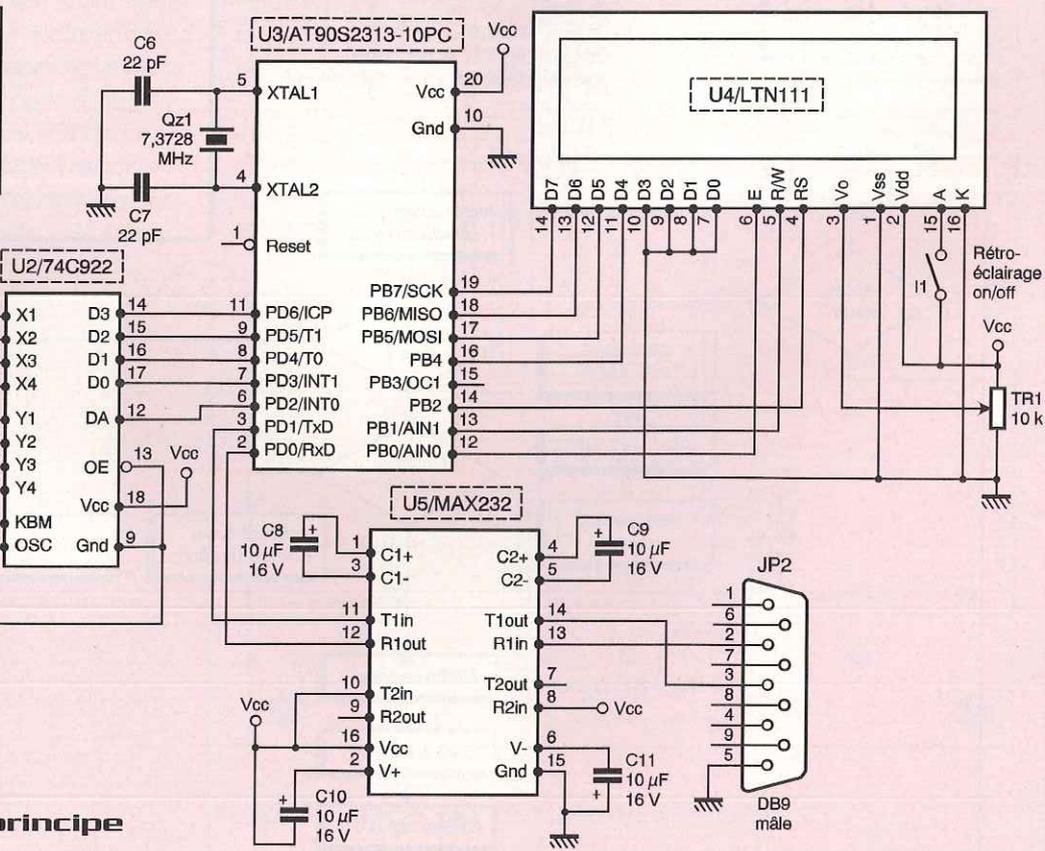
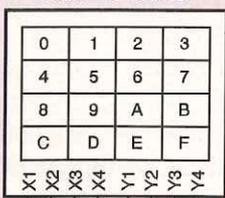
Le contrôleur de l'afficheur est piloté en mode 4 bits, ce qui économise des entrées/sorties, mais qui oblige un transfert des données de commande ou les caractères en deux phases (quartet haut, puis quartet bas).

A la mise sous tension, il convient d'envoyer à l'afficheur des codes bien définis pour le placer dans une configuration utilisable : mode 4 bits, activation et effacement de l'écran, curseur invisible, position 1ère colonne - 1ère rangée. Par la suite, les caractères reçus seront affichés les uns à la suite des autres, avec un saut de ligne automatique après la position 16ème colonne - 1ère ligne. L'opération se répète

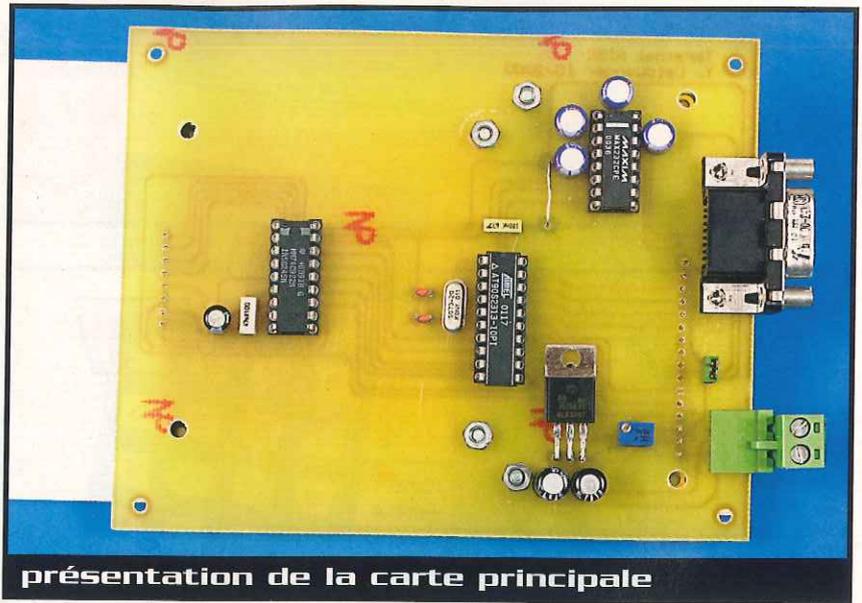
Le protocole de dialogue sera le suivant : donnée 8 bits, pas de parité, 1 bit de stop. La vitesse de transmission sera choisie parmi quatre disponibles : 2400, 4800, 9600, 19200 bauds. Cette configuration est accessible au démarrage du circuit et mémorisée en EEPROM (ce qui évite une reconfiguration à



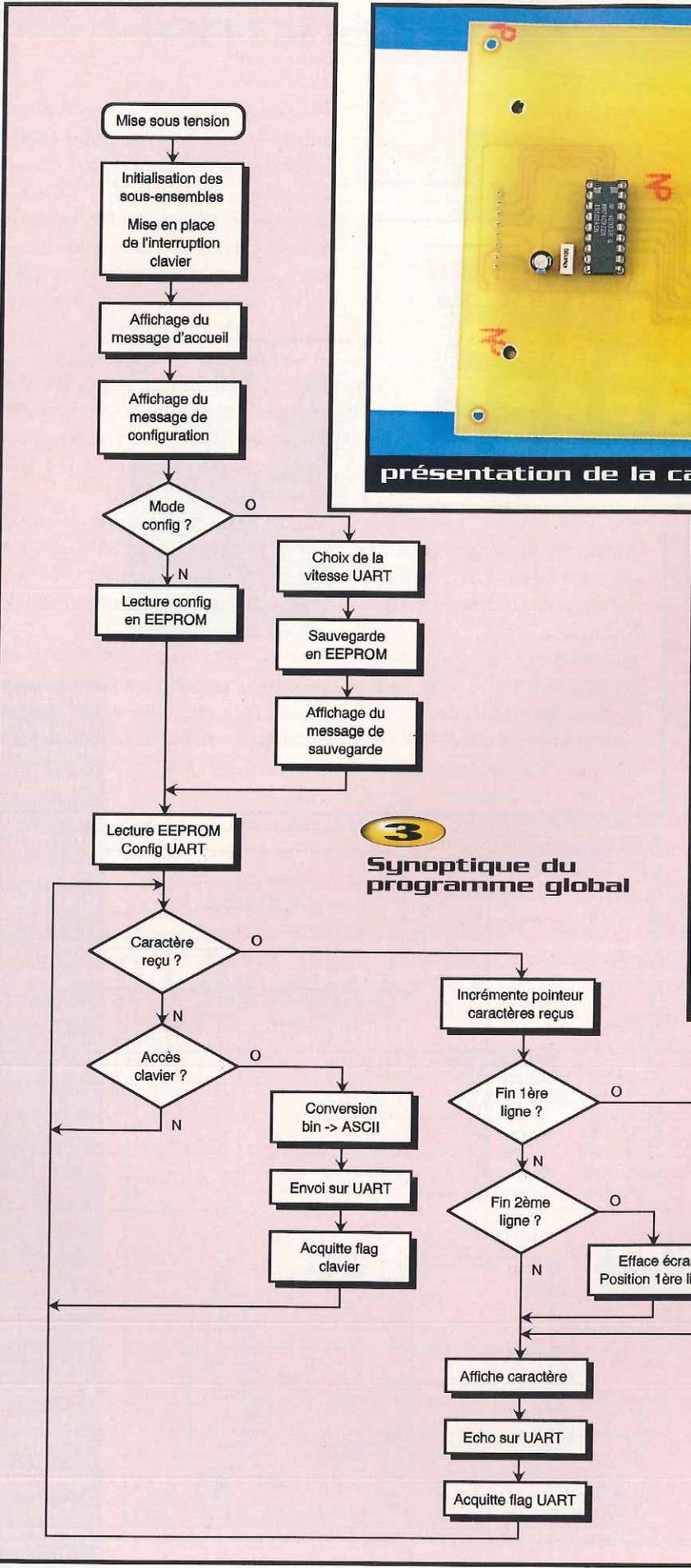
CL1  
Clavier 16 touches



## Schéma de principe



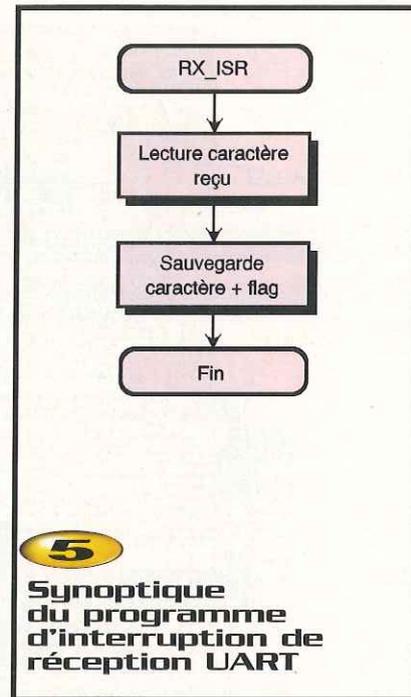
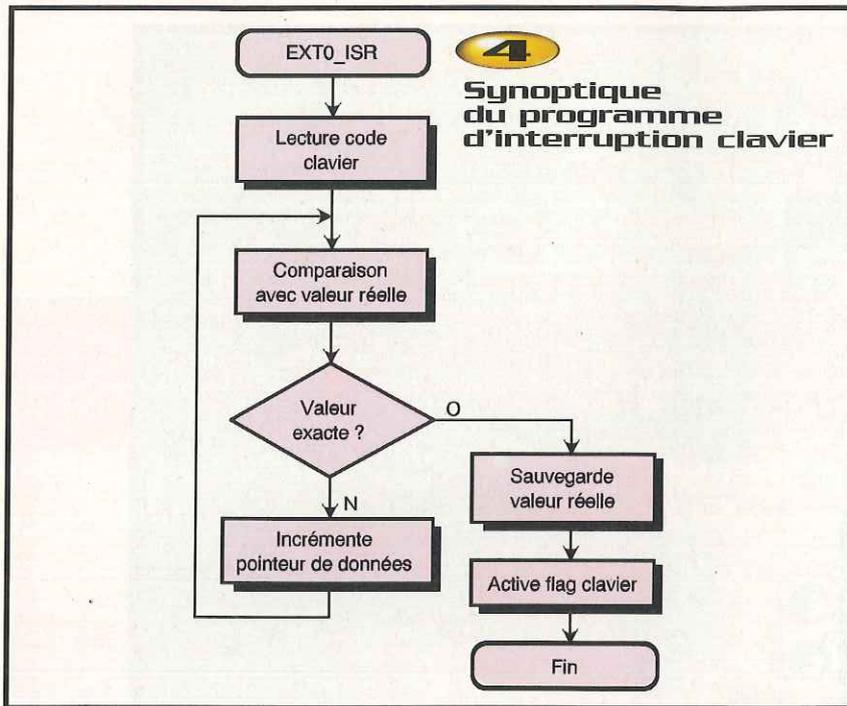
présentation de la carte principale



**3**  
Synoptique du programme global

tera jusqu'à la position 16ème colonne - 2ème ligne. Le caractère suivant provoquera l'effacement de l'écran et sera affiché en position 1ère colonne - 1ère ligne. Par conséquent, on peut envoyer des données à afficher caractère par caractère, par bloc de 16 (ligne complète) ou par bloc de 32 (écran complet). Les routines implantées se chargent donc automatiquement des commandes Carriage Return, Line Feed et effacement. Néanmoins, tous les caractères reçus sont présentés à l'afficheur, sans distinction. Les caractères "non imprimables" provoqueront sans doute des affichages étranges sur l'écran LCD.

Le programme du microcontrôleur va fonctionner principalement sur trois tâches, dont



les synoptiques sont données ci-dessous :

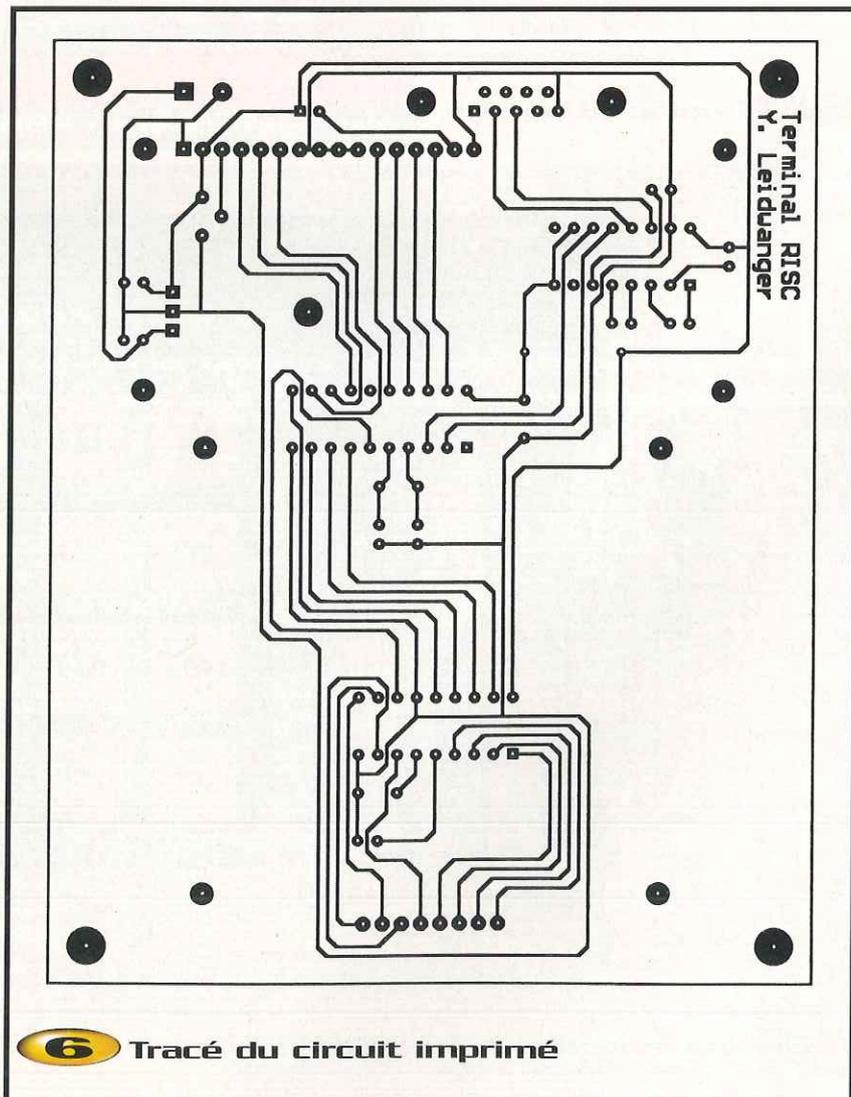
- Un module en temps réel de récupération de donnée reçue par l'UART,
- Un module en temps réel de récupération de donnée clavier,
- Un module en temps partagé qui orchestre la réception UART, la réception clavier, l'affichage et l'envoi.

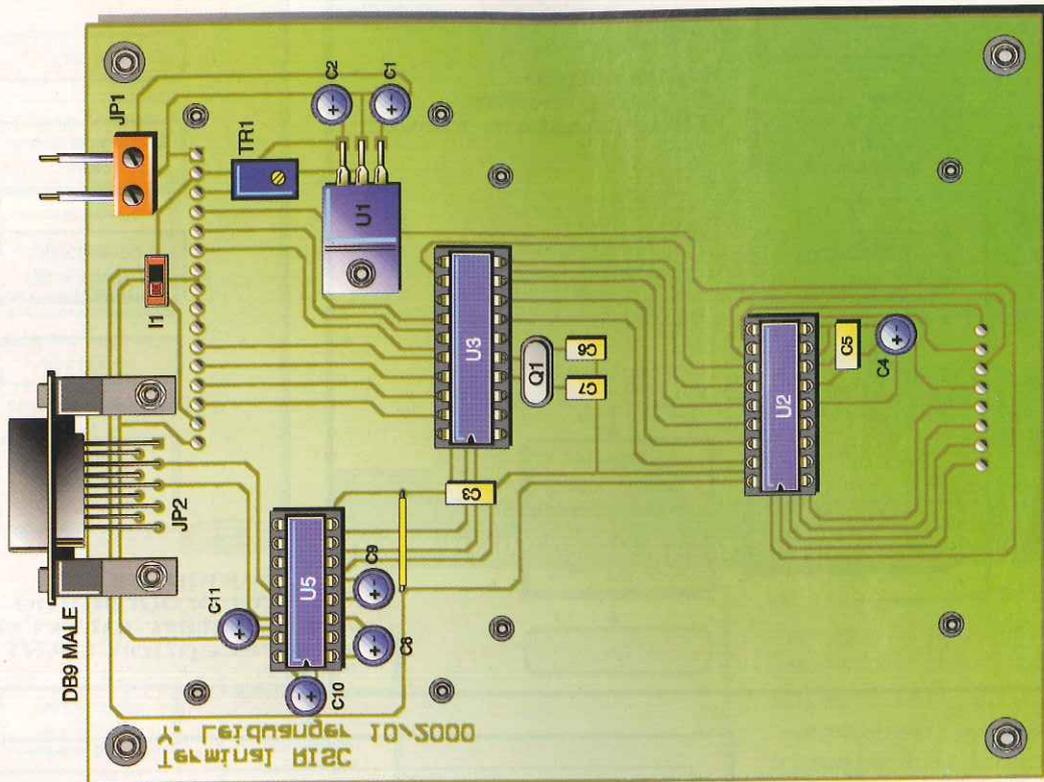
Le microcontrôleur sera programmé avec le fichier TERM.HEX (code FLASH) et le fichier TERM.EEP (code EEPROM), à l'aide du programmeur adéquat. Les fichiers sources sont TERM.ASM et 2313DEF.INC. La fonction de programmation ISP ne peut pas se faire ici, puisque toutes les I/O sont utilisées.

## Réalisation

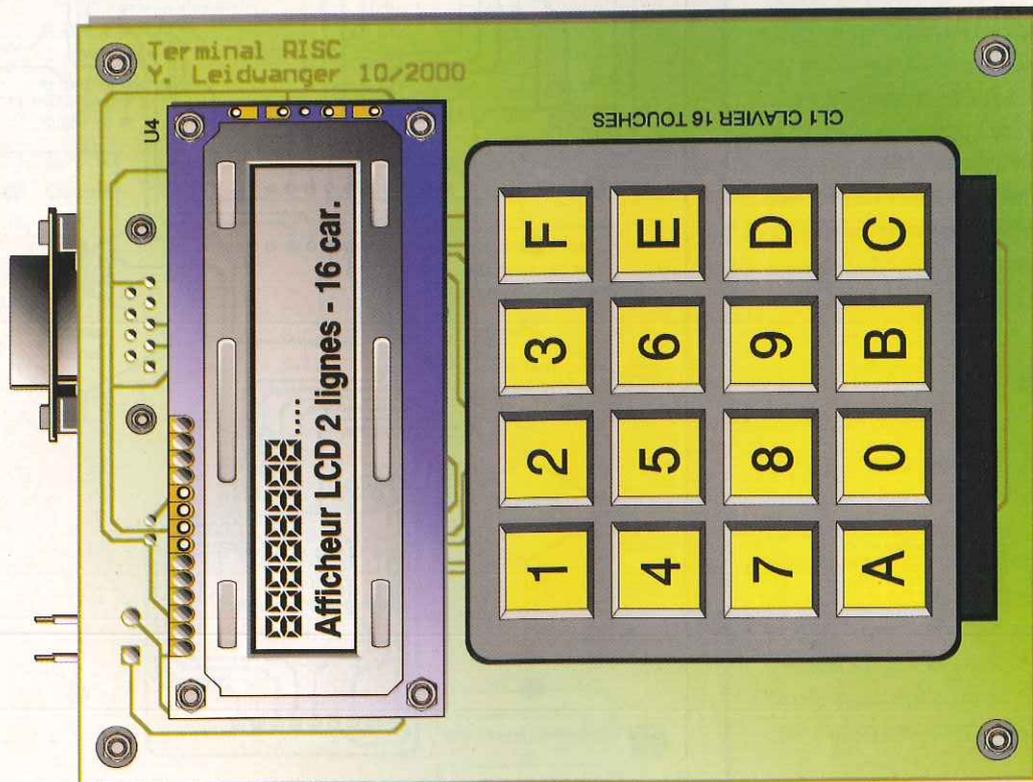
La réalisation de la carte électronique se fera de manière classique, avec de l'époxy simple face. On placera en premier le strap, puis chaque composant par ordre de taille croissante. Attention, l'afficheur LCD et le clavier seront soudés côté "cuivre" à l'aide de barrettes sécables droites et d'entretoises. Les trous de passage de vis seront de 3mm, pour des entretoises hexagonales inox M3.

Le connecteur DB9 est un modèle à harpons, pour une meilleure rigidité mécanique lors de l'insertion ou de l'extraction de la fiche femelle (un modèle à vis fera aussi





**7** Implantation des éléments



**8** Implantation côté cuivre

parfaitement l'affaire !). Les seuls réglages à effectuer sont le cavalier de commande du rétro-éclairage, ainsi que le trimmer de contraste.

Le circuit imprimé terminé sera intégré dans un coffret en aluminium, dont les découpes et gravures seront effectuées par une société spécialisée. Il conviendra à chacun

de coter les découpes, en fonction du coffret choisi.

En ce qui concerne l'alimentation, elle proviendra d'une source continue, supérieure ou égale à 6Vdc. Un bloc secteur 220V/7,5Vdc/800mA conviendra parfaitement. Le régulateur U<sub>1</sub> se chargera de four-

nir aux composants de la carte une tension de +5Vdc, décapée par les condensateurs C<sub>2</sub> et C<sub>3</sub>.

Y. LEIDWANGER

## Nomenclature

- CL<sub>1</sub> : clavier matricé 16 touches alpha-numériques
- Q<sub>1</sub> : quartz 7.3728MHZ
- JP<sub>1</sub> : bornier industriel coudé 2 points mâle
- JP<sub>2</sub> : DB9 mâle coudée
- I<sub>1</sub> : cavalier 2 points
- TR<sub>1</sub> : trimmer 10 Ω vertical
- U<sub>1</sub> : MC7805
- U<sub>2</sub> : 74C922 + support DIL18
- U<sub>3</sub> : AT90S2313-10PC + support DIL20
- U<sub>4</sub> : afficheur LCD 2 lignes de 16 caractères
- U<sub>5</sub> : MAX232 + support DIL16
- C<sub>1</sub>, C<sub>2</sub> : 100 µF/25V rad.
- C<sub>3</sub> : 100 nF
- C<sub>4</sub>, C<sub>5</sub> : 22 pF



# Programmateurs - Emulateurs - Simulateurs

**ELNEC**

Déjà présente dans plus de 20 pays différents, la Société ELNEC est à l'heure actuelle considérée comme un des principaux leaders mondiaux dans le domaine de la conception de programmeurs de composants professionnels. Leur grande fiabilité, leur étonnante simplicité d'utilisation et leurs prix extrêmement compétitifs sont à l'origine de leur succès auprès des unités de production, des bureaux d'études, des centres de S.A.V, des écoles, des universités, etc...

**JetProg** Ce modèle supporte **4455 composants**. Ses adaptateurs optionnels lui permettant de programmer jusqu'à 8 FLASHs ou EPROMs ou EEPROMs en même temps. **Garantie 3 ans**

**LabProg+**

- Support ZIF 48 broches
- 4526 composants gérés EPROM, Flash, EEPROM, GAL, PLD, microcontrôleurs ST6xx, SCENIX™ SXxxx, ZILOG™, 68HC11, COP8xxx, T1™MSP430, série MC51, MC48, 196 Intel™, AT90Sxxx, AT tiny, NEC™ (µP78Pxxx), famille "PIC", etc...

**Garantie 3 ans** ..... **5023 F TTC**

**Preprom-02alv**

- Support ZIF 32 broches
- 2544 composants gérés EPROM (27xxx / 27Cxxx jusqu'à 8 Mb), EEPROM (28xxx / 28Cxxx), Flash (28Fxxx, 29Cxxx 29Cxxx, 29Fxxx 29LVxxx...), test les RAMs non volatiles, peut (avec adaptateurs optionnels) supporter les PLD, GAL, microcontrôleurs PIC, AVR, MC551 - **Garantie 3 ans 1718 F TTC**

**PikProg**

- Support ZIF 40 broches
- 1021 composants gérés Série 12xxx, 14xxx, 16xxx, 18 xxx ainsi que les EEPROM série 17C xxx, 24xxx, 25Cxxx, 59Cxxx, 85xxx, 93xxx, 95xxx ..... **1234 F TTC**

**51&AVRprog**

- Support ZIF 40 broches
- 957 composants gérés Série mP51 87Cxxx, 87LVxx, 89C xxx, 89Sxxx, 89LVxxx (Atmel™, Philips™, Intel™...) - Série AVR AT90Sxxx (mode parallèle) ainsi que les EEPROM série 24xxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx ..... **1234 F TTC**

**T51prog**

- Support ZIF 40 broches
- 1040 composants gérés Similaire à ci-dessus, mais pouvant aussi être relié sur un port parallèle "PC" IEEE1284 pour un taux de transfert ultra rapide. Permet également la programmation série (ISP) ou parallèle des µP AVR (AT90Sxxx, AT mega, ATtiny) **1511 F TTC**

**SEEPROM**

- Support ZIF 24 broches
- 930 composants gérés EEPROM série 12C™ (24Cxx), Microwire™ (93Cxx), SPI™ (25Cxx) - supporte les composants basse tension (3,3 V) ..... **714 F TTC**

**EREPROM-01A**

Effaceur d'EPROM (jusqu'à 5 composants à la fois) - Livré avec bloc secteur - Tempo. gérée par µP..... **679 F TTC**

**COMFILE TECHNOLOGY**

Cette gamme d'émulateurs temps réel dotée d'un des meilleurs rapport qualité/prix du marché est spécialement conçue pour les microcontrôleurs "PIC" Microchip™. Ils disposent d'un logiciel commun (sous environnement Windows™) extrêmement puissant et convivial qui vous permettra d'utiliser un nombre infini de points d'arrêt, d'effectuer une émulation des différents processeurs (mode "XT" - jusqu'à 20 MHz ou en mode pas-à-pas avec visualisation / modification des registres).

**5H EMUL**

Modèle livré sous forme de platine nue, capable d'émuler les "PIC" de la série PIC16C5x et les PIC12C50x.

Livré avec sondes et logiciel ..... **1395 F TTC**

**PRO STATION-H3**

Modèle livré sous forme d'un boîtier avec modules d'extensions enfilables permettant d'émuler différentes familles de "PIC": série PIC 16C7x, PIC16C6x, PIC 16C71x, PIC16F8x, PIC 16F87x...

A titre indicatif, version pouvant émuler les microcontrôleurs PIC16C62, 63, 64, 65, 72, 73, 74 livrée avec sondes et logiciel ..... **3819 F TTC**

**MR.PIC**

Ce modèle intègre dans un même boîtier un émulateur et un programmeur de PIC pouvant gérer différentes familles: suivant l'extension utilisée: PIC 16C7x, PIC16C6x, PIC 16C71x, PIC16F8x, PIC16F87x... A titre indicatif: version pouvant émuler/programmer les micro-contrôleurs PIC16C62, 63, 64, 65, 72, 73, 74, PIC16C52, 54, 55, 56, 57, 58, PIC 12C508, 509 livrée avec ses sondes + son logiciel complet ..... **6323 F TTC**

Consultez-nous pour les extensions

**MR.68618**

Ce boîtier permet de programmer jusqu'à 10 microcontrôleurs PIC en même temps parmi les séries:

PIC12C 508, 508A, 509, 509A, CE518, CE519, 52, 54, 54A, 54B, 54C, 55, 55A, 56, 56A, 57, 57C, 58A, 58B, 505, PIC4C000, 554, 554A, 62, 62A, 62B, 63, 63A, 64, 64A, 65, 65A, 65B, 66, 67, 620, 620A, 621, 621A, 622, 622A, 710, 711, 712, 72, 72A, 73, 73A, 73B, 74, 74A, 74B, 76, 77, 773, 774, F83, C84, F84, F84A, 923, 924...

Le programmeur 10 supports ..... **1409 F TTC**

Même modèle mais que pour 5 microcontrôleurs. Le programmeur 5 supports ..... **1167 F TTC**

**Module Voice-extrême 364**

Ce module hybride intègre un microcontrôleur spécialisé entièrement programmable en langage "C". En plus de bénéficier d'entrées d'entrées/sorties, de RAM, de timers, de port série, il dispose d'instructions pour reconnaissance vocale mono-locuteur ou de mots de passe, de génération DTMF, d'enregistrement et de reproduction de la voix, de restitution à partir de fichiers ".WAV"... Il se programme facilement à l'aide d'un pack de développement comprenant un logiciel et un cordon série à raccorder au port série du PC. Une fois le programme "téléchargé" dans sa mémoire Flash, il peut être déconnecté de l'ordinateur afin de devenir autonome.

Le module seul ..... **430 F TTC**

Le pack de développement complet comprenant 1 module "Voice-extrême 364" + une platine support avec zone de développement + un câble de téléchargement + une suite logiciel (comprenant le langage "C" + un linker + un logiciel de traitement des fichiers ".wav") ..... **1177 F TTC**

Tous les programmeurs se connectent au port imprimante de votre compatible PC sans carte d'interface supplémentaire - La mise à jour des logiciels est "gratuite" et illimitée sur le site du Fabricant: [www.elnec.com](http://www.elnec.com) (nouvelles versions mises à disposition toutes les 2 semaines env.) Tous les logiciels fonctionnent sous environnement DOS™, Windows™ 95/98/Me/NT/2000.

**LEXTRONIC**

36/40 Rue du gal de gaulle  
94510 La Queue en Brie

CD-ROM 3,91 € port contenu

Tél: 01.45.76.83.88  
Fax: 01.45.76.81.41

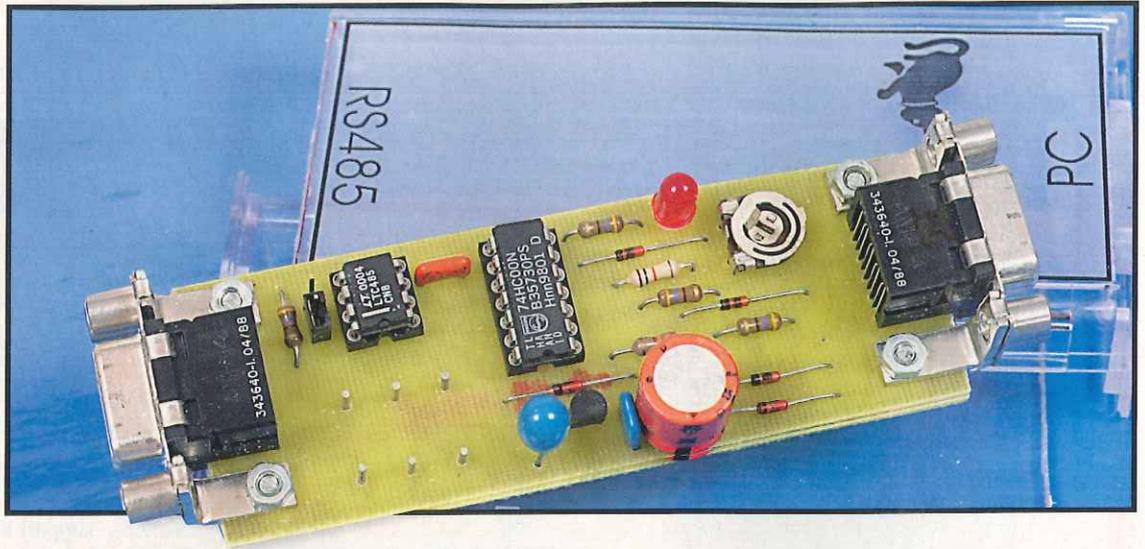
Site internet [www.lextronic.fr](http://www.lextronic.fr)

Importateur pour la France des produits:

- Comfile technology
- Elitech
- Elmec
- Kern-electronic
- Microrobot
- Radiometric
- Sansory
- Telecontrolli

Pour toute commande  
frais de port: 6,71 € (44 F)

# Chat en RS485



**Avec seulement deux liaisons série RS232 disponibles, le PC ne permet pas le dialogue avec plusieurs périphériques simultanément, PC compris. Naturellement, il existe des réseaux modernes (ethernet, USB), mais leur mise en œuvre est complexe et nécessite un PC récent, l'alternative : le RS485.**

L'interface RS485 décrite a été conçue pour fonctionner avec tous les PC, les vieux, de première génération ou les portables. Les applications sont multiples : l'instrumentation (un peu comme le bus GPIB), la causerie entre les PC de la maison (chat), la domotique. Le standard RS485 est utilisé aux USA en domotique, le X10 sur secteur est limité, les liaisons sans fil sont chères, posent des problèmes de compatibilités et sont écoutables...

Le standard RS485 permet de connecter plusieurs périphériques sur un même bus. Le nombre de périphériques pouvant être raccordés dépend de la topologie, de la vitesse de transmission etc. Dans les applications classiques il est voisin de trente.

La transmission est half duplex, une station émet pendant que les autres écoutent. La transmission s'effectue à 9600 bauds, c'est la vitesse maximale disponible avec le bios «classique» d'un PC. Un port de communication (Com1 ou Com2) est utilisé pour assurer la liaison entre le PC et le réseau RS485. C'est une vitesse rapide en comparaison avec la domotique sur le secteur qui utilise un débit de 1200 bauds. Sur un bus RS485, il faut placer les récepteurs

directement sur le bus et éviter les bretelles de dérivation (stub en anglais), à 9600 bauds cette règle est moins contraignante.

## Le montage

La difficulté principale de ce montage réside dans la gestion de l'alternat entre l'émission et la réception tout en conservant son caractère universel. Le bios du PC ne connaît pas l'alternat, il faut ouvrir le driver de transmission RS485 dès qu'un caractère est émis par le PC, puis, à la fin de la transmission de celui-ci, repasser en réception. Un monostable assure ce fonctionnement.

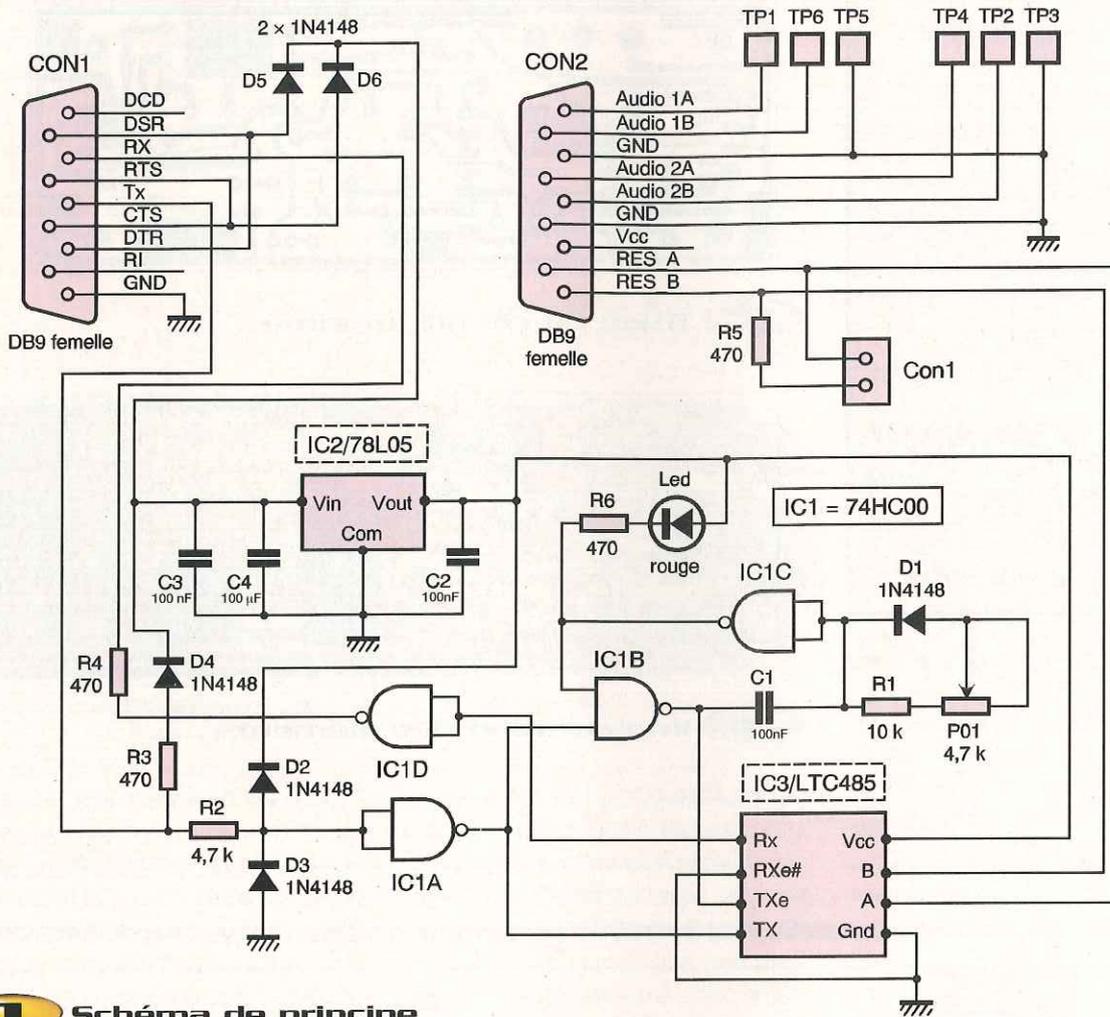
La **figure 1** présente le plan électrique de l'interface. Le montage est alimenté directement par le PC, les lignes de service et la ligne TX fournissent le courant nécessaire. Avec un PC de bureau, la ligne Tx n'est pas utilisée,  $R_3$  ne sera pas montée. Les portables sont souvent moins généreux en courant, TX pourra alors être utilisé, sinon on peut aussi utiliser une alimentation extérieure, sans  $R_3$  : on relie le + de l'alimentation sur l'anode de  $D_4$ .

Pour la gestion de la RS232, côté PC, on utilise un seul circuit ( $IC_1$ ), un 74HC00. Deux portes assurent la transmission et la réception RS232.

Le niveau 0/5V, bien qu'inférieur au niveau classique -12/+12V, fonctionne parfaitement. Deux autres portes sont utilisées pour le monostable. Enfin le circuit  $IC_3$  assure l'interface RS485. Ce circuit est toujours maintenu en réception par la mise à la masse de l'entrée Rx.

Ainsi tout caractère émis est reçu : c'est l'écho, c'est aussi un moyen de contrôler que le bus RS485 fonctionne correctement. En déconnectant l'interface du réseau on vérifie la carte, connectée on vérifie le bus.

A l'état repos, le niveau de Tx est à -12V, la sortie de  $IC_{1A}$  est à 1, comme  $IC_{1C}$  puisque son entrée est à la masse par  $R_1$  et  $P_{01}$ . Le circuit  $IC_{1B}$ , recevant deux 1 donne 0, le driver RS485 est inactif. Dès que Tx passe à +12V correspondant au start bit, le signal arrive sur l'entrée Tx du driver RS485 ( $IC_2$ ), mais le driver n'est pas ouvert, il faut que Tx passe à 1. Le signal Tx arrivant aussi sur  $IC_{1B}$  le fait basculer, le niveau passe à 1 et ouvre le driver, le temps de propagation dans la porte correspond au retard d'ouverture de la porte, une trentaine de ns, un temps négligeable. Le passage à 1 de  $IC_{1B}$  est transmis par  $C_1$  à l'entrée de  $IC_{1C}$  qui bascule à son tour et bloque définitivement  $IC_{1B}$  à 1. Les changements de Tx n'influenceront plus sur l'ouverture de la porte.



## 1 Schéma de principe

En fait cela n'est pas définitif, puisque la capacité  $C_1$  se charge lentement par  $R_1$  et  $P_{01}$ , avec une constante RC de 1 ms, soit le temps de transmission d'un caractère à 9600 bauds.

A partir du bit de stop, Tx repasse dans son état de repos -12V,  $C_1$  chargé fait repasser  $IC_{1C}$  à 1, puisque son entrée est à la masse, la porte du driver RS485 se referme.

Reste un détail qui a son importance : la

diode  $D_1$ . A la fermeture  $C_1$  est chargé, comme  $IC_{1B}$  passe à 0, l'entrée de  $IC_1$  peut devenir négative, la diode vide  $C_1$ , cela doit se faire rapidement puisqu'il faut que  $C_1$  soit vide pour que le prochain temps d'ouverture soit toujours le même.  $P_{01}$  permet de régler le temps d'ouverture de la porte.

Les valeurs fournies de RC le sont pour un fonctionnement à 9600 bauds.

Au niveau des émetteurs/récepteurs RS485, il existe deux types de circuits compatibles broche à broche : la série 75176 et la série LTC485, ces produits existent chez plusieurs constructeurs, la racine est toujours 75176 et 485.

Le 75176 est un circuit de première génération, il consomme 35mA en transmission à vide et 26mA en réception.

Dans les mêmes conditions, le LTC485 consomme seulement 0,5mA et 0,3mA respectivement. Tous ces circuits restent en haute impédance quand ils ne sont pas alimentés, ils ne perturbent pas le réseau.



Patte	Description
5	RS485
9	RS485
4	NC/alim réseau
8	GND
3	Analog 1 A
7	Analog 1 B
2	GND
6	Analog 2 A
1	Analog 2 B

PC	Nano-Reseau	Réseau
	9600	
	The Door	

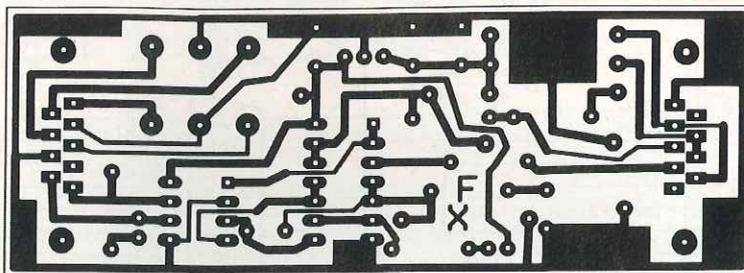

**Signaux sur le bus**

Comme le montage est auto-alimenté par la liaison RS232 du PC, on utilisera le LTC485 afin de conserver l'énergie pour la ligne RS485.

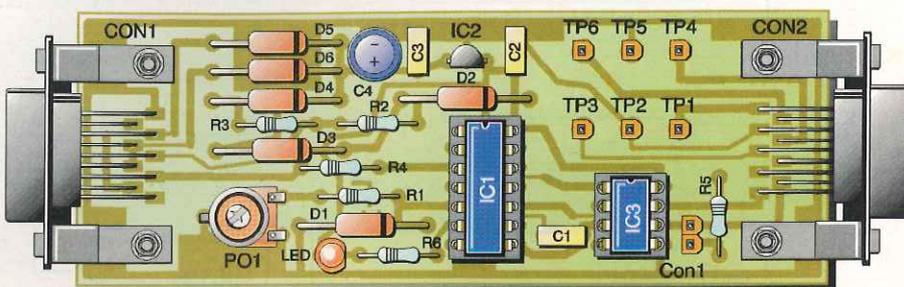
Pour éviter les réflexions et réduire l'influence des parasites sur le réseau, on peut mettre une résistance d'adaptation par le cavalier cav<sub>1</sub>, la valeur de R<sub>5</sub> dépend de l'impédance caractéristique du câble. En cas de parasite, on peut passer de l'infinie à 470 Ω pour faire un essai. Souvent, sans adaptation, le bus fonctionne correctement. Le bus utilise un connecteur DB9. Trois points (masse, A et B) servent pour la RS485. Les autres points sont réservés pour passer d'autres signaux analogiques, par exemple comme l'audio ou la vidéo. Un point est réservé pour une alimentation. Le **tableau 1** donne le descriptif du bus. Le bus peut être monté rapidement en utilisant des connecteurs DB9 à sertir. Seulement 4 fils sont nécessaires pour la RS485, avec 6 fils on dispose d'une voie double analogique. Dans cette configuration, on peut utiliser le câble 6 fils d'interphone, disponible dans les grandes surfaces, relativement peu coûteux, il faudra souder les prises.

### Montage (figures 2 et 3)

La **figure 3** montre l'implantation des composants. Le circuit est un simple face, aucun strap n'est nécessaire. Le montage n'a pas de difficultés particulières, toutefois voici quelques conseils.



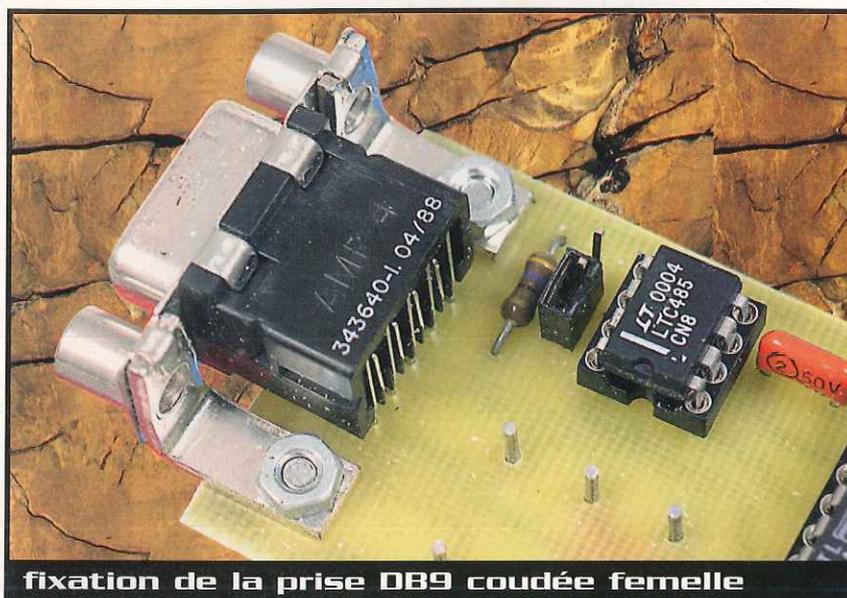
### 2 Tracé du circuit imprimé



### 3 Implantation des éléments

En ce qui concerne le perçage de l'époxy, il faut utiliser un foret au carbure, les forets normaux ne percent plus après une vingtaine de trous, on perfore et c'est un désastre. Un foret au carbure ne s'use pas, il se casse auparavant. Cela fait toujours mal de perdre la quarantaine de francs d'investissement. Même après un an d'utilisation, le trou est toujours impeccable. Le support de perceuse n'est pas absolument nécessaire (l'auteur n'en utilise pas), de toute façon le support bas de gamme est à proscrire, on casse à coup sûr...

Pour ne pas casser, il faut tenir correctement la mini-perceuse, jamais «en l'air», un faux mouvement et c'est la casse, la main posée sur la table fait office de support. Avec un appui sur la table, la main devient une véritable petite perceuse à colonne. Les gestes brusques sont à proscrire. Il ne faut pas trop de couple, la perceuse s'alimente en 12V avec un 7812, le régulateur limite le courant. Enfin l'interrupteur est à porter de l'autre main, on n'utilise pas celui sur la perceuse. Il peut arriver que le foret se bloque, ça n'a



fixation de la prise DB9 coudée femelle

pas encore cassé, il faut couper le courant, sans bouger la main «support», puis délicatement sortir le foret en tournant à l'envers avec l'autre main (la gauche qui vient de couper le courant).

Enfin la plaque d'époxy est surélevée par deux morceaux d'époxy, la sortie du foret s'effectue à l'air, on retient sa sortie. L'autre main (qui doit pouvoir couper le courant, la gauche !) tient par pression la plaque d'époxy. Très souvent, on enlève les copeaux en soufflant légèrement ou avec un doigt. Après une quinzaine de trous, on fait une pause pour le foret et le poignet. Le foret est rangé dans sa boîte après utilisation. Ainsi on peut conserver un foret plus d'un an.

Le boîtier transparent se perce au foret à bois, directement au diamètre final, à la perceuse à main. Le trou est rond, sans bavure et il n'y a pas eu d'échauffement du plastique. Le foret à bois a une pointe au centre et attaque sur le bord, en observant l'attaque du foret, on maintient la perceuse perpendiculaire. C'est une alternative à la perceuse à colonne.

Dans tous les cas, le boîtier est maintenu dans un étau avec un rembourrage pour ne

pas rayer le boîtier. Il faut aussi protéger le boîtier en collant du papier et du scotch pendant l'ajustage des découpes à la lime.

### Test du montage

IC<sub>1</sub> et IC<sub>3</sub> ne sont pas montés. Il faut relier le montage au PC. On s'aperçoit de l'absence du 5V, pas de panique c'est peut être normal. En effet, à l'état repos les sorties de gestion de lignes RS232 sont à -12V. Il faut envoyer un caractère pour les faire basculer à +12V.

Avec Windows, on peut utiliser Hyper Terminal (dans accessoire). Sous Dos, un des très nombreux logiciels de terminal.

Une fois vérifié l'alimentation, on peut débrancher l'interface et mettre les circuits dans leur support. En recommençant l'opération, on peut tester l'écho. Il ne reste plus qu'à relier plusieurs PC par réseau, quel que soit leur âge (des PC) pour pouvoir dialoguer en «chat».

Pour terminer, il faut faire attention au problème de masse. Les masses des appareils (par exemple les PC) ne sont pas aux mêmes potentiels, il n'est pas rare d'avoir une secousse électrique en branchant une

prise sur un terminal par exemple. Bien que protégés contre les décharges électriques, certains circuits ne résistent pas. L'idéal serait de raccourcir les points des connecteurs, sauf la masse afin que celle-ci soit reliée en premier (comme sur l'USB).

Sinon, dans l'ordre de préférence, il vaut mieux relier en dernier, le PC au secteur, la RS485, la RS232. La liaison RS232 est la plus fragile.

X. FENARD

### Nomenclature

- C<sub>1</sub> à C<sub>3</sub> : 100 nF
- C<sub>4</sub> : 100 µF/25V radial
- IC<sub>1</sub> : 74HC00
- IC<sub>2</sub> : 78L05
- IC<sub>3</sub> : LTC485
- D<sub>1</sub> à D<sub>6</sub> : 1N4148
- TP<sub>1</sub> à TP<sub>6</sub> : picots à souder
- R<sub>1</sub> : 10 kΩ
- R<sub>2</sub> : 4,7 kΩ
- R<sub>3</sub> à R<sub>6</sub> : 470 Ω
- LED rouge
- P<sub>01</sub> : 4,7 kΩ
- CON<sub>1</sub>, CON<sub>2</sub> : DB9 coudés femelle: souder

**petitfute.com**  
Le guide du Net Shopping  
Les 1001 meilleurs sites marchands

**les bonnes adresses du bout de la rue au bout du monde...**

- Pratique Autos, 2 roues
- Informatique
- Cadeaux
- Enfants
- Gourmandises
- Livres, Presse
- Loisirs & Sports
- Maison
- Mode, beauté
- Musique
- Services
- Financiers
- Tourisme
- Achats Futé
- En savoir plus

**Achetez nos guides sur le net!**  
Paiement sécurisé  
Livraison gratuite en 72h

**WWW.e-petitfute.com**

**BON DE COMMANDE**  
Net Shopping

Nom.....prénom.....Age.....Profession.....  
Je commande.....exemplaire(s) du guide  
(prix unitaire 79F) Je joins en règlement un chèque de.....

à l'ordre des : **Nouvelles Editions de l'Université**  
**18 rue des Volontaires 75015 PARIS**

# Programmateur de 24C16

portable



De nombreuses applications nécessitent la remise à jour d'une mémoire de type 24C16 régulièrement, cet appareil vous permettra d'éviter de déployer les grands moyens (PC, programmeur, prise de courant).

## Principe général

Le programmeur est constitué d'un boîtier avec un support 8 broches pour la 24C16 et d'un bouton de mise en route.

Le téléchargement du programme, à charger dans la 24C16, est effectué via un connecteur externe (jack stéréo) depuis un ordinateur PC. Le logiciel d'exploitation du programmeur est écrit en DELPHI 4 (cf. programmation) et vous permet de charger, dans le programmeur, des fichiers au format HEX INTEL.

La mise en route du programmeur est réalisée par un appui bref sur le bouton, la LED clignotante verte signale la mise sous tension effective du programmeur. Si le bouton n'est plus actionné, la mise hors tension du programmeur est automatique au bout de 10 secondes.

Une fois le programmeur en service, le programme de la 24C16 peut être téléchargé dans le programmeur à l'aide du logiciel fourni.

Pour réaliser la programmation de la mémoire, une fois le programmeur arrêté, placer la 24C16 sur le support,

puis actionner la touche de mise sous tension du programmeur. La LED rouge s'allume indiquant une programmation en cours, lorsque la programmation est terminée, si le cycle s'est passé correctement, la LED s'allume verte clignotante dans le cas contraire la LED est rouge clignotante.

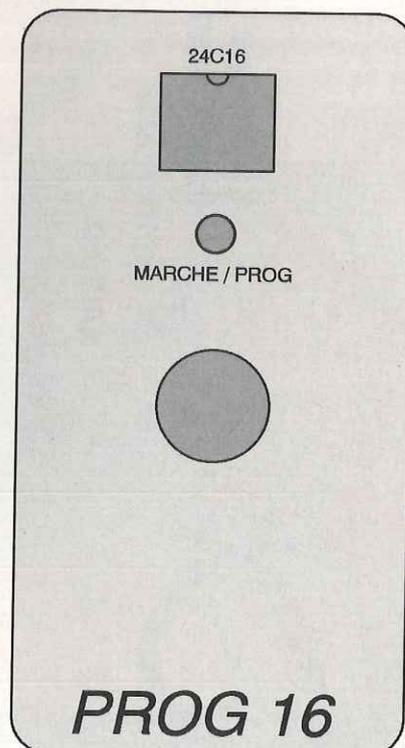
## L'électronique

Le programmeur est construit autour d'un microcontrôleur PIC 16F873, on trouve autour de ce composant le quartz de 11,0592 MHz et les 2 condensateurs de l'oscillateur. Le circuit de reset est constitué d'une résistance de 10 k $\Omega$  et d'un condensateur de 10  $\mu$ F. Côté liaison avec le PC, un convertisseur TTL/RS232 est utilisé (MAX232) et permet de rendre les niveaux compatibles avec ceux de la liaison série du PC.

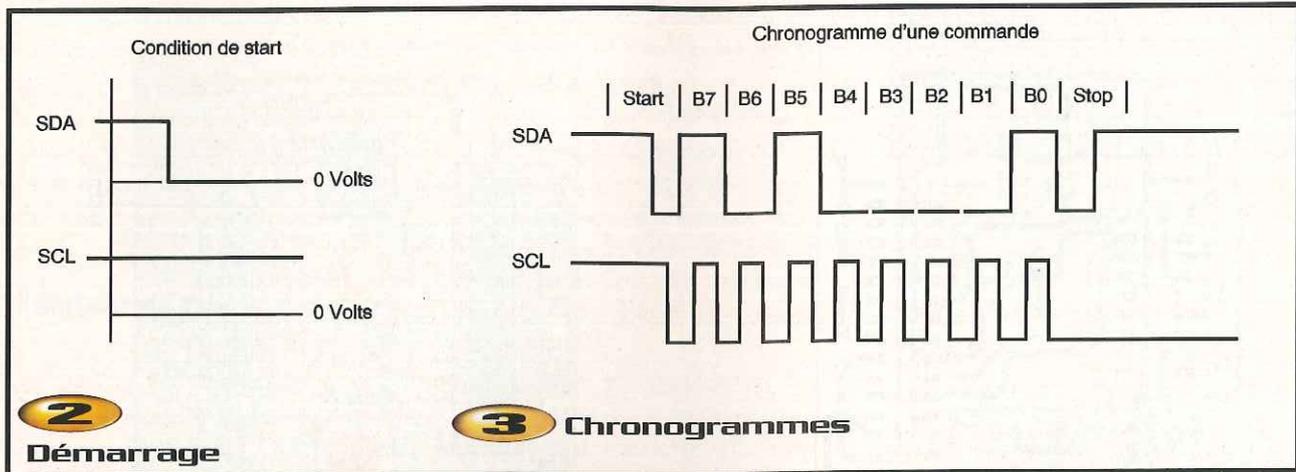
La mémoire 24C16 est direc-

tement câblée sur un port du 16F873.

Le circuit d'auto-maintien de l'alimentation est constitué de 2 transistors et a la particularité de ne consommer aucun courant lorsque la mise sous tension n'est pas demandée.



**1**  
Face avant



**2**  
Démarrage

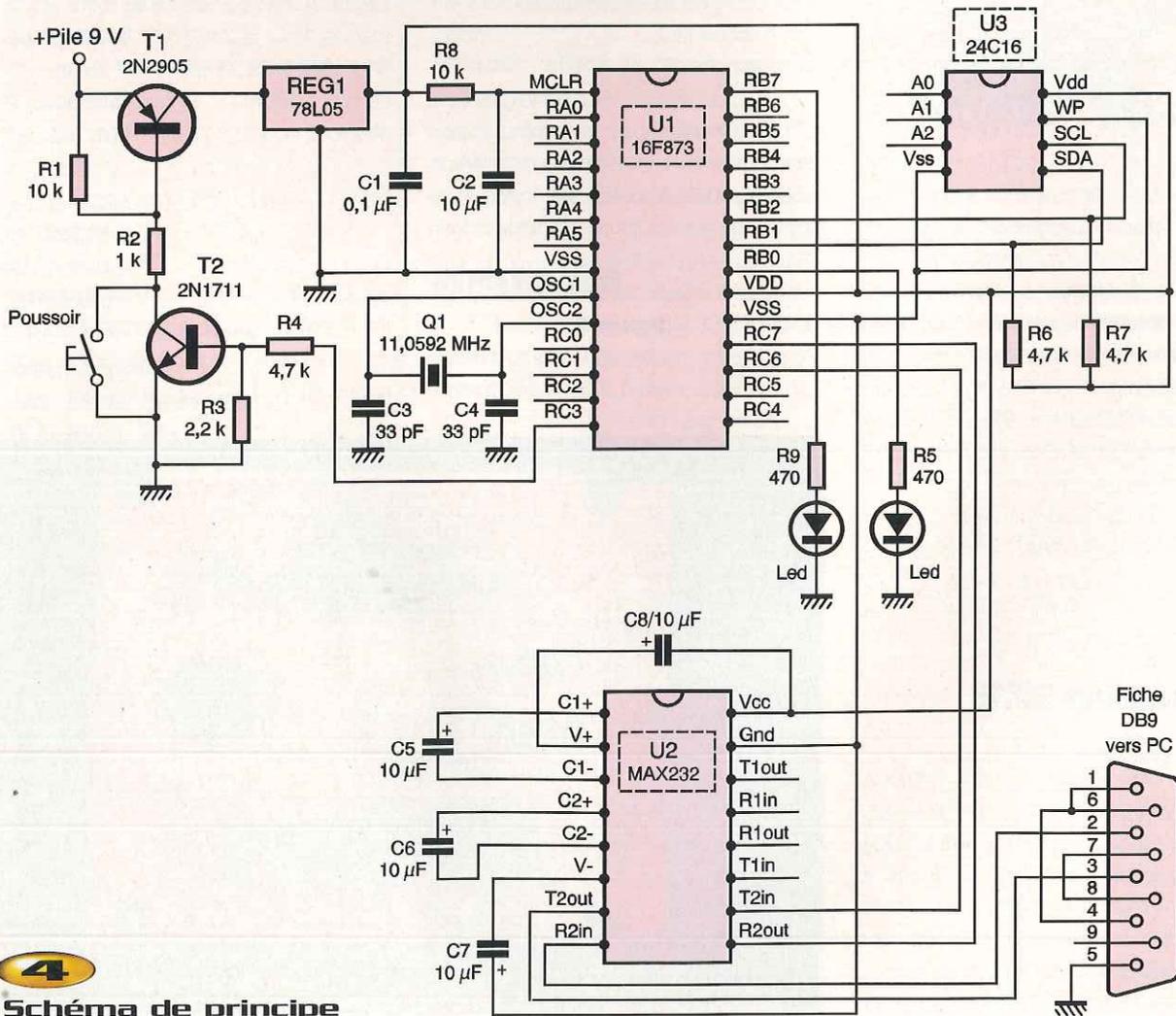
**3** Chronogrammes

**La programmation**

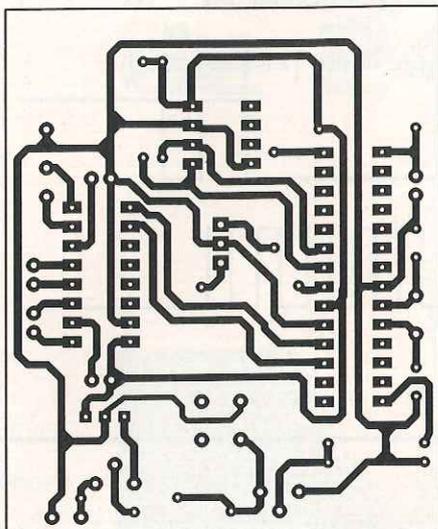
Le téléchargement des données du PC vers le programmeur est réalisé au travers des ports série COM1 à COM4 à une vitesse de 19200 bauds. Le programme envoie l'instruction suivante :  
W , AdresseH , adresseL , bloc de 16 octets

Le programme d'exploitation est écrit en DELPHI 4 et utilise une bibliothèque de communication série freeware que l'on pourra trouver sur la DELPHI SUPER PAGE et installer (il s'agit d'un composant). Pour ceux qui souhaiteraient utiliser directement la version exécutable de cette application, il ne sera pas nécessaire de récupérer ce composant.

L'utilisation du programme ne présente pas de difficultés particulières, le menu fichier permet de charger le fichier à envoyer dans le programmeur, le menu téléchargement permet d'envoyer le fichier au programmeur. Le menu configuration permet, enfin, de sélectionner le port série sur lequel est raccordé le programmeur.

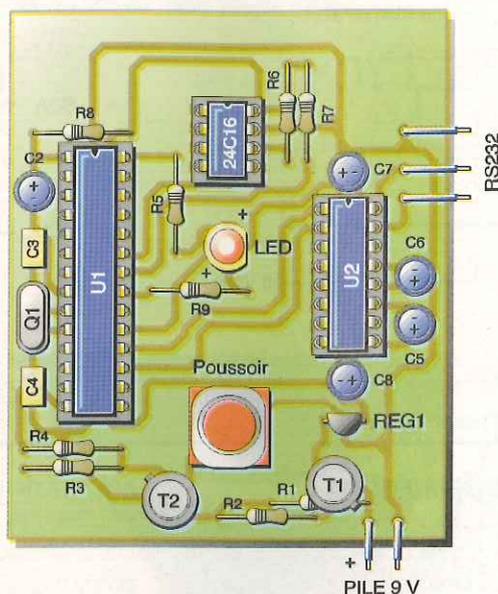


**4**  
Schéma de principe



5

Tracé du circuit imprimé



6

Implantation des éléments

## La réalisation (figures 5 et 6)

Le dessin du circuit imprimé donné destine le circuit au montage dans un coffret de chez CONRAD. On veillera à réaliser des trous oblongs pour le passage des vis ainsi que pour le passage du jack stéréo du châssis.

Une attention particulière sera portée à la mise en place du poussoir de mise en route. En effet, le modèle utilisé a une course de 1/10 ème de millimètre, il sera donc nécessaire de faire un ajustage précis lors de sa mise en place.

Les tensions pourront être vérifiées avant de placer le PIC sur son support en maintenant le bouton de mise en route actionné pendant le temps de la mesure.

La face avant du programmeur a été réalisée avec le logiciel PAINT et imprimée sur du papier jet d'encre autocollant, une protection par film adhésif transparent a été ensuite collé sur l'ensemble.

## Les mémoires 24C16

Les mémoires 24C16 sont des mémoires EEPROM utilisées dans de nombreux systèmes électroniques où la capacité de données à sauvegarder est faible. La 24C16 est une mémoire de 16 kbits, soit 2koctets. Ce composant dialogue en mode série avec le dispositif qui le pilote, ce qui explique le faible nombre de broches de

son boîtier (DIP8).

Pour ce type de mémoires série, 2 protocoles sont principalement utilisés :

- le protocole I2C
- le protocole SPI

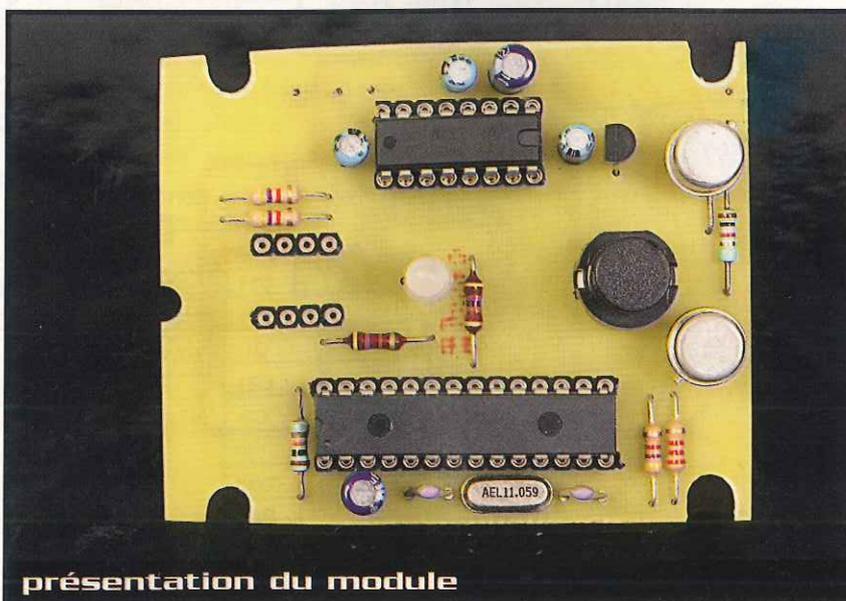
Dans le cas des 24C16, c'est le protocole I2C qui est utilisé pour la communication avec la mémoire. Le dialogue doit commencer en indiquant au boîtier une condition de start, pour cela il suffit de maintenir la ligne SCL au niveau haut et de provoquer une transition du niveau haut vers le niveau bas de la ligne SDA (**figure 2**).

On présente ensuite, sur le bus, un mot de commande constitué d'un code "compo-

sant" et des 3 bits de poids forts de l'adresse. Le code "composant" est constitué des 4 bits de poids fort du mot de commande. Pour la 24C16, le code composant est 0A.

Le bit 0 détermine le type d'opération à effectuer, pour la 24C16 il s'agit d'une lecture ou d'une écriture.

Si B0=1, il s'agit d'une opération de lecture ; si B0=0, il s'agit d'une opération d'écriture. Enfin les bits B1, B2 et B3 sont les 3 bits de poids fort de l'adresse à lire ou à écrire. Pour réaliser une opération de lecture, le mot de commande aura la valeur hexadécimale A1 ou 10100001 en binaire.



présentation du module

Pour réaliser une opération d'écriture, le mot de commande aura la valeur hexadécimale A0 ou 10100000 en binaire.

L'envoi du mot de commande devra être précédé d'un start bit. Le chronogramme de la **figure 3** est le reflet du bus au moment de l'envoi d'un mot de commande.

## Pointeur interne

La 24C16 est dotée d'un pointeur d'adresse interne qui est positionné par une instruction d'écriture. C'est ce pointeur qui

détermine l'adresse courante pour les opérations d'écriture, mais également pour les opérations de lecture.

Pour fixer l'adresse de ce pointeur, il suffit d'envoyer sur le bus une commande d'écriture, puis d'envoyer l'adresse.

Après avoir reçu l'acknowledge du boîtier, on génère une condition de start et un mot de commande de lecture : le boîtier répond alors la donnée pointée par l'adresse définie.

Pour exécuter une opération d'écriture, on envoie sur le bus un mot de commande d'écriture puis l'adresse et, enfin, la donnée à écrire.

## Nomenclature

$R_1, R_8$  : 10 k $\Omega$

$R_2$  : 1 k $\Omega$

$R_3$  : 2,2 k $\Omega$

$R_4, R_6, R_7$  : 4,7 k $\Omega$

$R_5, R_9$  : 470  $\Omega$

$C_1$  : 0,1  $\mu$ F

$C_2, C_5$  à  $C_8$  : 10  $\mu$ F

$C_3, C_4$  : 33 pF

$Q_1$  : quartz bas profil 11,0592 MHz

$T_1$  : 2N2905

$T_2$  : 2N1711

$U_1$  : 16F873

$U_2$  : MAX232

$U_3$  : 24C16

REG<sub>1</sub> : 78L05

LED bicolore

Jack 3,5 mm stéréo femelle

Bouton poussoir

Jack 3,5 mm stéréo mâle

Fiche DB9 femelle

J.P. MANDON

## Bibliographie

MICROCHIP : DataSheet des mémoires 24C16 et des processeurs 16F872

MICROCHIP : DataSheet des processeurs 16F872

PHILLIPS : spécifications techniques du bus I2C

Les sources des programmes DELPHI, ASM ainsi que les fichiers de face avant sont disponibles sur le site de l'auteur : [www.freepic.fr.st](http://www.freepic.fr.st)

## Adresses Web

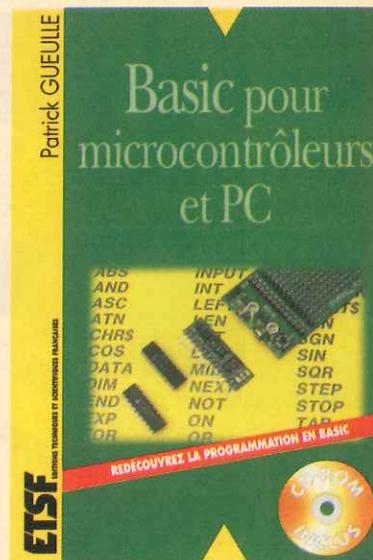
[www.microchip.com](http://www.microchip.com)

[www.phillips.com](http://www.phillips.com)

<http://ftp.optiva.ee/delphi/search.htm>

## Basic pour micro-contrôleurs et PC

Au travers de cet ouvrage, nous vous proposons de découvrir ou de redécouvrir les vertus de la programmation en BASIC.



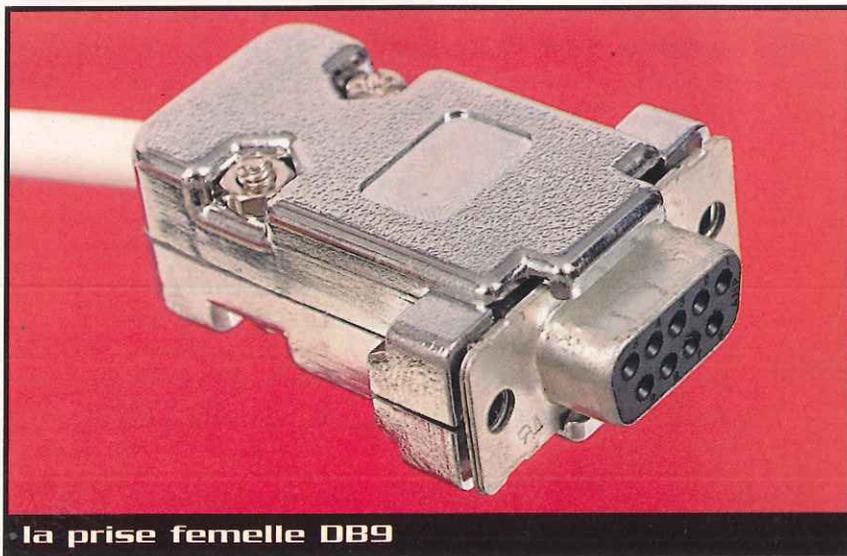
Il vous y sera notamment proposé de récupérer sur votre PC les programmes BASICODE des "vieux" ordinateurs familiaux. Vous y découvrirez également les BASIC qui vous permettront de développer des applications Windows ou encore de programmer des cartes à puce. Et, surtout, même si vous ne savez pas écrire une seule ligne d'assembleur, ce livre vous révélera comment programmer des PIC en BASIC.

Le CDROM contient tous les outils de développement nécessaires, de très nombreux autres programmes très utiles, ainsi que les tracés des circuits imprimés des montages électroniques proposés dans le livre.

Il ne vous en faudra pas davantage pour profiter pleinement de la "nouvelle jeunesse" accordée au langage BASIC.

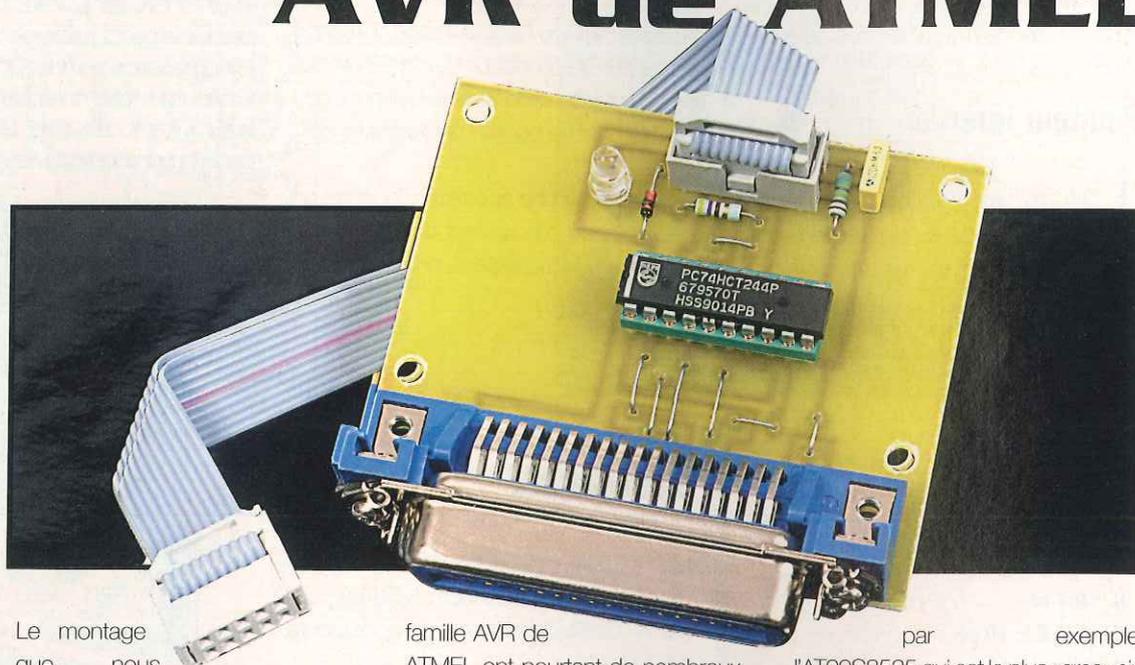
P. GUEULLE - ETSF / DUNOD

152 pages - 198 F (30,18 €)



la prise femelle DB9

# Programmateur en circuit pour microcontrôleurs **AVR de ATMEL**



**Il y a presque deux ans, nous vous proposons dans Interfaces PC de réaliser plusieurs programmeurs pour les microcontrôleurs de la famille AVR de ATMEL. Même s'ils étaient parfaitement fonctionnels, ces programmeurs ne permettaient pas de réaliser toutes les opérations supportées par ces microcontrôleurs.**

**Qui plus est, c'étaient des programmeurs «classiques» en ce sens qu'ils se connectaient sur le port parallèle d'un PC et qu'il fallait ensuite y placer le ou les microcontrôleurs à programmer.**

Le montage que nous allons vous présenter aujourd'hui est radicalement différent car c'est un programmeur «en circuit», c'est à dire que vous n'avez même pas besoin de retirer le microcontrôleur de votre application pour le programmer. De plus, ce programmeur est compatible de l'excellent logiciel de programmation ATMEL qui est mis gratuitement à notre disposition et donc disponible sur le site de la revue.

Ce logiciel, fonctionnant sous Windows 95, 98 et Me, supporte toutes les fonctions de programmation admises par les microcontrôleurs AVR de ATMEL et permet de travailler sur ces derniers avec un maximum de souplesse.

Enfin, et cela ne gêne rien, le prix de revient de notre programmeur est de l'ordre de 100 F alors qu'il est à même de programmer toute la gamme des circuits AVR de ATMEL.

## **Qu'est ce que la famille AVR de ATMEL ?**

Encore relativement peu connus du grand public et même de nombreux industriels, les microcontrôleurs de la

famille AVR de ATMEL ont pourtant de nombreux atouts pour séduire l'un et l'autre. Tout d'abord, ce sont des microcontrôleurs à architecture RISC ce qui, à la manière des célèbres PIC de MICROCHIP, leur confère une puissance et une rapidité d'exécution des programmes peu commune pour des circuits aussi peu coûteux.

Ensuite, et sur ce point ils se démarquent de nombreux microcontrôleurs concurrents, tous les circuits de la famille AVR sont équipés de mémoire de programme de type Flash, c'est à dire de mémoire programmable et effaçable électriquement en quelques secondes. Contrairement à ce qui est le cas pour la majorité des microcontrôleurs équipés de mémoires du même type, les microcontrôleurs ATMEL n'ont besoin d'aucune «haute tension» pendant leur phase de programmation. Ils se contentent donc de leur alimentation normale qui peut être comprise entre 4 et 6V. Ceci étant précisé, voici quels sont les points forts de la famille AVR de ATMEL.

Les circuits sont réalisés en technologie CMOS ce qui leur confère une consommation très faible puisque,

par exemple, l'AT90S8535 qui est le plus «gros» circuit de la gamme AT90 ne consomme que 6,4mA en fonctionnement et 1,9mA en mode attente. Selon les modèles, qui sont presque tous déclinés en gamme standard et en gamme basse consommation, reconnaissable à la lettre L qui apparaît au beau milieu de la référence, la tension d'alimentation peut varier de 4 à 6V pour les modèles standards ou de 2,7 à 6V pour les modèles L. Dans tous les cas, ils sont compatibles des circuits logiques CMOS classiques alimentés sous la même tension. En outre, s'ils sont alimentés entre 4,75 et 5,25V, ils sont également compatibles des circuits TTL en technologies classiques.

La fréquence d'horloge peut être comprise entre 0 et 8, 10 ou 12 MHz (selon les références) pour les modèles standards et entre 0 et 4 MHz pour les modèles L.

La «puissance» de ces circuits, même si nous n'aimons pas ce terme car il ne signifie pas grand chose pour un microcontrôleur dont le rôle est rarement de faire de l'exécution de programme pure et dure, est toutefois évaluée à 8 MIPS pour une fré-

quence d'horloge de 8 MHz ce qui est plus qu'honorable pour de «petits» microcontrôleurs 8 bits.

Tous les microcontrôleurs AVR contiennent de la mémoire de programme de type Flash, c'est à dire encore de la mémoire programmable et effaçable électriquement et ce avec une endurance minimum garantie de 1000 cycles.

Selon les versions, ils contiennent aussi de la mémoire vive, ou RAM, en plus ou moins grande quantité et, surtout, de la mémoire EEPROM de données. Une telle mémoire, qui se programme et s'efface, elle aussi, électriquement, mais cette fois-ci pendant l'exécution d'un programme au moyen d'instructions adéquates, est très utile pour stocker, de manière semi-permanente, certaines informations. C'est par exemple dans de la mémoire de ce type que sont conservés les réglages des chaînes de votre téléviseur ou de votre magnétoscope. Compte tenu du fait que cette mémoire EEPROM de données est susceptible d'être programmée beaucoup plus souvent que la mémoire de programme, son endurance garantie est supérieure à 100 000 cycles d'effacement/écriture.

Les autres ressources internes rencontrées sont classiques sur les microcontrôleurs actuels avec : des ports parallèles, des UART, des interfaces SPI, des convertisseurs analogiques/digitaux, des timers normaux, évolués ou bien encore à fonction chien de garde.

Enfin, pour ceux d'entre-vous qui êtes habitués au langage machine des microcontrôleurs «classiques» (familles 8051 de INTEL ou encore 68HC11 de MOTOROLA par exemple), sachez que le langage machine des AVR de ATMEL en est très proche ce qui facilite son apprentissage. Il ne s'agit là bien sûr que d'une rapide présentation et, si vous voulez en savoir plus sur ces circuits dont le succès commence à faire ombre aux PIC, nous ne saurions trop vous recommander la lecture de l'ouvrage intitulé «Microcontrôleurs AVR - Description et mise en œuvre» que nous leur avons consacré chez DUNOD.

## La programmation en circuit ou ICSP

La programmation en circuit, appelée en général dans les documentations américaines ISP, ce qui veut dire «in system programming» ou encore ICSP pour «in circuit serial programming», est apparue, il y a quelque temps déjà, avec les «gros» circuits logiques programmables tels que les GAL, LCA et autres réseaux de portes. Elle s'est depuis étendue à une partie du monde des microcontrôleurs dont les circuits AVR de ATMEL font partie.

Ce mode de programmation repose sur l'utilisation d'une liaison série un peu particulière, soit qu'elle existe déjà en tant que telle dans le microcontrôleur concerné, soit que certaines de ses pattes se voient

détournées de leur fonction première lorsque cette programmation a lieu.

Dans tous les cas, son principe général est de n'utiliser qu'un minimum de pattes du microcontrôleur afin qu'il soit possible de programmer ce dernier alors qu'il est déjà en place dans l'application finale, quitte à devoir prévoir sur cette dernière quelques commutateurs, manuels ou électroniques, afin d'isoler provisoirement les pattes nécessaires pendant cette phase de programmation.

Dans le cas des microcontrôleurs ATMEL, la liaison série utilisée pour cette programmation en circuit est la liaison série synchrone ou SPI. Elle peut exister en tant que véritable ressource interne, comme c'est le cas sur les plus «gros» circuits de la famille AVR, ou être seulement présente pour cette phase de programmation en circuit, comme c'est le cas pour les plus petits circuits.

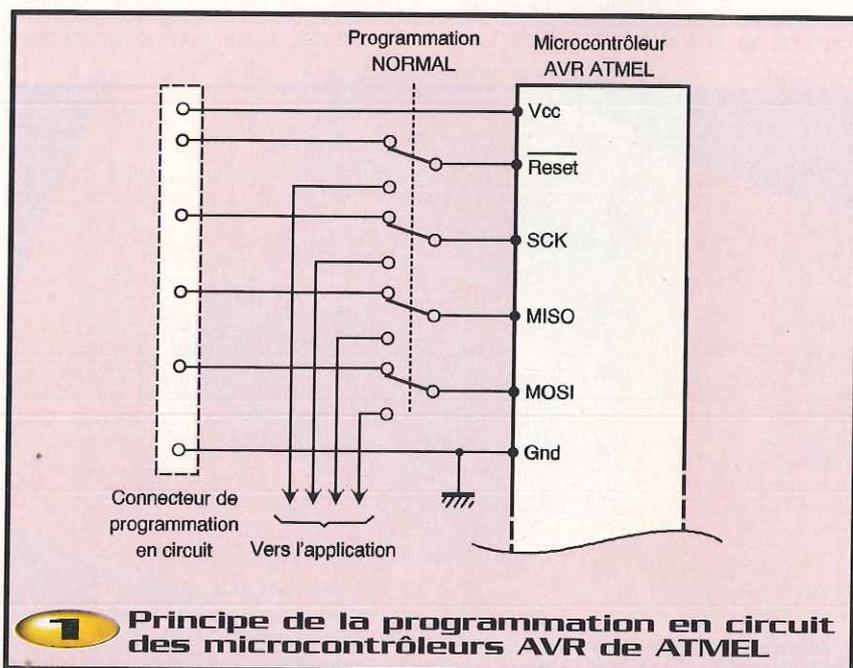
Cette liaison fait appel à trois lignes, dont les appellations se retrouvent d'ailleurs sur tous les microcontrôleurs disposant d'une SPI (68HC11 de MOTOROLA ou 16F876 de MICROCHIP par exemple). On rencontre donc :

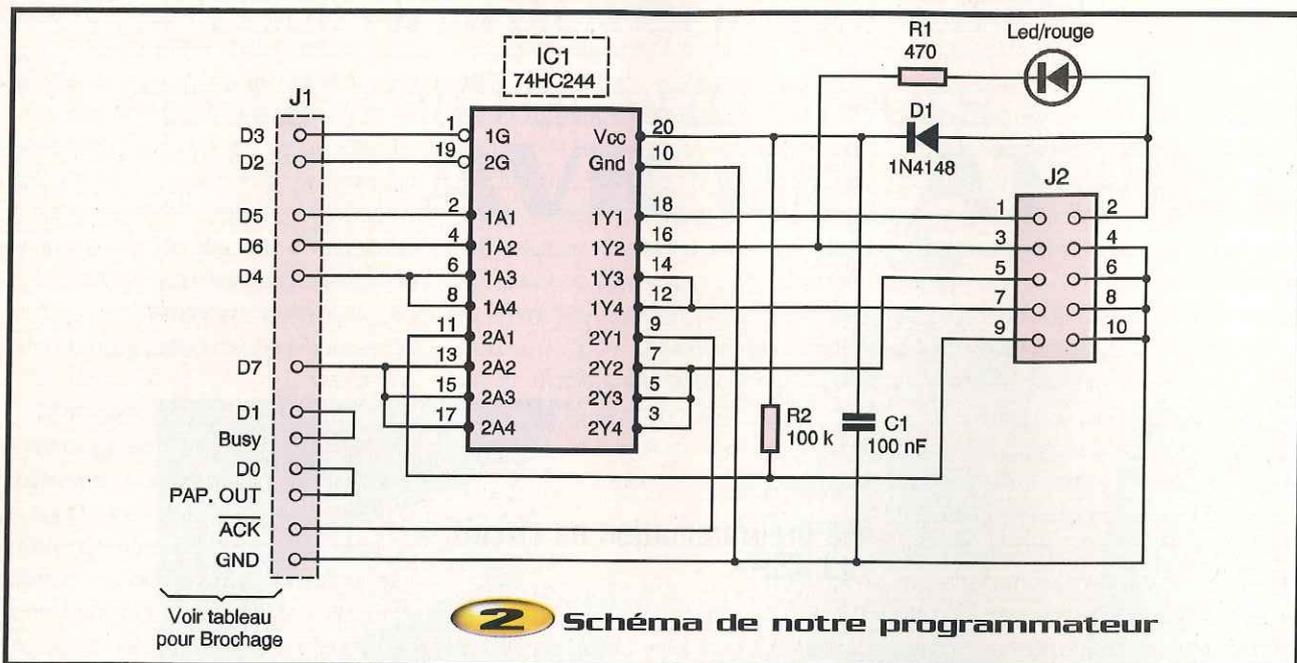
- SCK qui véhicule l'horloge de dialogue,
- MOSI qui véhicule les données dans le sens programmeur vers microcontrôleur et qui sert donc à l'envoi des données à programmer,
- MISO qui véhicule les données dans l'autre sens et qui sert donc à relire le microcontrôleur.

Afin de pouvoir commander correctement le processus de programmation en circuit, il faut aussi pouvoir accéder à l'entrée de reset du microcontrôleur car c'est au moyen de cette dernière que l'on fait passer les circuits ATMEL en mode programmation en circuit.

Une application prévue pour supporter la programmation en circuit d'un microcontrôleur ATMEL adopte donc le schéma de principe présenté **figure 1**. Les straps visibles sur cette figure peuvent évidemment être réalisés de multiples façons : straps ordinaires, interrupteurs mécaniques ou interrupteurs électroniques.

Vous pouvez voir un exemple de mise en œuvre de ce système dans l'article consacré à la réalisation du générateur de fonctions sur port série.





## Schéma du programmeur en circuit

Ceci étant précisé, le schéma d'un programmeur en circuit piloté par PC se résume à très peu de choses. En effet, il faut surtout du logiciel côté PC pour générer les chronogrammes adéquats sur les lignes SCK, MISO et MOSI et, bien sûr, un minimum de circuiterie d'adaptation de niveau afin de pouvoir utiliser un des ports standards du PC.

Le schéma de notre programmeur se réduit donc à ce que vous pouvez découvrir en **figure 2**. Il est destiné à être relié au port parallèle du PC dont les lignes sont faciles à piloter par programme pour générer les chronogrammes nécessaires. Il ne comporte en tout et pour tout qu'un circuit intégré IC<sub>1</sub>, qui est un simple 74HC244 ou 74HCT244, c'est à dire encore un sextuple buffer destiné à remettre en forme les signaux transmis sur la sortie imprimante du PC avant de les appliquer au circuit AVR à programmer. On est sûr ainsi de disposer de signaux bien rectangulaires et d'éviter tout aléa de programmation.

Comme vous pouvez le constater, ce circuit est câblé de façon à « amplifier » les signaux SCK, MOSI et RESET d'une part, puisque ceux-ci vont du PC vers le microcontrôleur, et le seul signal MISO, dans l'autre sens, puisque celui-ci va du microcontrôleur vers le PC.

Remarquez également que l'une des sec-

tions de IC<sub>1</sub>, pilote une LED à partir du PC ce qui permet de signaler, par son allumage, que le microcontrôleur est en cours de programmation et qu'il ne faut donc pas y toucher !

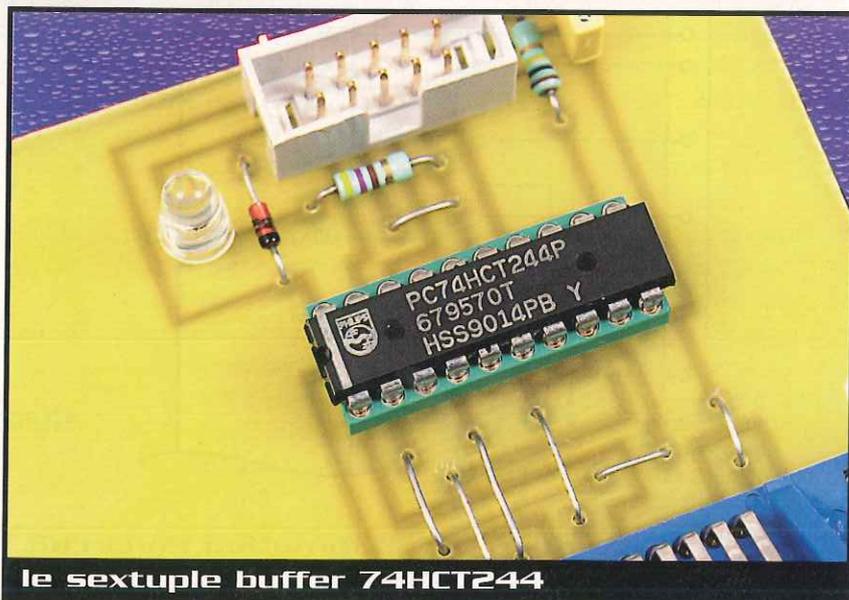
L'alimentation du programmeur est prélevée directement sur l'application supportant le microcontrôleur à programmer, via la diode D<sub>1</sub>.

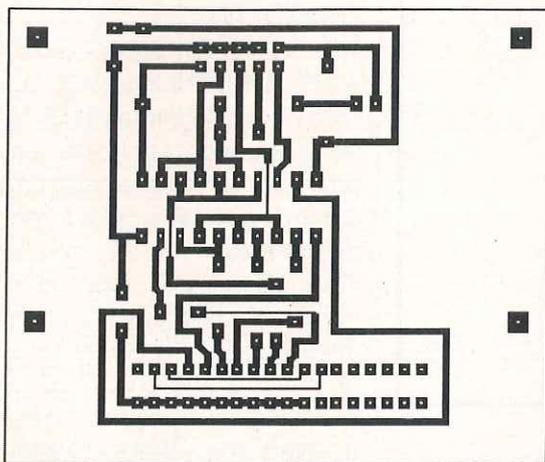
## Réalisation

Nous vous proposons deux circuits imprimés différents pour recevoir notre programmeur. En effet, comme celui-ci se connecte sur le port parallèle, il est pos-

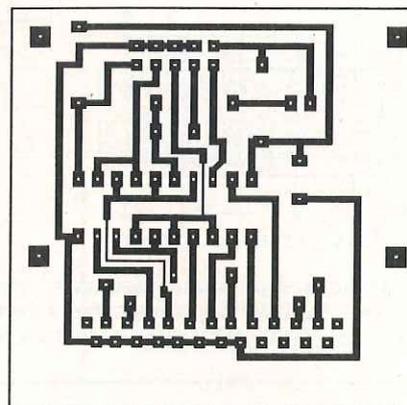
sible de le raccorder au PC directement à la place d'une imprimante au moyen de cordon muni de la prise Centronics normalisée destinée à cette dernière. Il est également possible d'utiliser un port parallèle réservé à cet usage et de faire appel alors à un câble muni de connecteurs standards DB25. Les tailles et brochages de ces connecteurs étant totalement différents, nous avons donc dû dessiner deux circuits imprimés distincts dont les tracés vous sont proposés en **figures 3 et 4**.

Ces circuits reçoivent bien évidemment, l'un comme l'autre, tous les composants du montage, dont le connecteur de pro-





**3** Circuit imprimé (version avec connecteur Centronics)



**4** Circuit imprimé (version avec connecteur DB25)

grammation en circuit qui est un connecteur mâle 10 points pour câble plat. Précisons que le brochage de ce connecteur (**figure 7**) est conforme à ce que préconise ATMEL dans ses documents les plus récents et qu'il est donc identique à celui des programmeurs du commerce.

La réalisation ne présente aucune difficulté en suivant les indications des **figures 5** et **6**, selon le cas.

Le 74HC244 ou HCT244 est impérativement monté sur support car, en cas de grossière erreur de connexion, il est en première ligne.

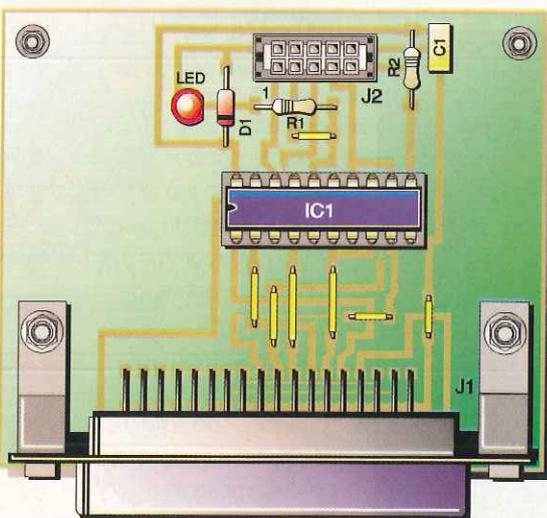
### Mode d'emploi

Pour faire fonctionner notre programmeur, il faut évidemment un logiciel. Fort heureusement, ATMEL nous fournit gracieusement un excellent programme que vous trouverez sur le site de la revue. Commencez par décompresser le fichier qui le contient, baptisé `avr_isp.zip`, dans un répertoire temporaire quelconque puis procédez à l'installation du programme en lançant le programme «setup» apparu lors de cette décompression.

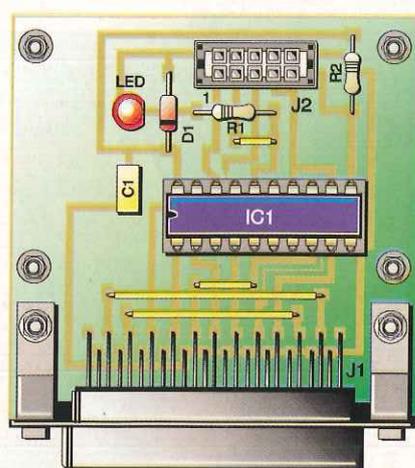
Ne lancez pas tout de suite le programme ainsi installé car, pour qu'il fonctionne de

façon complète, il faut qu'il puisse détecter la présence de notre montage. Celui-ci doit donc évidemment être relié à un port parallèle du PC d'une part, et à l'application contenant le microcontrôleur AVR à programmer d'autre part. Cette liaison est à faire avec un morceau de câble plat à 10 conducteurs dont la longueur ne devra pas dépasser une vingtaine de centimètres environ afin de ne pas dégrader les signaux.

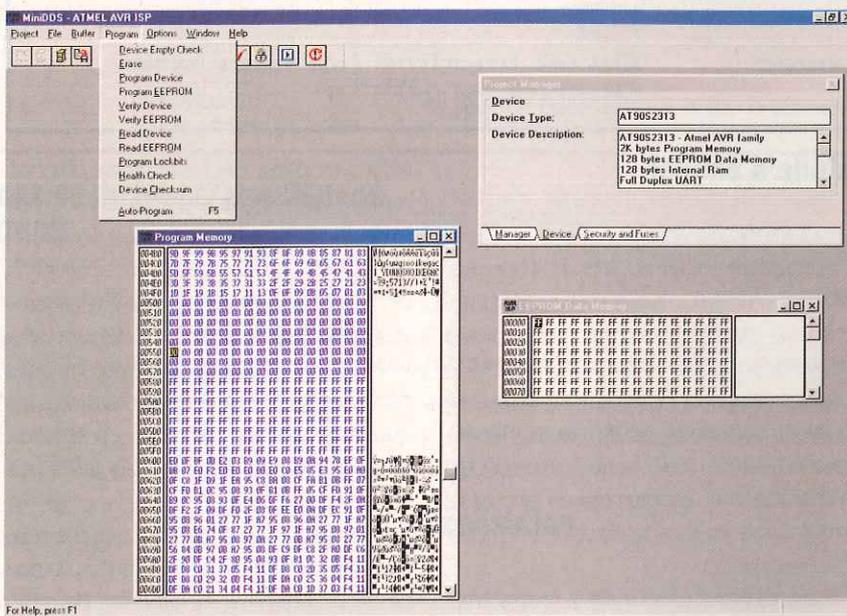
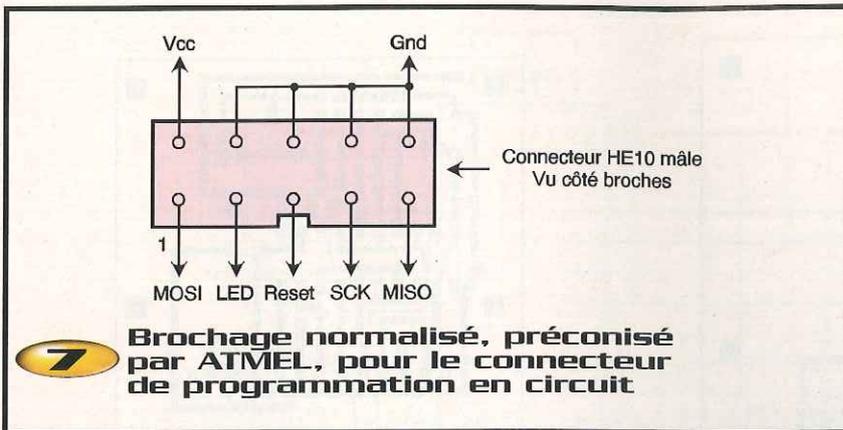
Sur l'application, les straps ou interrupteurs permettant d'isoler les lignes pilotées par le programmeur du reste de cette dernière seront mis en position ouverte et



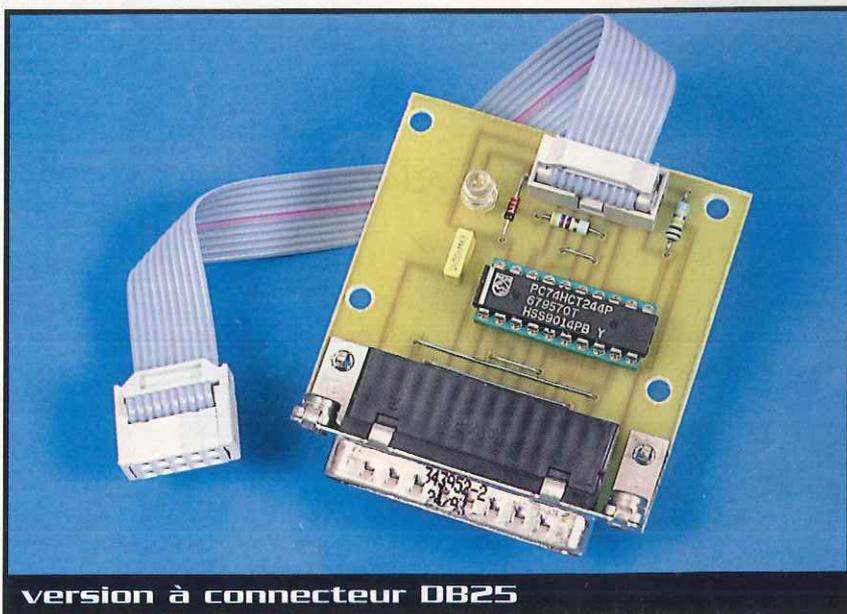
**5** Implantation des composants (version avec connecteur Centronics)



**6** Implantation des composants (version avec connecteur DB25)



**8 Écran principal du logiciel de commande du programmeur**



l'application sera mise sous tension normalement.

Vous pourrez alors lancer le programme ATMEL AVR ISP ce qui vous donnera accès à une fenêtre analogue à celle visible **figure 8**. Cette fenêtre sera initialement vide car il vous faudra tout d'abord définir un nouveau projet au moyen du menu «Project».

Cette définition vous proposera en premier lieu de choisir le type de microcontrôleur à programmer et vous demandera de donner un nom à votre projet. Vous pourrez aussi, à cette occasion, ajouter à votre projet divers commentaires optionnels.

Vous pourrez alors charger dans le logiciel le fichier contenant le programme à placer dans le microcontrôleur au moyen du menu «File», rubrique «Load». Le contenu de ce fichier deviendra alors visible dans la fenêtre «Program Memory» ainsi que, le cas échéant, dans la fenêtre «EEPROM Data Memory» s'il charge aussi cette mémoire. Vous pourrez évidemment éditer librement tout ou partie de ces données, si nécessaire, en cliquant directement dans la fenêtre concernée mais vous pourrez, aussi et surtout, accéder à toutes les fonctions de programmation, lecture et vérification du microcontrôleur au moyen du menu déroulant «Program» que nous avons volontairement ouvert sur la recopie d'écran de la figure 8.

Le lancement de l'une quelconque de ces fonctions conduit à l'ouverture d'une nouvelle fenêtre indiquant son bon déroulement, tandis que la LED de notre programmeur s'allume tant qu'elle est en cours d'exécution.

Sous réserve de n'avoir commis aucune erreur de câblage et d'avoir convenablement sélectionné le microcontrôleur lors de la définition du projet, toutes les opérations de programmation, lecture et vérification doivent se dérouler sans aucun problème.

Précisons, avant de conclure, que le schéma de notre programmeur est identique dans son principe à celui fourni avec le «Starter Kit» STK200 de ATMEL. Il est donc compatible également de tous les logiciels de programmation que vous pourriez trouver et qui seraient présentés comme étant eux-mêmes compatibles de ce «Starter Kit».

Signal	DB25 mâle	Centronics femelle
D0	2	2
D1	3	3
D2	4	4
D3	5	5
D4	6	6
D5	7	7
D6	8	8
D7	9	9
/ACK	10	10
BUSY	11	11
PAPER OUT	12	12
Masse	18 à 25	19 à 29

 **Brochages des connecteurs DB25 et Centronics**

## Nomenclature

- IC<sub>1</sub> : 74HCT244 ou 74HCT244
- D<sub>1</sub> : 1N914 ou 1N4148
- LED : LED rouge
- R<sub>1</sub> : 470 Ω 1/4W 5%  
(jaune, violet, marron)
- R<sub>2</sub> : 100 kΩ 1/4W 5%  
(marron, noir, jaune)
- C<sub>1</sub> : 100 nF Mylar
- J<sub>1</sub> : connecteur DB25 mâle ou Connecteur Centronics femelle, coudé à 90° à implanter sur CI
- J<sub>2</sub> : connecteur HE10 mâle droit pour câble plat à souder sur CI
- 2 connecteurs HE10 femelles à sertir sur câble plat
- Câble plat standard 10 conducteurs (une vingtaine de cm)
- 1 support de CI 20 pattes

### Conclusion

Avec ce programmeur et son logiciel associé, vous êtes à même de programmer avec tout le confort souhaitable la majorité des microcontrôleurs de la famille AVR de .ATMEL et, ce, pour un prix de revient inférieur à 100 F (15,24 Euros). Il

nous semble difficile aujourd'hui de faire mieux !

C. TAVERNIER

## MULTIPROGRAMMATEUR

Superbe programmeur qui peut programmer : PIC16F84A, PIC16F84, PIC16C84; PIC12C508, PIC12C509, PIC16C622, PIC16F628, PIC16F876 et eeproms, Funcards, Jupiter 1 and 2 etc...  
Through-pic programming utilisable pour Goldcards, PICcard2 et Funcards etc. de manière TRANSPARENTE pour l'utilisateur. C'est le programmeur le plus simple à utiliser. Pas besoin de "Loader" pour les cartes Goldwafer... tout se fait AUTOMATIQUEMENT grâce à des mémoires et des PIC intégrés. Programme en une passe les cartes goldwafer, les Funcard 2, les Silvercard 2, les Pic-card v1, v1.1 & v2.

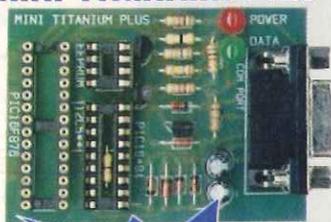


**15 €**  
**Goldwafer**  
tarifs dégressifs par quantité

**30 €**  
**Funcard 2**  
tarifs dégressifs par quantité

**27 €**  
**Silvercard 2**  
tarifs dégressifs par quantité

### Mini Titanium Plus



**34,90€**

### Smarcard + Phoenix 2 en 1



**74,90€**

**106€**

Spécial Playstation 2  
**PUCE  
NEO4**



**EXCLUSIVITE**

OMINFO.COM

BP 207  
52006 CHAUMONT CEDEX  
tél. 03 25 31 47 28  
fax. 03 25 31 69 79  
RCS : CHAUMONT 430 150 292

Achetez sur ce site en toute sécurité

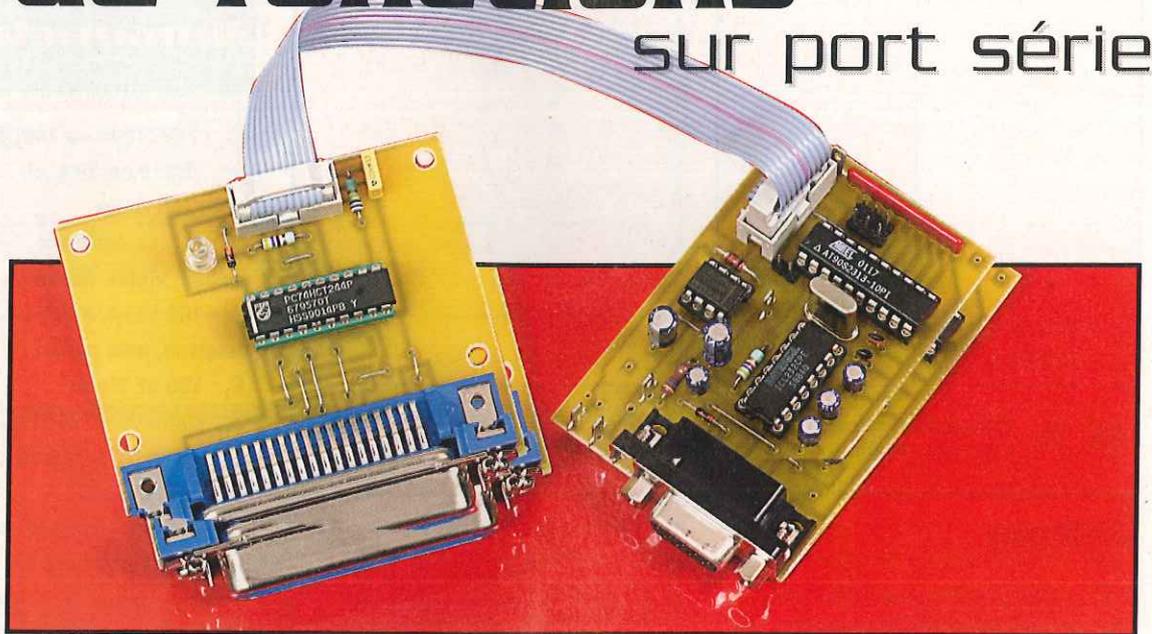


**Livraison 48H**

**www.ominfo.com**

# Générateur de fonctions

## sur port série



**Le générateur de fonctions fait partie des appareils de mesure les plus utiles dans un laboratoire d'électronicien, surtout s'il est capable de produire de nombreuses formes de signaux et si la plage de fréquences qu'il couvre est très étendue.**

**Un tel appareil est cependant relativement coûteux ou, lorsqu'on le fabrique soi-même, nécessite un important câblage avec de nombreux commutateurs et potentiomètres destinés à sélectionner les formes d'ondes et la fréquence des signaux générés.**

C'est pourtant un appareil de ce type que nous vous proposons de réaliser aujourd'hui mais, si vous avez déjà jeté un coup d'œil à cet article, vous avez pu vous rendre compte que notre montage était remarquablement simple et compact et qu'il n'en émergerait pas la traditionnelle forêt de commutateurs et potentiomètres. Malgré cette simplicité, le générateur de fonctions que nous allons vous faire réaliser, dont le prix de revient est seulement de l'ordre de grandeur d'un bon film sur DVD, n'a rien à envier à ses homologues commerciaux : jugez vous-mêmes ...

### Caractéristiques principales

Notre montage est capable de produire des sinusoïdes, des carrés, des triangles et des dents de scie. La fréquence de ces signaux peut être librement programmée de 0,4 Hz à 100 kHz par pas de 1/10 de Hz, La stabilité de fréquence des signaux générés est celle d'un quartz, soit environ  $10^{-6}$ , ce qui fait de notre générateur de fonctions un véritable synthétiseur de fréquence. Notre montage s'alimente sous une

tension unique pouvant varier de 8 à 12V qui n'a pas besoin d'être stabilisée. Il se met sous tension et s'arrête automatiquement lorsque l'on fait appel à lui au moyen de son logiciel de commande.

La programmation des formes d'onde et de la fréquence a lieu au moyen d'une liaison série RS232 normalisée grâce à l'envoi de simples caractères codés en ASCII. Notre montage peut donc être piloté par n'importe quel micro-ordinateur disposant d'une telle liaison et avec n'importe quel langage de programmation. Cependant, pour que celui-ci soit immédiatement opérationnel, nous vous fournissons un beau logiciel à fenêtre fonctionnant sous Windows 95, 98 ou Me capable de piloter toutes ses fonctions.

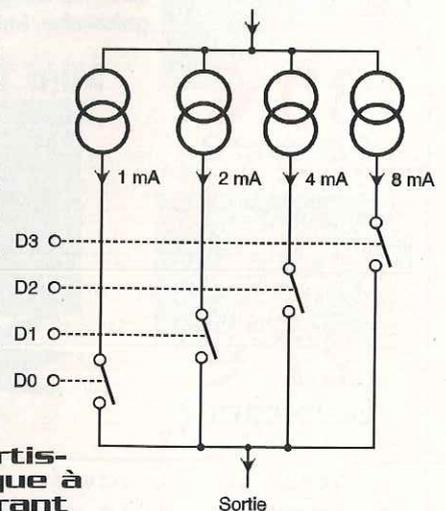
Vous en conviendrez à la lecture de cet exposé, c'est bien un véritable générateur de fonctions que nous vous proposons de réaliser ici mais, ce qui est le plus remarquable, c'est la simplicité avec laquelle nous y

sommes parvenus grâce à la réalisation d'un convertisseur digital/analogique avec un microcontrôleur.

### C'est dans les vieux chaudrons ...

... qu'on fait les meilleures soupes dit le proverbe. Et c'est un vieux chaudron que nous utilisons aujourd'hui puisque l'on fait appel à un convertisseur digital/analogique à réseau R/2R : mais commençons par le commencement.

Si l'on ne veut pas faire appel à la cir-



### Principe d'un convertisseur digital/analogique à générateurs de courant

cuiterie analogique classique, la seule solution pour réaliser un générateur de fonctions est d'utiliser un convertisseur digital/analogique. Même s'il existe de nombreux boîtiers externes capables de réaliser cette fonction et même des microcontrôleurs disposant en interne d'une telle ressource, ce n'est pas la solution que nous avons choisie. En effet, sous réserve d'utiliser un microcontrôleur suffisamment rapide, il est très facile de transformer n'importe lequel de ses ports parallèles 8 bits en convertisseur digital/analogique. Il suffit pour cela d'utiliser un réseau  $R/2R$ .

Voyons rapidement comment cela fonctionne car, lorsque nous l'aurons étudié, nous aurons vu 99% du schéma de notre montage ! Et, pour bien comprendre son principe, commençons par faire un court rappel de numération binaire.

Lorsque nous écrivons le nombre binaire 1010 par exemple, cela signifie en fait :  $1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$ .

En d'autres termes, les chiffres binaires qui composent le nombre définissent les puissances de 2 successives à ajouter pour parvenir à son équivalent décimal. Une réalisation électrique d'un tel système peut être envisagée sous la forme présentée **figure 1**.

Nous y voyons quatre générateurs à courant constant, étagés selon les puissances successives de 2. Le premier génère un courant de 1mA, le second de 2mA et ainsi de suite jusqu'au dernier qui produit 8mA c'est à dire  $2^3$ mA.

Si notre mot binaire à convertir en analogique commande des interrupteurs placés en série avec ces générateurs, de telle sorte qu'un 1 fasse fermer l'interrupteur et qu'un 0 le fasse ouvrir, on va disposer en sortie de ce montage d'une source de courant qui sera l'équivalent décimal exact du mot binaire appliqué à l'entrée.

Ainsi, dans le cas de notre mot binaire 1010, on va relier les générateurs de 8mA et de 2mA à la sortie et disposer ainsi d'une source de courant de 10mA. Or 10 est bien l'équivalent décimal de 1010.

Ce principe étant vu, il ne reste plus qu'à lui trouver une réalisation pratique aussi astucieuse que possible pour disposer d'un véritable convertisseur digital/analogique. Une des méthodes les plus employées sur les convertisseurs de résolution égale ou inférieure à 8 bits est celle du réseau  $R/2R$

schématisé **figure 2**. Nous y voyons la présence de deux valeurs de résistances seulement, des résistances de valeur  $R$  et des résistances de valeur double, soit  $2R$ , ce qui facilite sa réalisation pratique.

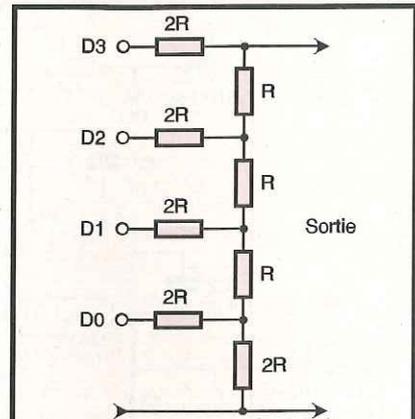
L'entrée D0, ou entrée de poids le plus faible, voit deux résistances de  $2R$  qui forment un diviseur de tension de valeur 2. La moitié de D0 est donc ajoutée au signal suivant résultant de D1. L'entrée D1, quant à elle, voit une résistance de valeur  $2R$  en série avec le réseau de retour à la masse. L'application du théorème de Thévenin à cet ensemble montre qu'en fait, D1 voit un nouveau diviseur par 2 et que la moitié de D1, et donc le quart de D0, se trouve ajoutée au signal résultant de D2.

On pourrait poursuivre cet examen jusqu'à D3 mais le principe montre à l'évidence que le choix des valeurs de résistances et de leur mode de connexion fait que le signal de sortie est bien la somme pondérée des puissances de 2.

C'est ce principe qui est mis en œuvre en sortie du port parallèle du microcontrôleur qui constitue le cœur de notre montage, non pas sur 4 bits comme dans cet exemple, mais sur 8 bits pour disposer d'une résolution suffisante, c'est à dire encore de 256 échantillons distincts pour produire nos signaux.

## Notre schéma

Après cet exposé, le schéma que nous vous présentons **figure 3** se laisse facilement analyser. Le cœur du montage est



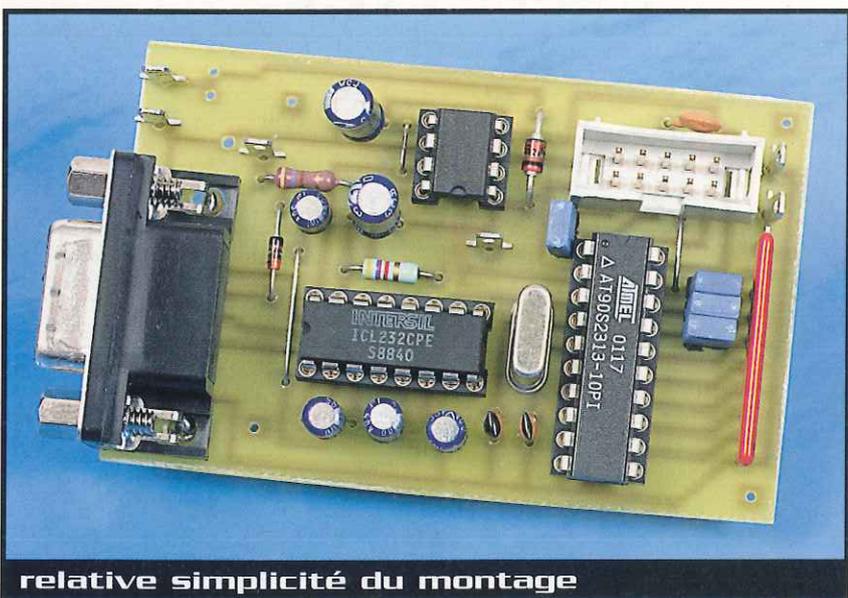
**2**

## Principe d'un convertisseur digital/analogique à réseau $R/2R$

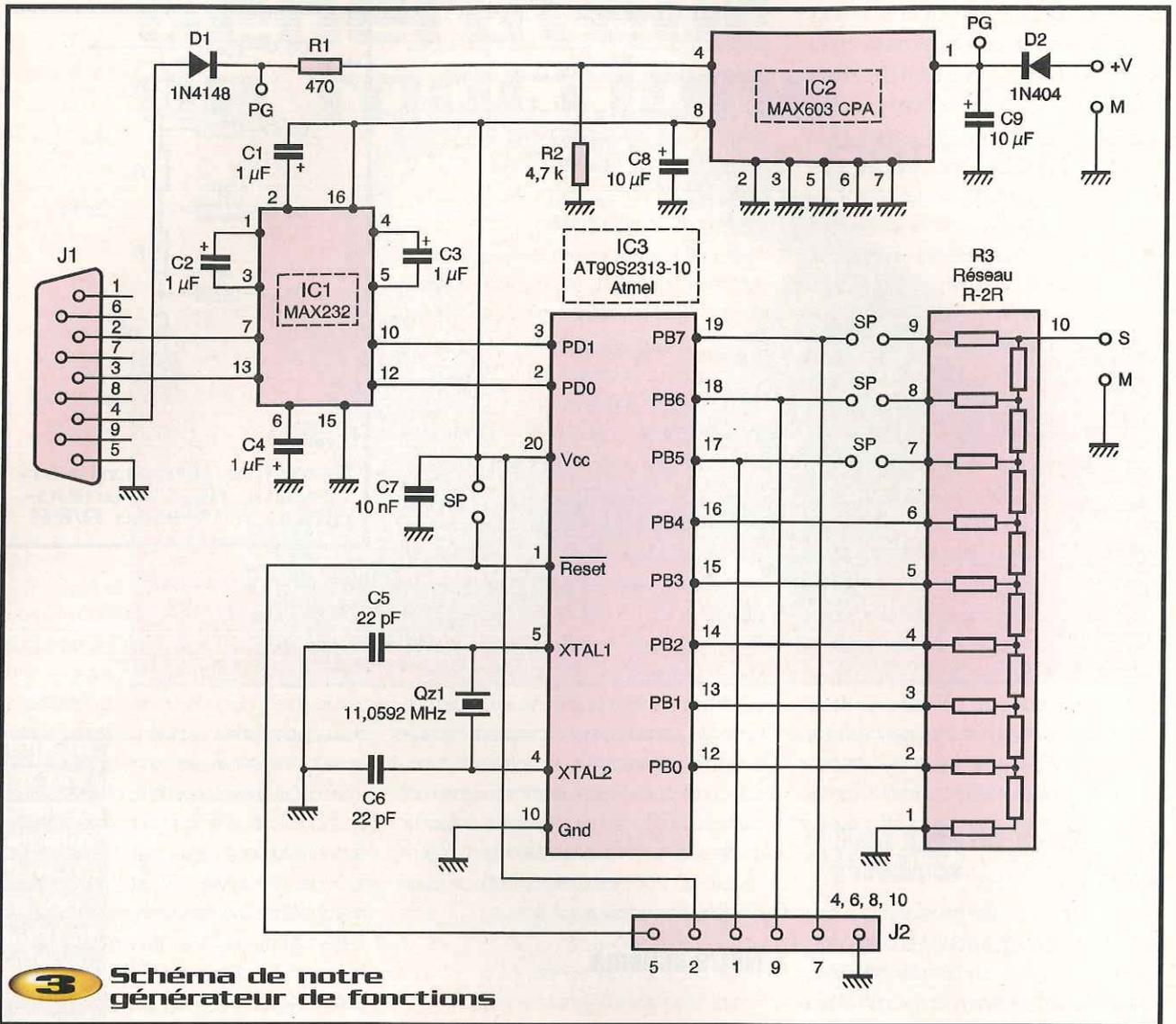
évidemment le microcontrôleur IC<sub>3</sub>.

Il est alimenté par l'intermédiaire de IC<sub>2</sub> qui est un régulateur de tension 5V, analogue dans son principe aux régulateurs trois pattes bien connus, mais qui présente deux particularités supplémentaires : il dispose d'une entrée de commande externe marche/arrêt sur sa patte 4 et, lorsqu'il est en mode arrêt, il ne consomme quasiment aucun courant. Il nous sert donc tout à la fois de régulateur 5V et d'interrupteur marche/arrêt à commande électrique.

Cette commande a lieu via la résistance R<sub>1</sub>, la diode D<sub>1</sub> et le signal DTR de la liaison série RS232 utilisée pour programmer notre générateur. Les signaux de données de cette liaison série se trouvent convertis en TTL dans les deux sens, c'est à dire en



relative simplicité du montage

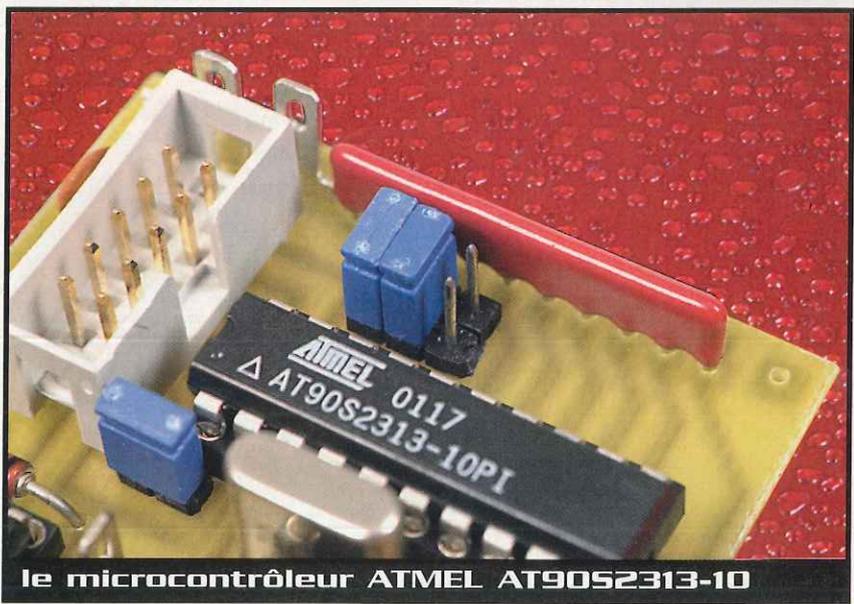


**3** Schéma de notre générateur de fonctions

émission et en réception, par un classique MAX232 repéré IC<sub>1</sub>, qui assure l'interfaçage

avec les entrée et sortie séries du microcontrôleur.

Ce dernier est un microcontrôleur RISC de la famille AVR de ATMEL qui s'avère particulièrement intéressante. Si vous voulez en savoir plus sur cette famille, capable aujourd'hui de concurrencer avec succès les célèbres PIC de MICROCHIP, nous ne saurions trop vous recommander la lecture de l'ouvrage que nous leur avons consacré chez DUNOD sous le titre «Microcontrôleurs AVR - Description et mise en œuvre». Vous pouvez aussi, pour une présentation rapide des points forts de ces produits, lire l'article «programmeur en circuit pour microcontrôleurs AVR ATMEL». Toujours est-il que notre AT90S2313, puisque telle est sa référence, fonctionne avec un quartz à 11,0592 MHz. Il dispose d'un UART interne qui lui permet de s'interfacer très facilement avec la liaison série, une fois celle-ci convertie au niveau TTL par IC<sub>1</sub>. Il dispose aussi d'un port parallèle



le microcontrôleur ATMEL AT90S2313-10

8 bits, PBO à PB7, qui est relié au réseau R/2R.

La qualité de la conversion dépendant de la précision des résistances de ce réseau ou, plus précisément, de l'exactitude d'égalité de toutes les résistances R entre elles ainsi que de l'exactitude du rapport 2 entre R et 2R, nous avons fait appel à un réseau intégré qui respecte très bien ces critères et qui s'avère moins coûteux que les seize résistances de précision qu'il remplace ! Remarquez que la sortie de notre montage est directe après ce réseau R/2R, ce qui fait que l'on dispose de signaux de pratiquement 5V d'amplitude crête à crête. Nous vous laissons le soin de faire suivre cette sortie d'un atténuateur ou d'un amplificateur tampon (buffer) en fonction de vos besoins. N'importe quel schéma utilisé en sortie d'un générateur de fonctions «classique» peut en effet être mis en place ici. Reste à voir les différents straps SP présents sur le schéma et dont nous n'avons pas encore parlé. Ils sont destinés à la programmation en circuit du microcontrôleur. En fonctionnement normal du synthétiseur, ils sont tous mis en place. Par contre, lorsque vous allez réaliser le montage, il vous faudra bien programmer le microcontrôleur. Pour cela, il suffira d'enlever tous les straps SP et de brancher la «tête» de programmation en circuit sur le connecteur J<sub>2</sub>. La programmation se fait alors en moins d'une minute sans nécessiter la moindre «haute tension» de 12V externe comme c'est souvent le cas avec d'autres microcontrôleurs. C'est encore là un point fort des AVR de ATMEL !

## La réalisation

Dans la nomenclature proposée, seuls deux composants peuvent vous faire chercher un peu : le réseau R/2R et le MAX603 disponibles, entre autres, chez FARNELL. Tous les autres composants sont classiques et sont disponibles chez tout revendeur digne de ce nom.

Le circuit imprimé **figure 4** supporte tous les composants du montage, connecteur DB9 pour la liaison série compris. Pour ce qui est de la connexion à l'alimentation, il peut recevoir deux cosses classiques ou bien un jack normalisé de 2,1mm selon vos préférences.

L'implantation des composants est à réali-

ser en suivant les indications de la **figure 5**. Vous procéderez dans l'ordre habituel : straps, supports de circuits intégrés, composants passifs et, en dernier, composants actifs. Attention à l'orientation des composants polarisés et du réseau R/2R : sa patte 1, orientée du côté opposé à J<sub>2</sub>, est repérée par un trait ou un point sur son boîtier. Les straps SP sont réalisés avec des picots mâles au pas de 2,54 mm sur lesquels viennent prendre place des cavaliers de court-circuit au même pas. Le connecteur de programmation J<sub>2</sub> est un connecteur normalisé mâle à 10 points pour câble plat. Son brochage est évidemment compatible de celui du programmeur pour AVR de ATMEL mais il est également compatible des programmeurs en circuit commerciaux destinés à ces mêmes microcontrôleurs.

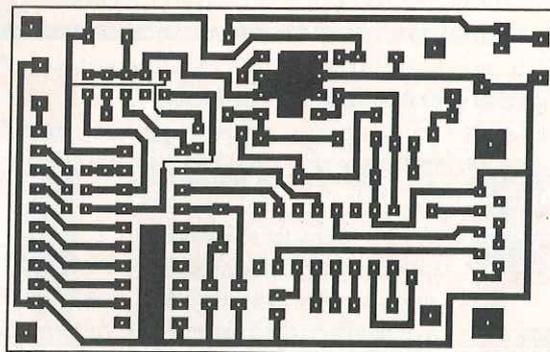
## Programmation et utilisation

La première opération à réaliser consiste à programmer l'AT90S2313 avec le fichier adéquat qui est disponible sur le site de la

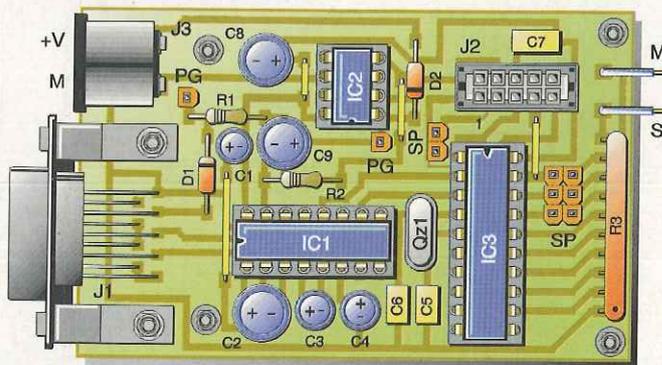
revue sous l'appellation synthavr.hex. Son format est compatible de tous les programmeurs pour microcontrôleurs AVR de ATMEL.

Si vous désirez réaliser cette programmation en circuit, avec un montage du commerce ou avec celui publié, procédez de la façon suivante :

- Enlevez tous les straps SP,
- Connectez le connecteur provenant du programmeur en circuit sur le connecteur J<sub>2</sub>. Ce connecteur est en principe détrompé mais, s'il ne l'est pas, remarquez que sa pinoche n°1 se trouve à côté de la patte 1 du microcontrôleur (voir figure 5),
- Court-circuitez les deux points repérés PG sur le circuit imprimé afin de valider «de force» le régulateur de tension MAX 603,
- Connectez l'alimentation 8 à 12V au montage (une simple pile de 9V suffit),
- Lancez le logiciel de programmation en circuit dans lequel vous chargerez le fichier synthavr.hex évoqué ci-dessus,
- Programmez l'AT90S2313, ce qui prend moins d'une minute,
- Coupez l'alimentation du montage,



### 4 Circuit imprimé



### 5 Implantation des composants

- Enlevez le court-circuit entre les points PG,
- Remettez en place tous les straps SP,
- Votre synthétiseur est alors prêt à fonctionner.

A ce stade des opérations, et même sans aucun logiciel de commande, vous pouvez mettre le montage sous tension pour vérifier son fonctionnement. En effet, il se configure automatiquement en mode sinusoïde à 1 kHz après toute mise sous tension. Vous pouvez donc vérifier la présence de cette dernière en sortie sous réserve de «forcer» à nouveau le MAX603 en court-circuitant provisoirement les deux points PG du circuit imprimé.

L'alimentation peut être confiée à un bloc secteur «prise de courant» délivrant de 8 à 12V sous une centaine de mA ou à une simple pile 9V pour des utilisations occasionnelles. La consommation est quasi nulle en mode arrêt, ce qui rend inutile tout interrupteur, et elle est inférieure à 20mA en fonctionnement.

Pour programmer votre générateur, vous pouvez écrire votre propre logiciel, car nous verrons dans un instant le principe à respecter, mais vous pouvez aussi faire appel à celui que nous vous proposons sur notre site.

Il se nomme synthe.zip et vous devez d'abord le décompresser dans le répertoire temporaire de votre choix. Lancez ensuite son installation au moyen du programme «setup» apparu suite à cette décompression non sans avoir au préalable fermé toutes les applications en cours d'exécution.

Connectez votre générateur à un des ports

séries COM1 ou COM2 de votre PC (COM2 par défaut) et lancez le logiciel que vous venez d'installer en double cliquant sur son icône, ce qui doit faire afficher une fenêtre analogue à celle de la **figure 6**. Le logiciel démarre par défaut sur le port COM2 mais vous pouvez modifier ce choix en cochant la case COM1 si nécessaire. Les fonctions du programme parlent d'elles-mêmes. Une pression sur le bouton «Fréquence ?» fait afficher au-dessus de lui la fréquence réelle produite. Celle-ci peut être choisie en entrant directement sa valeur numérique dans la boîte de texte prévue à cet effet dans la zone «Fréquence demandée» Les valeurs inférieures à 0,4 Hz et supérieures à 100 kHz sont automatiquement rejetées avec un message d'information, de même que toute valeur non numérique. Cette fréquence est transmise au générateur en appuyant sur le bouton «Validation», ce qui est concrétisé par l'apparition de la valeur réellement générée dans la fenêtre «Fréquence en cours».

Le changement de forme d'onde, quant à lui, est pris immédiatement en compte dès que la case correspondante est cochée sur la gauche de l'écran.

Si vous voulez écrire votre propre programme de commande du générateur, pour réaliser un banc de mesure automatisé par exemple, c'est fort simple car le dialogue avec ce dernier a lieu au moyen de simples caractères ASCII envoyés et reçus sur la liaison série.

La sélection de la forme d'onde se fait ainsi en envoyant un seul caractère numérique, codé en ASCII, comme indiqué tableau 1.

Suite à cette commande, le générateur ne répond rien mais il se met à produire immédiatement la forme d'onde demandée à la fréquence en cours de validité.

L'interrogation de fréquence en cours se fait en envoyant au générateur le caractère «point d'interrogation» toujours codé en ASCII. Le générateur répond alors par une trame de six caractères dans cet ordre : la lettre F (comme fréquence) codée en ASCII, un octet nul, trois octets significatifs (voir ci-dessous pour leur codage) et le caractère saut ligne de code ASCII 0A en hexadécimal ou 10 en décimal.

La programmation de la fréquence désirée se fait d'une manière très proche puisqu'il faut envoyer au générateur la trame de six caractères suivante : la lettre s minuscule codée en ASCII, un octet nul et trois octets significatifs (voir ci-dessous pour leur codage). Le générateur répond alors avec la même trame que celle présentée ci-dessus pour l'interrogation de fréquence en cours, ce qui permet de vérifier qu'il a bien reçu la commande.

Reste à voir le codage des trois octets significatifs qui n'est pas très compliqué mais qui demande un peu de réflexion. La fréquence produite par le générateur est donnée par la relation :

$$F = K \times 0,0732421 \text{ (F en Hertz)}$$

C'est la valeur de K qui doit être transmise au générateur dans les trois octets significatifs (ou qui est envoyée par le générateur en réponse à une interrogation) en respectant le principe suivant :

- La valeur de K extraite de la relation ci-dessus doit être convertie de décimal en hexadécimal,

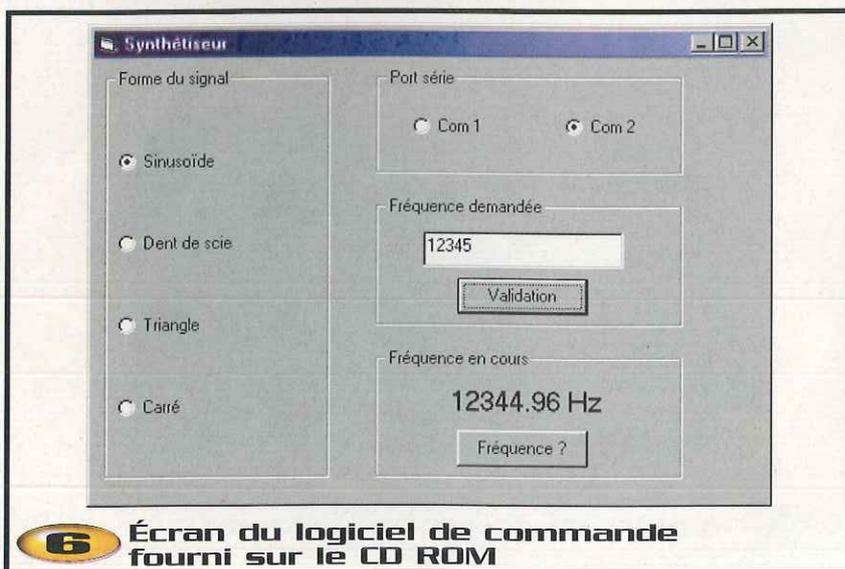
- Le nombre obtenu, composé de 1 à 6 chiffres selon le cas, doit être normalisé à 6 chiffres en le faisant précéder d'autant de zéros que nécessaire (ainsi 25 doit être écrit 00 00 25 par exemple),

- Le nombre de 6 chiffres ainsi obtenu doit être découpé en groupes de 2 chiffres qui constituent chacun un des octets significatifs,

- Ces trois octets doivent être envoyés dans l'ordre : poids forts, poids intermédiaires, poids faibles.

Un exemple parlant aussi bien qu'un long discours, voici comment procéder pour programmer la fréquence de 1000 Hz. Nous avons tout d'abord :

$$K_{\text{décimal}} = F / 0,0732421$$



**6** Écran du logiciel de commande fourni sur le CD ROM

Ce qui donne  $K_{\text{décimal}} = 13653$  ou encore  $K_{\text{hexadécimal}} = 3555$

Nous «normalisons»  $K_{\text{hexadécimal}}$  à six chiffres. Il s'écrit alors : 00 35 55.

La trame envoyée au générateur pour programmer cette fréquence est donc : s 00 00 35 55 (attention le «s» est en minuscule).

Comme vous pouvez le constater, cette programmation des fonctions du générateur est particulièrement simple et peut être écrite dans le langage de votre choix. Il suffit juste que celui-ci sache manipuler des caractères sur une liaison série ce qui est à la portée du moins élaboré des langages.

Mais, bien sûr, cette programmation n'est nullement indispensable puisque le logiciel que nous vous fournissons sur le CD associé à ce numéro d'Interfaces PC ou sur notre site permet de piloter avec un maximum de confort toutes les fonctions de notre générateur.

C. TAVERNIER

Caractère	Code ASCII (décimal)	Code ASCII (hexadécimal)	Forme d'onde
1	49	31	Sinusoïde
2	50	32	Dents de scie
3	51	33	Triangles
4	52	34	Carrés

**Caractères de sélection de la forme d'onde**

## Nomenclature

IC<sub>1</sub> : MAX232

IC<sub>2</sub> : MAX603 CPA (FARNELL)

IC<sub>3</sub> : AT90S2313-10 ATMEL

D<sub>1</sub> : 1N914 ou 1N4148

D<sub>2</sub> : 1N4004

R<sub>1</sub> : 470 Ω 1/4W 5%

(jaune, violet, marron)

R<sub>2</sub> : 4,7 kΩ 1/4W 5% (jaune, violet, rouge)

R<sub>3</sub> : Réseau R/2R 10 kΩ - 20 kΩ (FARNELL)

C<sub>1</sub> à C<sub>4</sub> : 1 µF/25V chimique radial

C<sub>5</sub>, C<sub>6</sub> : 22 pF céramique

C<sub>7</sub> : 10 nF céramique

C<sub>8</sub>, C<sub>9</sub> : 10 µF/25V chimique radial

Qz : quartz 11,0592 MHz en boîtier HC18/U ou HC49/U

J<sub>1</sub> : connecteur DB9 femelle soudé à 90° à implanter sur CI

J<sub>2</sub> : connecteur mâle droit 10 points pour câble plat à implanter sur CI

SP : 4 x 2 picots mâles au pas de 2,54mm et cavaliers de court-circuit correspondants

1 support de CI 20 pattes

1 support de CI 16 pattes

1 support de CI 8 pattes

Jack 2,1mm à implanter sur CI (facultatif - alimentation)

## Environnement de Développement

### Basic Tiger :

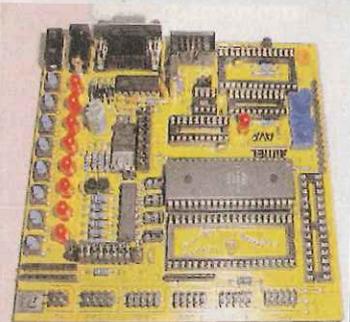
- \* Basic Multitâches
  - \* Mise au point sur carte
  - \* Drivers pour périphériques
  - \* Jusqu'à 4MB de Flash
  - \* Jusqu'à 1920 E/S Num ou Ana
- Starter kit 1 : 1247 F TTC



### AVR :

\* Carte de développement AVR  
STK200 : 635 F TTC

\* Compilateur Basic avec simulateur intégré, gestion du bus I2C, 1 Wire, SPI, lcd, Bus Can : 773 F TTC



Carte d'application montée format barrette mémoire avec AVR 2313 : 316 F TTC, avec AVR 8535 : 427 F TTC

**PIC** : Compilateurs C, Basic disponibles.

**optiminfo**

www.optiminfo.com

Route de Ménétreau  
18240 Boulleret  
Tel : 0820 900 021  
Fax : 0820 900 126

www.puissance3.fr



Puissance 3

# La Puissance par 3

## PRODUCTION

- Gravure de CD Rom

## PRÉCISION

- Rapidité garantie
- Matériel de production certifié

## PERSONNALISATION

- Impression de vos CD-R
- Quadri, jet d'encre
- Fournitures boîtiers, pochettes...

Siège social :

15, avenue des Grenots - SUDESSOR  
91150 Etampes

Tél. : 01 69 16 17 33

Fax : 01 69 16 17 34

e-mail : commercial@puissance3.fr

Bureau Commercial :

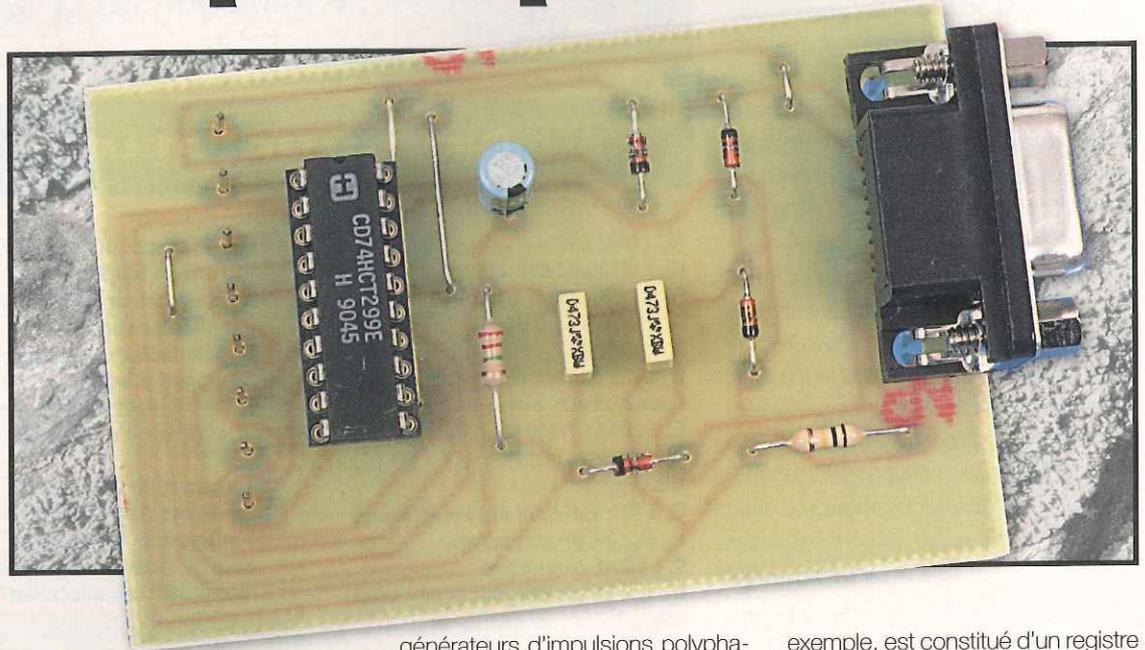
100, rue Emile COSSONNEAU

93330 Neuilly sur Marne

### LES SERVICES PUISSANCE 3

- Gravure de CD-R
- Pressage de CD Rom
- Duplication de disquettes
- Packaging
- Imprimerie
- Conditionnement
- Stockage
- Routage

# Transformation d'un port série en port parallèle



**Le circuit d'interface décrit dans cet article propose la génération d'un port parallèle bidirectionnel 8 bits à travers le port série d'un ordinateur de type PC. Ce port série fournit aussi les tensions d'alimentation nécessaires pour le fonctionnement de ce montage, donc aucune alimentation extérieure.**

## Étude du montage

Nous allons, dans une première partie, expliquer le fonctionnement d'un registre à décalage pour ensuite, dans une seconde partie, utiliser ce mécanisme dans notre application.

Partout où des informations numériques doivent être traitées, mémorisées temporairement ou tout simplement transmises, on a recours à la fonction de registre à décalage. Que ce soit sous forme directe, comme un convertisseur parallèle/série ou série/parallèle par exemple, ou bien sous forme indirecte comme bloc fonctionnel multiplicateur/diviseur rapide pour les opérations arithmétiques numériques, on rencontre les registres à décalage dans presque tous les domaines du traitement numérique de l'information. Parmi les autres domaines d'application, on compte les mémoires intermédiaires, petites mais rapides, les registres collecteurs de données avec une vitesse de transmission différente, les

générateurs d'impulsions polyphasés et les mémoires circulantes (compteur à registre à décalage). Outre le mode de fonctionnement général, les paragraphes suivants traitent des différences fondamentales de conception et de fonction en tenant compte des temps de propagation. Les registres à décalage sont constitués de bascules montées en série, de sorte que la sortie d'une bascule est reliée à l'entrée de la suivante. Si on applique une information à l'entrée de cette chaîne de bascules, cette information est prise en charge avec l'impulsion d'horloge suivante et mémorisée dans la première bascule. L'impulsion d'horloge suivante décale alors l'information à la deuxième position du registre, tandis que la première position reçoit une nouvelle information. Au bout de "n" impulsions d'horloge, une information a donc été décalée de "n" positions dans un registre à décalage.

Les registres de ce type peuvent être conçus avec tous les types de bascules. Le 74LS74, par

exemple, est constitué d'un registre à décalage de 4 bits conçu avec des bascules D. L'impulsion de synchronisation est conduite en principe à toutes les bascules de manière synchrone. Le registre à décalage est décrit après quatre impulsions d'horloge. L'information d'entrée originelle est alors à disposition en sortie et se trouve expulsée en série à chaque nouvelle impulsion d'horloge.

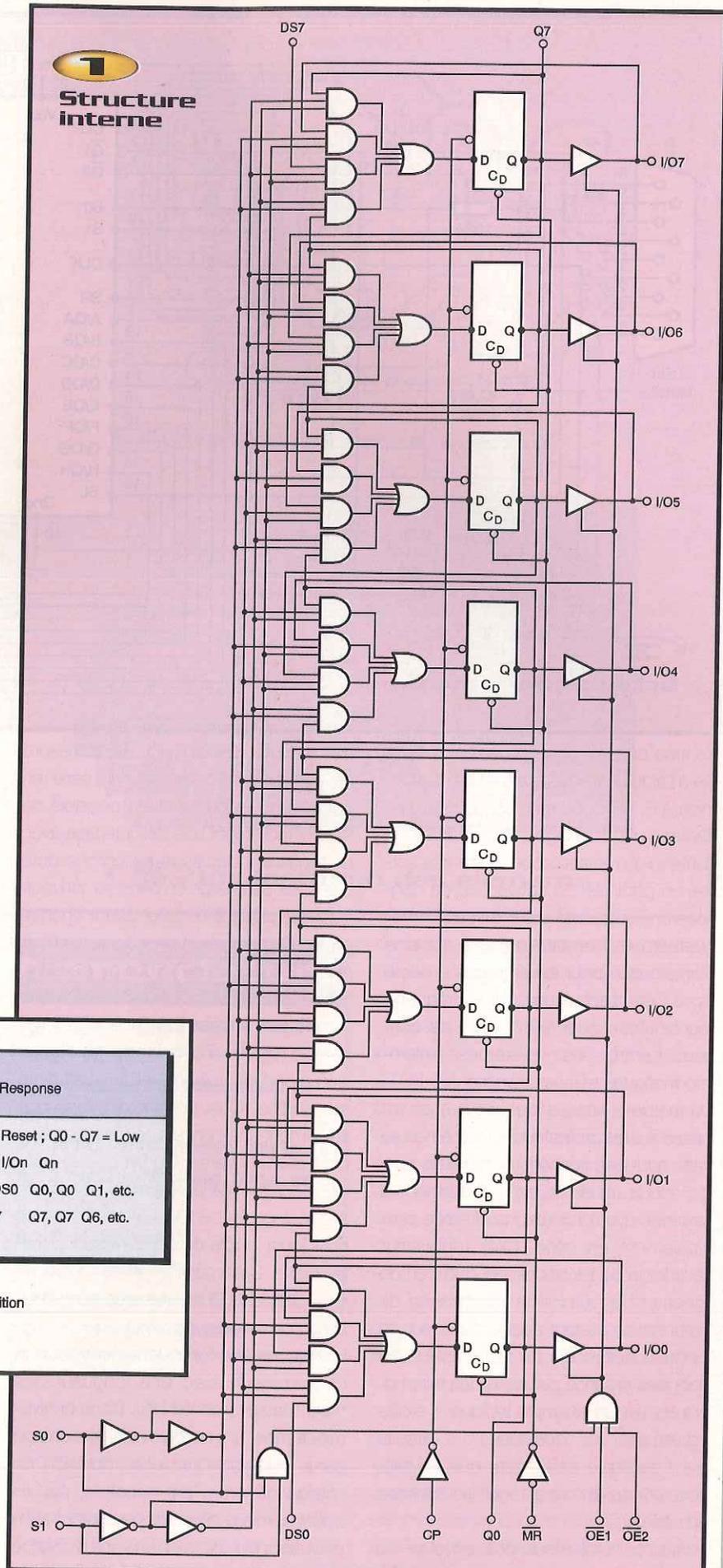
Grâce à la structure interne simple, il est très facile de concevoir des registres à décalage plus importants. La synchronisation en parallèle écarte les inconvénients des temps de réponse longs. Étant donné que ni la relation logique servant à la formation de la retenue, ni les temps de retard ne résultent d'une logique de commande, la vitesse de décalage maximale ne dépend que des temps de commutation des bascules utilisées. Il s'agit du temps de retard  $T_{pd}$  (CLK->Q) de l'entrée D à la sortie Q, du temps  $T_{su}$  (temps d'anticipation) qui doit s'appliquer au signal d'entrée au moins avant le front

d'horloge suivant et du temps  $T_h$  (temps de maintien) qui doit encore s'appliquer au signal d'entrée après l'impulsion d'horloge, afin de garantir un changement d'état sans incident. Le temps de maintien étant, en règle générale, plus court que le temps de retard de commutation  $T_{pd}$  (CLK->Q), le temps de retard  $T_{pd}$  (CLK->Q) et le temps d'anticipation  $T_{su}$  sont déterminant pour le calcul de la vitesse de décalage maximale. Par conséquent, une chaîne de décalage conçue avec le bloc fonctionnel a un temps de retard maximal :

$T_{pq \text{ max.}} = T_{pd} \text{ (CLK->Q)} + T_{su}$ . Il en découle une fréquence d'horloge maximale de l'inverse de  $T_{pd \text{ max.}}$  et est indépendante de la taille du registre à décalage. Mais la proposition énoncée ci-dessus a des limites étant donné que, sur les très longues chaînes de décalage, le temps de passage des signaux dans les lignes de données et d'horloge doit être pris en compte dans le calcul. Cela est notamment le cas lorsque certaines parties du registre sont commandées par des sources d'horloge différentes, ou bien se trouvent très éloignées les unes des autres, de sorte que des différences de temps de passage influent sur les lignes de signaux.

Le registre à décalage présenté ci-dessus ne possède qu'une entrée de données en série et une sortie de données en série. Mais dans la plupart des cas, il s'avère également nécessaire de pouvoir inscrire

**1**  
Structure interne



Inputs				Response
MR	S1	S0	CP	
L	X	X	X	Asynchronous Reset ; Q0 - Q7 = Low
H	H	H	⌊	Parallel Load ; I/On Qn
H	L	H	⌊	Shift Rights ; DS0 Q0, Q0 Q1, etc.
H	H	L	⌊	Shift Left ; DS7 Q7, Q7 Q6, etc.
H	L	L	X	Hold

H = HIGH Voltage Level x = Immaterial  
L = LOW Voltage Level ⌊ = LOW - to - HIGH Transition

**2** Table de vérité

une information en parallèle dans ce type de registre et/ou de pouvoir la sélectionner. Un tel registre à décalage peut être chargé en parallèle de manière asynchrone (c'est-à-dire indépendamment de l'impulsion d'horloge) par des entrées indépendantes les unes des autres.

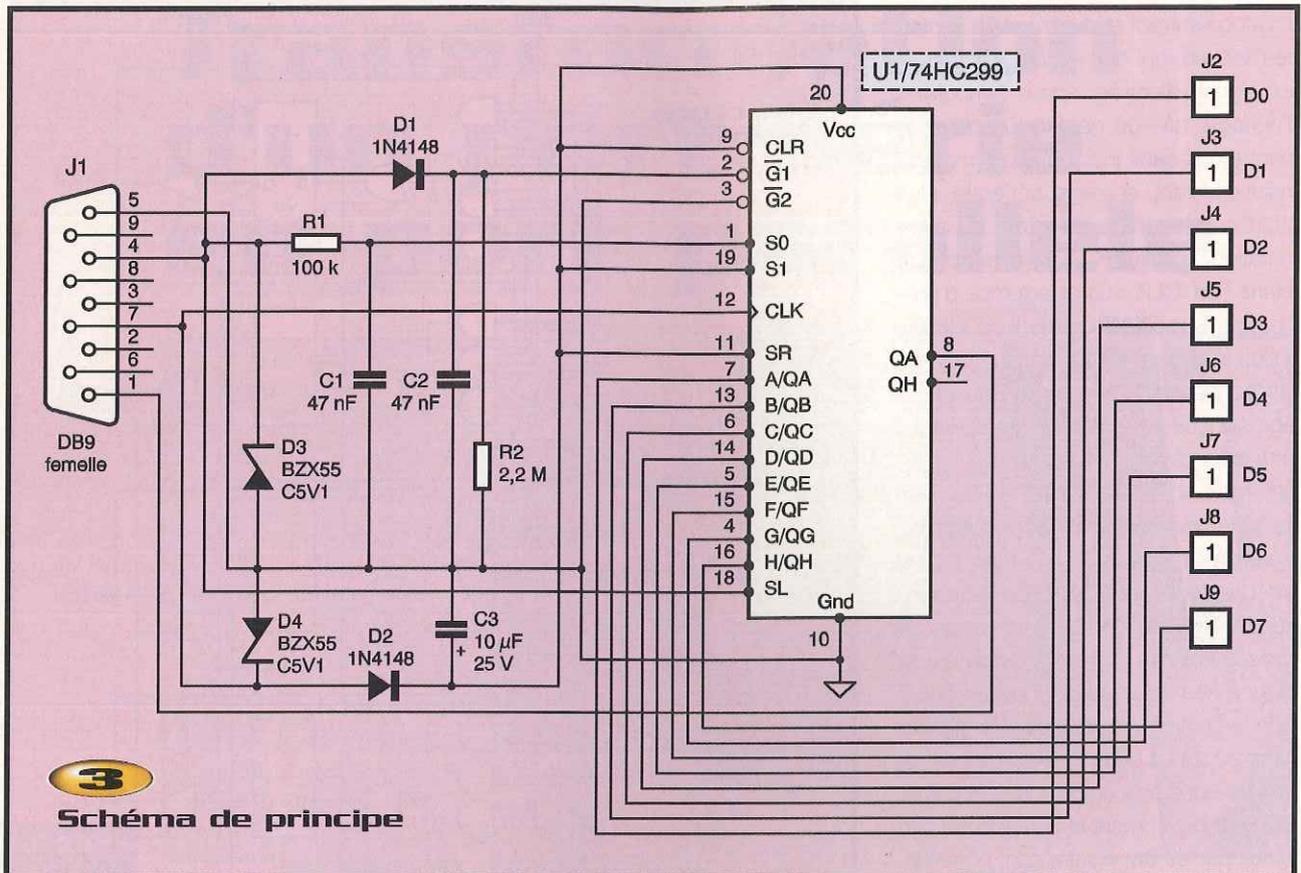


Schéma de principe

L'entrée parallèle asynchrone est exécutée à l'aide des entrées de mise à 1 et de mise à 0, qui sont, elles aussi, indépendantes de l'horloge. De même, le contenu du registre apparaît sur des sorties en parallèle. La conversion de données en série en données en parallèle, et inversement, constitue un vaste domaine d'application pour les registres à décalage. Cette technique est très fréquemment utilisée pour le transport de données entre les systèmes micro-informatiques et les périphériques.

Du fait de la vitesse plus élevée, on est passé sur les ordinateurs eux-mêmes au traitement des données en parallèle, tandis que la transmission des données à l'extérieur de l'ordinateur s'effectue principalement en série. Cela représente l'avantage de ne nécessiter que peu de lignes et de fournir une transmission de l'information restant économique sur de longues distances. La transmission de données en série par les lignes téléphoniques est un exemple typique. Il existe également quelques circuits d'entrée/sortie lents, tels que le téléscripteur qui envoie et reçoit les données en série.

Lorsqu'un ordinateur doit envoyer ou recevoir des données en série, il a alors

besoin d'une interface qui soit en mesure de convertir des données en série en données en parallèle. Il est possible de simuler facilement une telle interface avec un registre à décalage synchrone doté d'entrées de charge en parallèle. Un multiplexeur placé à l'entrée, avant chaque entrée D, permet un changement d'état entre décalage série et charge parallèle. Sur la charge en parallèle, l'information fournie aux entrées est prise en charge dans le registre à décalage avec l'impulsion d'horloge suivante et apparaît alors aux sorties. Cela permet d'obtenir non seulement une conversion parallèle/série mais aussi l'inverse. Du fait de sa capacité pour l'entrée en parallèle, le registre à décalage 74LS165 est tout désigné pour jouer le rôle de convertisseur parallèle/série. Ce circuit intégré est constitué de 8 bascules D montées en série avec synchronisation synchrone.

Expliquons son fonctionnement pour un convertisseur avec une longueur des mots transmis de 16 bits. Dans un premier temps, un mot de 16 bits est chargé dans le registre par une impulsion de charge générée, par exemple, par un système micro-informatique. L'information peut alors être décalée en série à chaque impulsion. Au bout de 15 impulsions de

décalage (on compte l'action de charge après 16), le dernier bit se trouve sur la dernière sortie. Avec l'impulsion de commande suivante (impulsion de charge), le deuxième mot peut alors être pris en charge puis décalé en série et ainsi de suite pour le troisième, le quatrième, etc. Le circuit 74LS164 permet de réaliser des convertisseurs parallèle/série. Ce registre à décalage à 8 bits est doté de deux entrées en série reliées par une fonction logique ET et d'une entrée de remise à zéro grâce à laquelle tous les étages peuvent être remis en position initiale de manière asynchrone. L'association de deux de ces composants constitue un convertisseur série/parallèle 16 bits.

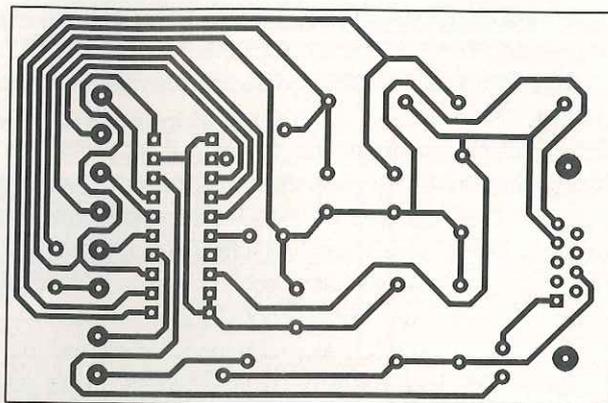
Dans un premier temps, le registre est réglé à l'aide d'une impulsion de remise à zéro pour un état initial défini (toutes les sorties sont au niveau logique zéro). Les 16 premiers bits du flux de données arrivant en série dans le registre peuvent alors être mémorisés. Au bout de la seizième impulsion d'horloge, le registre est entièrement décrit et le premier mot de 16 bits peut être extrait en parallèle par les sorties. Le deuxième mot est mémorisé avec les 16 autres impulsions, puis le troisième, etc.

Sur les registres à décalage décrits jusqu'ici, il n'est possible de décaler une information que dans un seul sens d'une position du registre à la suivante. Les additionneurs, par exemple, nécessitent des registres dans lesquels les données doivent être décalées aussi bien vers la gauche que vers la droite. Il en va de même pour la multiplication/division binaire pour laquelle on a recours à la fonction de décalage de la gauche vers la droite. Il est facile de réaliser un registre à décalage de la gauche vers la droite à partir d'un registre à décalage à entrées de charge parallèles. Pour cela, on raccorde les entrées de charge parallèles à la sortie attenante de droite. On obtient alors, soit un décalage de données de droite à gauche, soit un décalage de données de gauche à droite par une seule entrée de sélection de la direction du décalage.

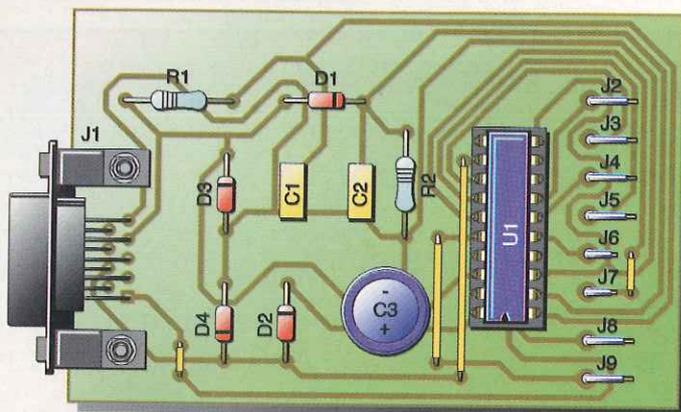
Outre la fonction de décalage de gauche à droite, sur les systèmes à déclenchement par horloge, il s'avère généralement avantageux que les fonctions de remise à zéro (tous les registres sont alors remis à zéro), charge parallèle et maintien de l'information (l'information est maintenue indépendamment du signal d'horloge et des entrées en parallèle) restent également disponibles. Le circuit 74LS299, qui est un registre à décalage 8 bits universel (qui est d'ailleurs utilisé dans notre application), fournit les cinq modes de fonctionnement suivants : remise à zéro asynchrone, maintien de l'information, décalage vers la droite, décalage vers la gauche, charge parallèle. Ce circuit intégré offre par conséquent de multiples possibilités d'utilisation y compris dans le domaine de l'informatique, comme par exemple les mémoires intermédiaires, les registres à accumulateur.

Grâce à une simple mise en cascade, il est possible de réaliser des chaînes de décalage de longueur voulue. Bien que le circuit soit monté dans un boîtier à 20 broches seulement, il délivre aussi bien la conversion parallèle/série que série/parallèle et, ce, grâce à une structure d'entrée/sortie multiplexée. Les quatre entrées de commande permettent d'ajuster les différents modes de fonctionnement.

Les systèmes logiques complexes nécessitent des générateurs d'impulsion



#### 4 Tracé du circuit imprimé



#### 5 Implantation des éléments

### Nomenclature

**U<sub>1</sub> : 74HC299 + support pour circuit intégré DIP 20**

**J<sub>1</sub> : connecteur DB9 femelle pour circuit imprimé**

**R<sub>1</sub> : 100 kΩ 1/4W (marron, noir, jaune)**

**R<sub>2</sub> : 2,2 MΩ 1/4W (rouge, rouge, vert)**

**C<sub>1</sub>, C<sub>2</sub> : 47 nF**

**C<sub>3</sub> : 10 µF/25V radial**

**D<sub>v</sub>, D<sub>2</sub> : 2 diodes 4148**

**D<sub>3</sub>, D<sub>4</sub> : 2 diodes zénors BZX55C5V1**

**J<sub>2</sub> à J<sub>9</sub> : connecteurs 1 point**

polyphasés pour la commande des fonctions isolées, très facilement réalisables avec des registres à décalage comme le 74LS95. Toutes les sorties de ce registre entrent dans une porte logique NON OU dont la sortie mène à l'entrée en série du registre. Tant qu'une sortie du registre est

de niveau logique haut, des signaux de niveau logique bas sont toujours mémorisés dans le registre. Au bout de quatre impulsions d'horloge, la porte change d'état, après quoi le potentiel 1 s'applique à l'entrée en série pendant une période d'horloge. Il en résulte qu'une seule sortie est toujours de niveau logique 1. Pour éviter les chevauchements de signaux ainsi générés, toutes les sorties sont synchronisées avec l'horloge par une porte logique ET.

En modifiant la condition de rétroaction du générateur d'impulsions triphasé décrit ci-dessus, il est très simple de réaliser des générateurs pseudo aléatoires. On peut, par exemple, concevoir un circuit sur lequel la rétroaction sur l'entrée en série a été effectuée par une porte logique OU exclusif. On obtient, par conséquent, un signal de sortie qui semble passer de manière aléatoire du niveau logique haut au niveau logique bas. Sur le circuit équivalent au chemin

de rétroaction, la période maximale (nombre de combinaisons de bits) s'élève à  $2^{(n-1)}$  impulsions d'horloge, dans laquelle "n" indique le nombre de bascules du registre à décalage.

La modification du chemin de rétroaction peut également générer deux autres longueurs de cycle. Des générateurs pseudo aléatoires de ce type sont utilisés partout où on a besoin (pour analyser, par exemple, le parcours de la transmission de données) d'une courbe statistique du signal, mais que l'on souhaite en revanche que le tracé du signal soit déterminable à l'avance. De plus, ces circuits sont utilisés comme générateurs de signaux parasites pour effectuer des

intégrés contient 8 bascules D qui sont déclenchables sur front et des étages logiques intermédiaires nécessaires pour effectuer les opérations synchrones de décalage à droite, de décalage à gauche, de chargement parallèle et de maintien. Le type d'opération est déterminé par S0 et S1 comme le montre la table de vérité à la **figure 2**. Toutes les sorties des bascules contiennent un étage à trois états pour séparer les broches d'entrée/sortie qui servent aussi comme données en entrée dans le mode de chargement parallèle. Q0 et Q7 sortent aussi sur d'autres broches pour étendre le décalage série de mots plus longs. Un niveau logique bas sur la broche /MR a la priorité

décalage, chargement et validation des sorties. Ces signaux sont l'horloge CLK, le contrôle du décalage/chargement S0, l'entrée série SL et la validation de la sortie G1. De plus, le port série doit aussi fournir une tension d'alimentation stable pour le 74HC299.

Parce que le port série ne possède que deux broches 4 et 7 directement contrôlables, quelques signaux doivent se partager une ligne de sortie. La tension d'alimentation et l'horloge partagent la broche 7, qui est normalement au niveau haut. Le signal d'horloge est généré par le programme avec un rapport cyclique élevé. À noter que les broches 4 et 7 sont toutes les deux mises à la tension 5,1V par les diodes D<sub>3</sub> et D<sub>4</sub>. Le composant est alimenté par la capacité C\*3 lorsque la broche 7 est au niveau bas. C<sub>3</sub> se recharge lorsque la broche 7 est au niveau haut. Les signaux S0, SL et G1 sont formés sur la broche 4. En mode de chargement, la broche 4 est au niveau haut de telle sorte qu'un signal d'horloge effectue le chargement des entrées D0 à D7 dans le composant. En mode de décalage, le rapport cyclique du signal sur la broche 4 est sélectionné pour garder S0 au niveau bas et G1 au niveau haut. Dans le mode de sortie, la broche 4 reste au niveau bas de telle sorte que les données dans le composant apparaissent sur D0 à D7.



**Le registre à décalage 8 bits universel**

recherches en technique de régulation. Grâce à son spectre de bruit étendu, on peut enfin utiliser ce circuit comme générateur de bruit.

Dans notre application de transformation du port série en port parallèle, nous utilisons un 74HC299 qui est un registre universel à décalage/stockage avec des broches communes d'entrée-sortie parallèle et avec des sorties à trois états. Les entrées de chargement parallèle et les bascules de sortie sont multiplexées afin de réduire le nombre total de broches pour ce composant. Des sorties supplémentaires sont fournies aux bascules Q0 et Q7 pour permettre une mise en cascade en série facile. Une broche de remise à zéro générale, active au niveau logique bas, est utilisée pour remettre à zéro le registre.

La structure interne du 74HC299 est représentée à la **figure 1**. Ce circuit

sur les entrées S0, S1 et CP, et remet à zéro toutes les bascules. Tous les autres changements d'états sont initialisés par un front montant de l'horloge. Des entrées peuvent changer lors du front montant de l'horloge que si elles répondent au temps d'établissement et de maintien. Un niveau logique haut sur soit /OE1 ou /OE2 dévalide les sorties trois états et mettent les broches d'entrée/sortie en haute impédance. Dans ces conditions, les opérations de décalage, de maintien, de chargement et de remise à zéro peuvent toujours se produire. Les sorties trois états sont aussi dévalidées par des niveaux logiques haut sur les deux entrées S0 et S1 en préparation à une opération de chargement parallèle. Le schéma de notre application est représenté à la **figure 3**. Le 74HC299 a donc besoin de quatre signaux de contrôle pour effectuer les fonctions de

## Réalisation pratique

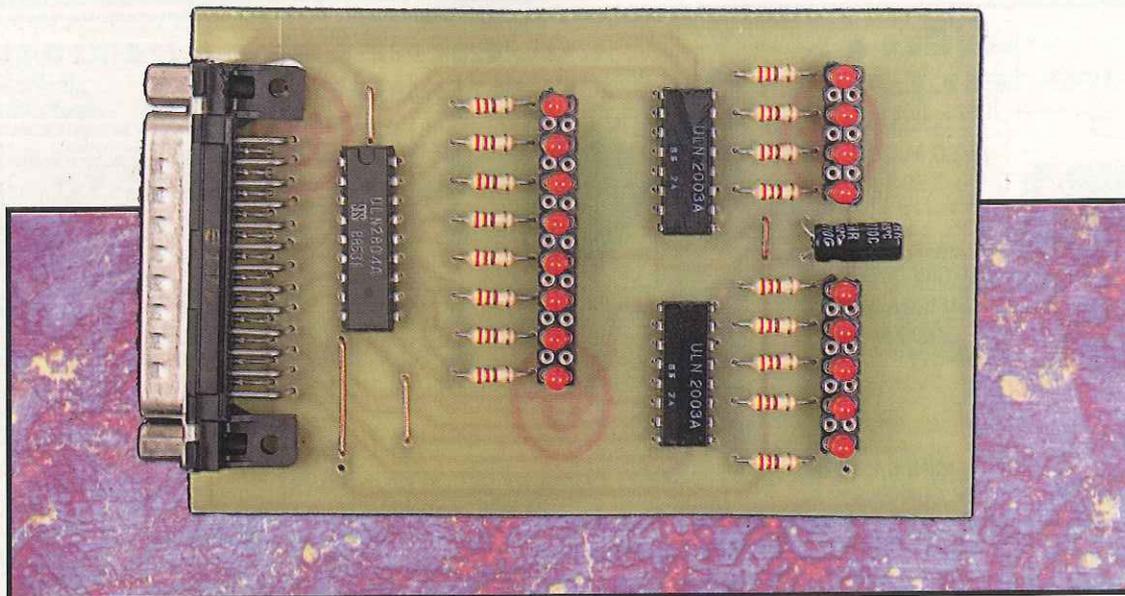
Le câblage de notre circuit ne pose aucune difficulté particulière : Il est bien sûr recommandé de mettre le 74HC299 sur support au cas où une mauvaise manipulation survient. La **figure 4** représente le circuit imprimé côté cuivre et la **figure 5** l'implantation des composants.

## Conclusion

Ce montage, qui transforme le port série en un port parallèle, peut être utilisé facilement lorsque l'on dispose d'un port série libre et que l'on a besoin d'un port parallèle supplémentaire sur un ordinateur de type PC. Le programme, écrit en langage C, contrôle le 74HC299 pour effectuer cette transformation.

**M. LAURY**

# Analyseur de port parallèle



Pour contrôler le fonctionnement d'un port parallèle de PC ou lors de la mise au point d'interfaces connectées à ce port, il est souvent utile de connaître l'état logique des différentes lignes des registres de données, de sorties et d'entrées.

Il n'est plus besoin de présenter l'intérêt du port parallèle d'un PC ou de tout autre micro-ordinateur. Les applications utilisant l'interface Centronics sont nombreuses et la souplesse de l'informatique permet à chacun de personnaliser, à son gré, l'utilisation des différentes cartes. Lors du développement et de la personnalisation d'un logiciel et même au moment des premiers essais d'un montage, il est intéressant de connaître l'état des entrées/sorties de cette interface parallèle.

Le montage proposé est prévu pour être raccordé directement à l'interface du micro-ordinateur par un câble parallèle ou pour être connecté en parallèle sur la liaison entre l'ordinateur et une interface déjà connectée au port parallèle. Pour ce dernier cas, la liaison est assurée par un câble en nappe de 25 conducteurs, sur lequel est sertie, à chaque extrémité, une fiche DB25 (SUB-D). Sur la longueur de ce câble est sertie un autre connecteur DB25 femelle sur lequel sera raccordé l'analyseur (figure 1). L'analyseur a été prévu pour que son alimentation soit prélevée sur la carte

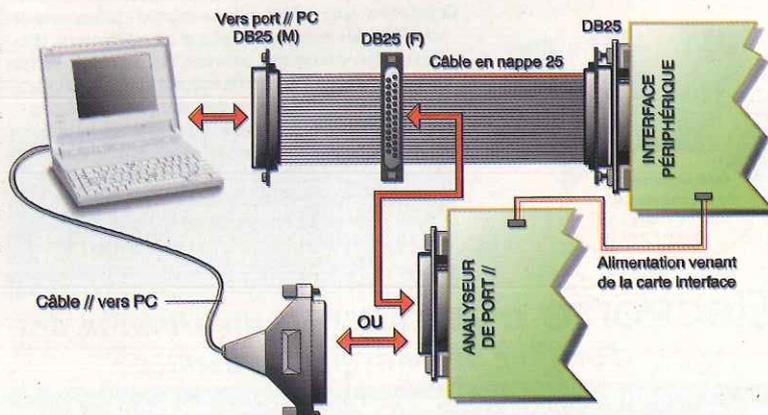
d'application ou qu'elle soit fournie par un adaptateur secteur.

## Le schéma

La figure 2 présente le schéma de principe du montage. La désignation et la fonction des broches du connecteur DB25 du port parallèle d'un PC sont données par le tableau de la figure 3. Chaque ligne de l'interface est associée à une DEL (Diode ElectroLuminescente)  $D_1$  à  $D_{17}$  qui indique l'état logique de la ligne. Si une DEL est allumée, la ligne est à l'état 1. Dans le cas contraire, la ligne sera à l'état 0. Le courant dans une DEL est limité par une résistance en série avec la diode ( $R_1$  à  $R_{17}$ ). La brillance de la DEL dépend de la valeur du courant qui la traverse et, par conséquent, de la résistance de limitation. Ce courant

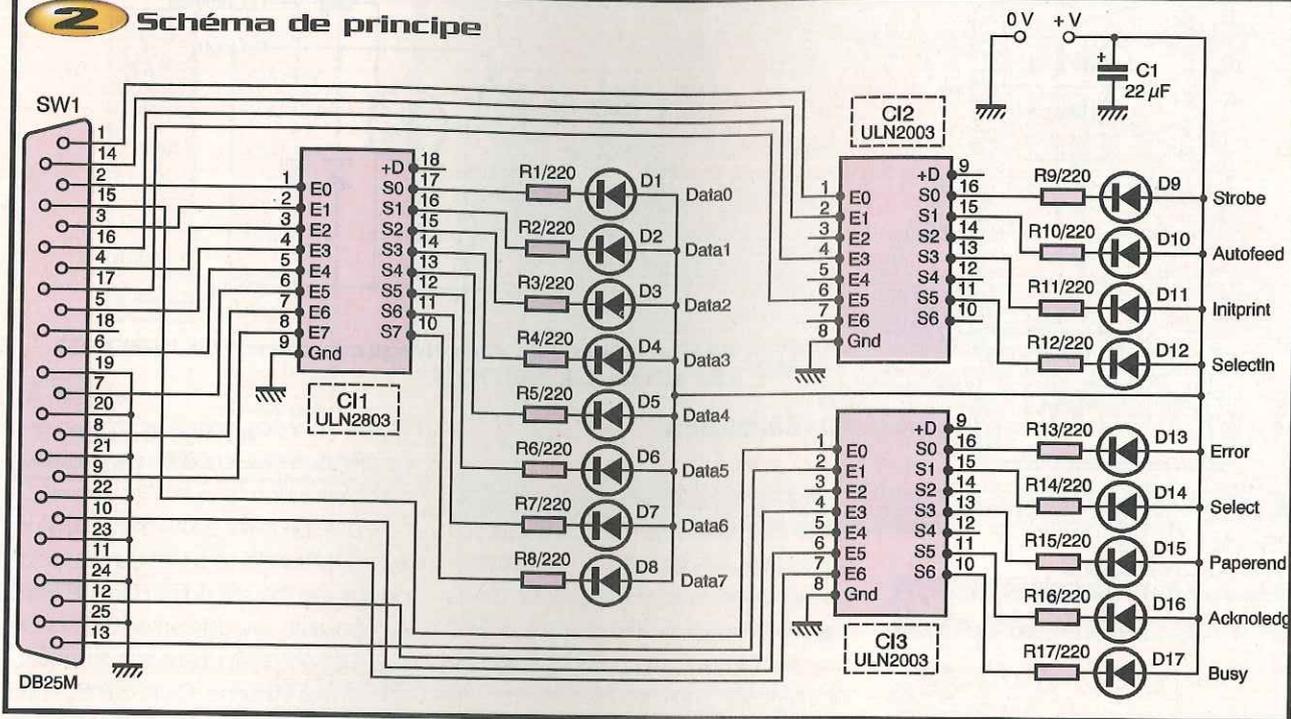
peut être compris entre 5 et 30mA pour une DEL standard. Avec un courant de 20mA, la DEL est déjà très lumineuse. Chaque DEL est commandée par un étage à collecteur ouvert, de sorte que le courant dans une DEL est donné par la relation :  $I_{DEL} = (+V - V_{DEL} - V_{sat}) / R$  où  $R$  est la résistance ( $R_1$  à  $R_{17}$ ),  $+V$  est la tension d'alimentation,  $V_{DEL}$  de l'ordre de 1,6 à 2,4V est la tension directe de la DEL, plus souvent 1,6V pour une DEL de couleur rouge,  $V_{sat}$  est la tension de saturation de l'étage de commande (environ 0,7V).

Trois circuits intégrés de technologie identique sont utilisés pour détecter l'état logique d'une ligne. Ces circuits intégrés,  $CI_1$ ,  $CI_2$  et  $CI_3$ , renferment un réseau de 8 ou 7 Darlington. Leur brochage est donné par la figure 4 et la figure 5 détaille la structure



Liaisons générales

**2** Schéma de principe



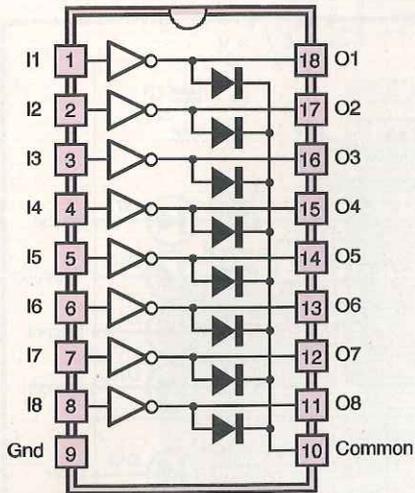
d'une voie. Si un état haut est appliqué sur une entrée, la base du Darlington de la voie considérée est suffisamment polarisée pour entraîner la saturation de l'étage Darlington et ainsi la polarisation de la DEL placée dans

le circuit collecteur de ce transistor. Par contre, avec un état bas en entrée ou si l'entrée n'est pas connectée, le Darlington est bloqué. Aucun courant ne circule dans le circuit de collecteur et la DEL est éteinte.

Une alimentation ordinaire de +5V fournira les quelques 200mA nécessaires à l'ensemble des DEL, pour un courant par DEL d'environ 10mA, soit une résistance de limitation de 220 Ω. Toutefois, puisque chaque Darlington

N°	Désignation	Nature	Poids	Observations
1	STROBE	sortie	1	out ad.out,1 => STROBE = 0
2	DATA 0	sortie	1	out ad.data,2 => DATA0 = 1
3	DATA 1	sortie	2	out ad.data,2 => DATA1 = 1
4	DATA 2	sortie	4	out ad.data,2 => DATA2 = 1
5	DATA 3	sortie	8	out ad.data,2 => DATA3 = 1
6	DATA 4	sortie	16	out ad.data,2 => DATA4 = 1
7	DATA 5	sortie	32	out ad.data,2 => DATA5 = 1
8	DATA 6	sortie	64	out ad.data,2 => DATA6 = 1
9	DATA 7	sortie	128	out ad.data,2 => DATA7 = 1
10	ACKNOWLEDGE	entrée	64	inp(ad.in) and 64 = 64 => ACKNOWLEDGE = 1
11	BUSY	entrée	128	inp(ad.in) and 128 = 0 => BUSY = 1
12	PAPER END	entrée	32	inp(ad.in) and 32 = 32 => PAPEREND = 1
13	SELECT	entrée	16	inp(ad.in) and 16 = 16 => SELECT = 1
14	AUTOFEED	sortie	2	out ad.out,2 => AUTOFEED = 0
15	ERROR	entrée	8	inp(ad.in) and 8 = 8 => ERROR = 1
16	INIT PRINT	sortie	4	out ad.out,4 => INITPRINT = 1
17	SELECT IN	Sortie	8	out ad.out,8 => SELECTIN = 0
18	GND	Autres remarques : ad.data est l'adresse du registre de données (888 ou 632 ou 956) ad.in = ad.data+1 est l'adresse du registre d'entrée ad.out = ad.data+2 est l'adresse du registre de sortie out ad.data,0 met à l'état bas toutes les lignes de données (DATA) out ad.out, 0 met à l'état haut les lignes STROBE, AUTOFEED et SELECTIN et place à l'état bas la ligne INITPRINT inp(ad.in) and 128 = 128 => BUSY = 0		
19	GND			
20	GND			
21	GND			
22	GND			
23	GND			
24	GND			
25	GND			

**3** Désignation des broches du connecteur DB25



**4**

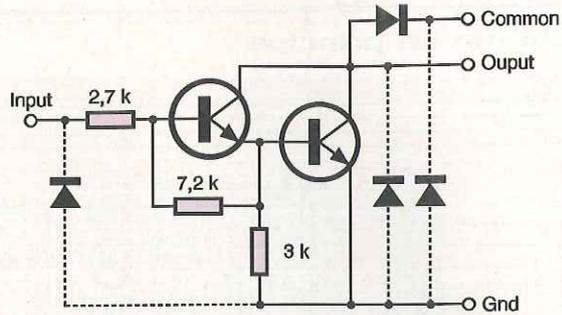
**Brochage du ULN2003 et du ULN2803**

présente un collecteur ouvert, une source de tension non régulée, égale ou supérieure à 5V, convient. Si une interface est déjà connectée au port parallèle, cette tension pourra être prélevée à l'entrée de son régulateur, plutôt que sur sa sortie, afin d'éviter un échauffement inutile du régulateur. Bien entendu, un adaptateur secteur peut être utilisé pour alimenter l'ensemble des DEL.

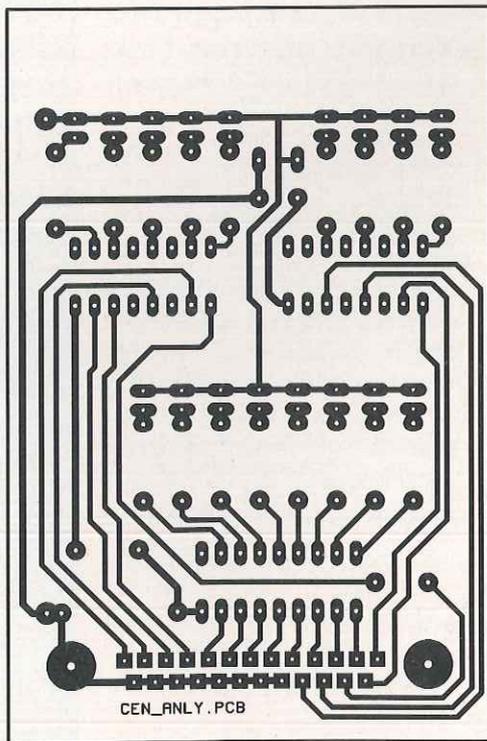
**5 Structure du Darlington d'un ULN2803 ou d'un ULN2003**

**La réalisation**

La **figure 6** présente le tracé des pistes du circuit imprimé dont la reproduction ne présente pas de difficulté. Vous débuterez l'implantation des composants de la **figure 7** par les quatre straps que vous pourrez réaliser avec des queues de résistances. Vous poursuivrez par les résistances et, ensuite, par les circuits intégrés, pour finir par les DEL. Ces dernières seront des modèles ronds de 3 mm de diamètre ou rectangulaires. Leur couleur pourra être choisie de manière à différencier les trois re-

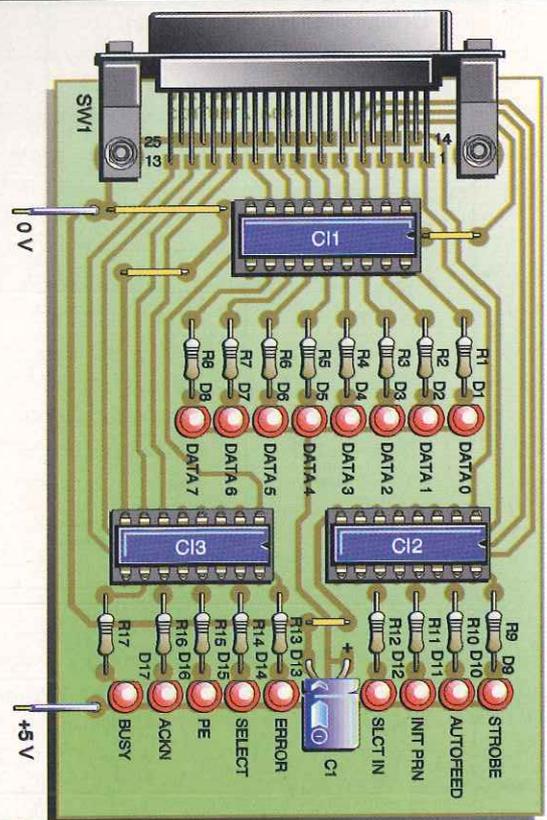


gistes (données, sorties et entrées). Le montage ne sera pas mis en boîtier, mais sera utilisé tel quel. Par sécurité, le côté piste peut être enduit d'une résine ou le circuit imprimé sera visé sur une petite plaque de bois (contre-plaqué). Côté composants, les différentes DEL seront repérées par une inscription au marqueur ou par une étiquette. Cette dernière sera réalisée à la main ou avec une imprimante, chaque inscription étant espacée d'un pas de 5,08. Il n'est alors pas nécessaire de recourir à un support auto-collant. Vous pourrez utiliser du papier ordinaire, collé sur le circuit imprimé avec



**6**

**Dessin du circuit imprimé**



**7**

**Implantation des composants**

Data0	Strobe
Data1	Autofeed
Data2	Init PRN
Data3	SLCT IN
Data4	
Data5	Error
Data6	Select
Data7	PE
	ACKN
	Busy



**Étiquettes de désignation des DEL**

de la colle liquide blanche. Vous pourrez également photocopier, puis découper la figure 8, reproduisant l'ensemble des désignations.

Pour les premiers essais, le montage sera directement relié au port parallèle de l'ordinateur. Avec quelques lignes de BASIC, vous entreprendrez alors la manipulation des différentes lignes. L'adresse de l'interface parallèle au sein du PC dépend de la fois du type de carte et du type de modèle de PC. Les trois adresses les plus com-



**mise en place des LED**

munes sont données dans le tableau de la figure 3, qui précise également l'instruction BASIC permettant d'activer ou de lire une ligne particulière.

Un programme "Testcen.exe" compilé complet et un second "programme.doc" pour une première approche sont disponibles sur le site de la revue.

**Nomenclature**

- R<sub>1</sub> à R<sub>17</sub> : 150 à 330 Ω (voir texte)
- C<sub>1</sub> : 22 µF/16V
- D<sub>1</sub> à D<sub>17</sub> : DEL rouges, vertes, jaunes rectangulaires ou de diamètre 3mm
- CI<sub>1</sub> : ULN2803
- CI<sub>2</sub>, CI<sub>3</sub> : ULN2003
- SW<sub>1</sub> : embase DB25 mâle
- 2 boulons M3

H. CADINOT

# SAINT-QUENTIN RADIO

## SPECIAL ALIMENTATIONS

**ALIMENTATIONS À DÉCOUPAGE**

**PSSMV1**

Adaptateur secteur 10 W à découpage. Sortie : 3 V - 4,5 V - 6 V - 7,5 V - 9 V - 12 V. Entrée 220 V ou 110 V. Livrée avec fiches standards. Prix : **145 F**



**PSSMV4**

Alimentation compacte à découpage 28 W. Sortie : 5 V - 6 V - 7,5 V - 9 V - 12 V - 15 V/ max 3,6 A. Entrée 100/240 V. 50/60 Hz 800 mA. Avec 8 fiches différentes. Prix : **359 F**

**PSSMV5**

Idem 12-15-18-20-22-24 V/max 2,3 A. Prix : **359 F**



**V924** Transformateur d'alimentation universel 9/12/15 VDC 1500 mA 22,5 VA - 18/20 VDC 1200 mA - 24 VA 24 VDC 1000 mA - 24 VA. Prix : **189 F**

**ALIMENTATIONS FIXES À DÉCOUPAGE 13,8 V**

**PSS1306**

Entrée 220 V 50 Hz tension 13,8 V - sortie 6 A (8 A en pointe) - poids 1,1 kg. Prix : **350 F**



**PSS1310**

Entrée 220 V 50 Hz tension 13,8 V - sortie 10 A (12 A en pointe) - poids 1,7 kg. Prix : **549 F**

**ALIMENTATIONS FIXES 13,8 V**

**PS1306**

Entrée 220 V 50 Hz tension 13,8 V - sortie 6 A (8 A en pointe) - poids 2,7 kg - ondulation 100 mV. Prix : **209 F**



**PS1310**

Entrée 220 V 50 Hz tension 13,8 V - sortie 10 A (12 A en pointe) - poids 4 kg - ondulation 100 mV. Prix : **329 F**

**PSS1320**

Entrée 220 V 50 Hz tension 13,8 V - sortie 20 A (22 A en pointe) - poids 3,5 kg. Prix : **810 F**



**PS1320**

Entrée 220 V 50 Hz tension 13,8 V - sortie 20 A (22 A en pointe) - poids 6,7 kg - ondulation 100 mV. Prix : **610 F**



**Avantages des alimentations à découpage : moins de composants de puissance, moins de chaleur - meilleure stabilité - moins de volume - moins de poids**

**CORDONS FIBRES OPTIQUES**

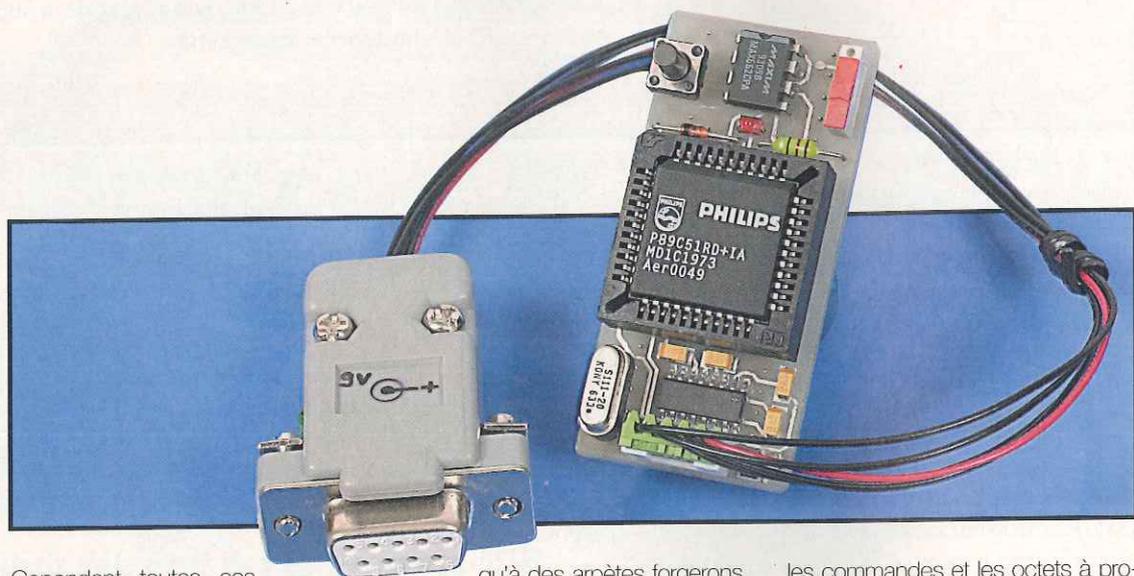
Faible déperdition mâle/mâle (Toslink/Toslink)  
1 m **85 F** 5 m **169 F** 10 m **249 F**

**EMBOÛT PROLONGATEUR DE CORDONS FIBRES OPTIQUES**

Permet d'additionner bout à bout différentes longueurs de cordons fibres optiques - femelle/femelle **19 F**

EXPEDITION COLISSIMO ENTREPRISE (\*) UNIQUEMENT : mini 100F de matériel Tarifs postaux Ile de France (75-77-78-91-92-93-94-95) : 0-250 g : 20 F ; 250g-2kg : 28 F ; 2kg-5kg : 48 F ; 5 kg - 10 kg : 58 F ; Autres dép. France Métropole : 0-250 g : 28 F ; 250g-2kg : 38 F ; 2kg-5kg : 58 F ; 5 kg-10 kg : 72 F. Paiement : chèque, mandat, carte bleue. DOM-TOM et étranger nous consulter. Horaires : du lundi au vendredi de 9 h 30 à 12 h 30 et de 14 h à 18 h 30. Le samedi de 9 h 30 à 12 h 30 et de 14 h à 17 h 30. (\*) équivaut à un recommandé

# KIT-51 : Module programmable à base de 89C51RD+



Le 8051 est un microcontrôleur tellement diversifié dans son espèce que les membres ne se comptent plus. Différents «fondeurs» (PHILIPS, ATMEL, DALLAS, etc.) ont su pérenniser le cœur originel tout en lui ajoutant de multiples périphériques innovants. Les capacités mémoires n'ont cessé de croître depuis l'emploi de mémoire «FLASH» qui autorise des méthodes de programmation «en circuit» par les voies de communication telles les bus RS232, SPI, CAN ou encore USB.

Cependant toutes ces performances n'ont pas suffi à cristalliser, au travers d'utilisations légitimes, les amateurs débutants. Le «bijou» proposé dans cette description aidera ceux qui, lassés de promener leur microcontrôleur entre programmeur, montage cible et «douche» à UV, veulent essayer le code fraîchement compilé sans jamais extraire de circuit de la maquette.

Cette dernière remarque prend toute son importance quand on sait que l'attention de l'opérateur est canalisée par le «pourquoi cette variable passe à zéro, ici ?» que par «pourquoi maintenant ce montage ne donne aucun signe de vie ?» Les interrogations se dissipent au moment où l'on constate l'inversion de polarité du pavé. Cela devait finir par arriver ! On ne peut pas prétendre que ce montage vous ruine en circuit imprimé. Votre bijou sera taillé dans une simple chute double face de 25,5x54 mm, à peine plus grande qu'un boîtier DIL 40 broches, sur laquelle pas moins de 189 points de connexion jouent des coudes. Cette réalisation s'adresse plus particulièrement à des orfèvres joailliers

qu'à des arpêtes forgerons. Le concept est de réunir dans un volume restreint quelque chose qui s'apparente à un microcontrôleur équipé, n'ayons pas peur d'effrayer les autres espèces, de 64 Ko de mémoire de programme FLASH, 1 Ko de RAM décomposé en 256 octets de IRAM et 768 octets de XRAM, un double DPTR, une horloge pouvant atteindre 33 MHz, voire même 40 MHz et 2 Ko EEPROM. Le challenge ne s'arrête pas là, incluons les composants permettant l'interface avec le PC pour la descente du code, l'électronique d'aiguillage, de programmation, un rien de boutons et inverseurs, et le dessin apparaît clairement. Un 89C51RD+ ou mieux, un 89C51RD2 de PHILIPS réunit à lui seul l'ensemble des caractéristiques évoquées. Le peu d'électronique associée vient compléter les caractéristiques au demeurant fort acceptables.

## Description (figure 1)

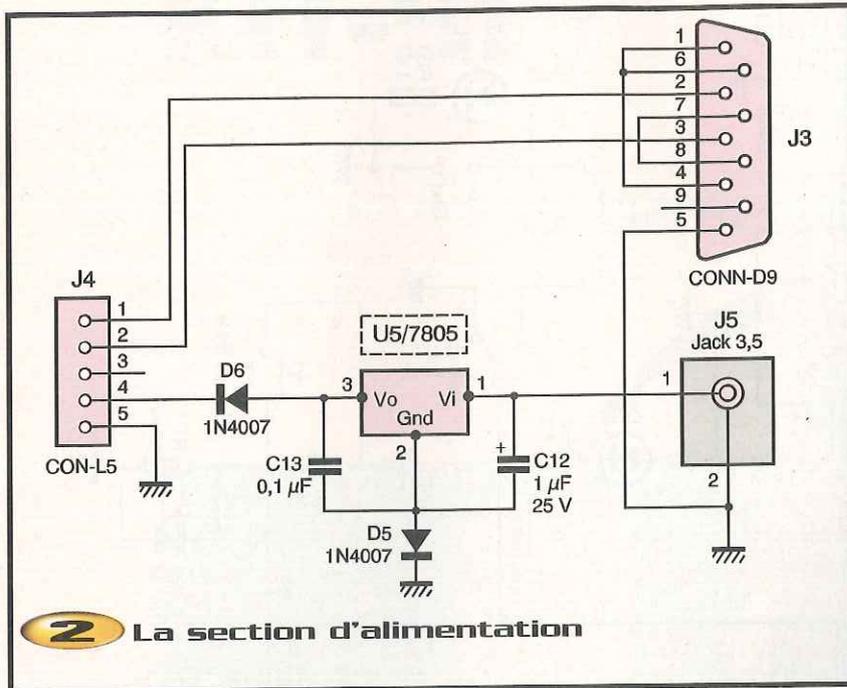
Le microcontrôleur inclut la capacité à être programmé en circuit (In System Programming). Pour cela les broches de port TX et RX véhiculent

les commandes et les octets à programmer tandis que d'autres broches reçoivent les tensions et signaux qui précipitent le circuit en mode programmation. Pour entrer dans le mode ISP, les broches suivantes doivent présenter des caractéristiques précises au front descendant appliqué sur RST :

- la broche PSEN est à l'état bas,
- la broche ALE est laissée non connectée ou soumise à un état logique haut,
- la broche EA est portée à un potentiel supérieur au potentiel représentant usuellement un 1 logique (12V en l'occurrence).

A partir de cet instant, si les registres STATUS REGISTER et BOOT VECTOR contiennent respectivement 00h et 0FCh, le PC du microcontrôleur branche à l'adresse 0FC00h. Cette adresse correspond à un petit programme appelé 'Boot loader' d'une taille de 1 Ko situé dans l'extrême haut de l'espace mémoire. Ces 1 Ko n'empiètent en rien sur l'espace de 64 Ko disponible pour l'utilisateur, ils occupent un espace mémoire parallèle. Le Boot loader comporte un set de routines élémentaires qui va





prendre en charge toutes les opérations de programmation, lecture, effacement...

Du côté PC, un petit programme sous Windows se charge de l'activation de chacune des options par des boutons aux significations instinctives. Rappelons que ce programme mis au point par PHILIPS est totalement gratuit et disponible sur le site Internet du fabricant précité.

Pour ceux qui désirent obtenir plus amples informations, la note d'application AN461 et la DataSheet en langue anglaise couvrent toutes les questions que l'on est en mesure de se poser, de plus le site PHILIPS regorge d'idées et trésors.

Après ces quelques éclaircissements, la tâche semble bien moins rude. Le micro-

contrôleur en boîtier PLCC 44 broches est logé dans un support afin d'aider à l'implantation. Chaque broche de port du boîtier PLCC est connectée à la broche de même nom d'un support ayant l'empreinte d'un 8051 en boîtier DIL40, cette fois. Quelques liaisons entre les deux supports sont, soit détournées, soit interrompues. C'est le cas pour l'alimentation, les broches de l'oscillateur, Rx, Tx, PSEN et EA. Passons en revue ces différents signaux afin de comprendre le schéma.

L'alimentation est prélevée directement sur la broche 40 du support DIL implanté sur le montage que vous voulez mettre au point. Elle peut aussi provenir du

connecteur J<sub>2</sub> lorsque vous voulez programmer un 89C51RD de manière autonome.

La diode D<sub>1</sub> protège votre bijou en cas d'inversion de polarité. Cette diode doit absolument être du type Schottky afin de limiter les pertes dues à la tension de seuil. Elle peut être remplacée par une self (cas de la maquette) ou encore par un strap isolé tout simplement.

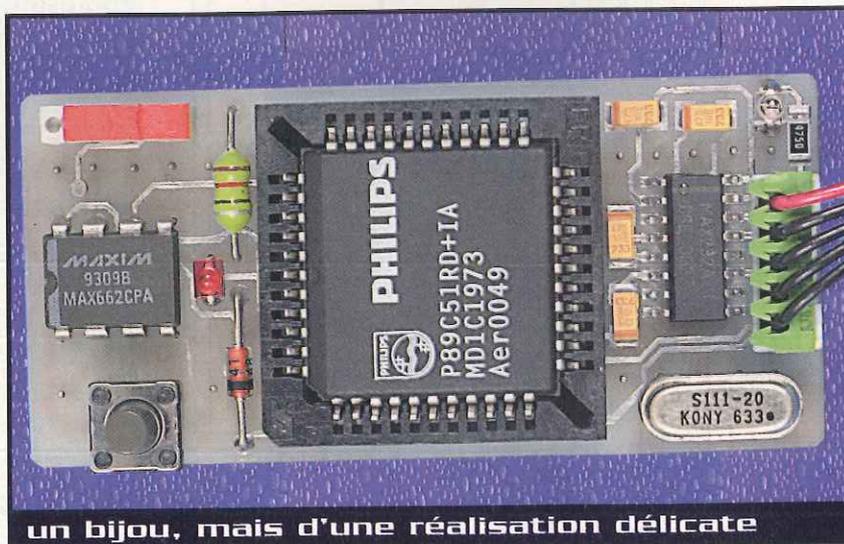
Les broches de l'oscillateur, X1 et X2, sont connectées à un support de quartz sur lequel chacun implantera au gré des besoins le quartz de son choix. Si le signal d'horloge provient du montage cible, ce dernier devra être appliqué sur la broche X1, le quartz sera retiré de son support, les condensateurs C<sub>1</sub> et C<sub>2</sub> ne provoqueront pas de gêne particulière. Les broches de port P3.0 et P3.1 aux fonctions doubles, interviennent aussi lors des communications série (Rx et Tx du RS232) en mode programmation (fil RUN/PROG = 0V).

La broche PSEN doit subir le même traitement, elle est reliée à la masse lors de la programmation ou au support lors de l'utilisation normale. Le circuit U<sub>3</sub>, triple commutateur analogique, résout efficacement ce problème. Il reçoit les trois signaux cités sur ses entrées X, Y, Z et les aiguille sur X0, Y0, Z0 ou sur X1, Y1, Z1 en fonction de la position de l'inverseur IT<sub>1</sub> reflétant l'intention de programmer (RUN/PROG = 0) ou d'essayer le soft (RUN/PROG = 1).

Reste la broche EA dont la fonction habituelle est d'autoriser l'exécution du code stocké en mémoire (ROM) externe quand cette dernière est soumise à un état bas. Ici, à l'inverse, elle sera soumise à un état logique haut afin d'exécuter le code contenu dans la mémoire FLASH interne. Lors de la programmation, cette broche sera portée à un potentiel de 12V.

Le circuit intégré U<sub>2</sub> (MAX662A) produit localement la tension de programmation 12V. Son fonctionnement est basé sur le principe de la pompe de charge très connu pour le MAX232 et qui a rendu populaire son fabricant.

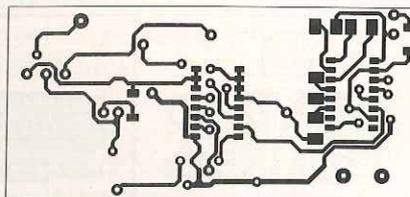
Le MAX662A nécessite deux condensateurs pour la pompe de charge et deux pour l'alimentation. La broche 8 (SHDN) permet à la tension de sortie (VOUT) de passer de 5V à 12V ±5% quand cette dernière passe respectivement de 1 à 0 et vice versa. On ne pouvait espérer mieux !



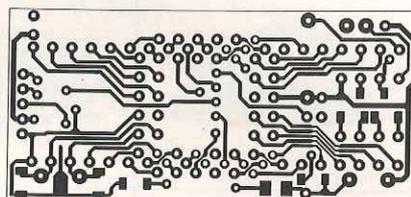
U<sub>1</sub> (MAX232), aidé des condensateurs C<sub>3</sub> à C<sub>6</sub>, s'intercale en médiateur entre le microcontrôleur et le micro ordinateur. La communication semble passer entre les mondes TTL et RS232. Il ne reste plus qu'à saupoudrer le tout de quelques interrupteurs et voyants avant de passer à la réalisation.

La diode LED D<sub>3</sub> visualise la présence de l'alimentation sur la maquette. Cette simple mesure permet bien souvent d'éviter d'extraire le module sous tension ou encore de passer cinq minutes à se demander pourquoi le programme ne semble pas fonctionner alors que l'alimentation générale est coupée.

L'inverseur IT<sub>1</sub> permet de passer du mode programmation (LED rouge D<sub>4</sub> allumée) au mode d'essai du programme et vice versa. Quant au petit bouton poussoir IT<sub>2</sub>, il permet d'effectuer, en plus de l'action automatique de la cellule R<sub>1</sub>-C<sub>10</sub>, un reset ponctuel au gré de la volonté du développeur ou systématiquement après avoir basculé IT<sub>1</sub>. Cette dernière contrainte est obligatoire sans quoi le 89C51RD+ ne prendra pas en compte le changement de mode.



**3** Tracé du circuit imprimé côté composants



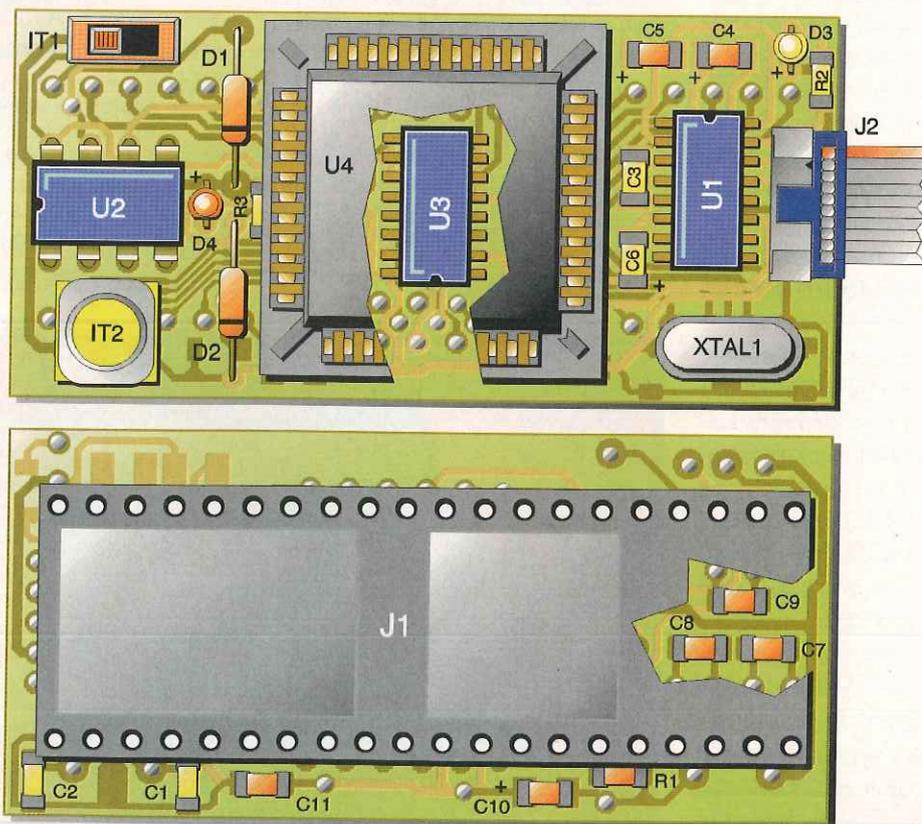
**4** Tracé du circuit imprimé

**Réalisation (figures 3, 4 et 5)**

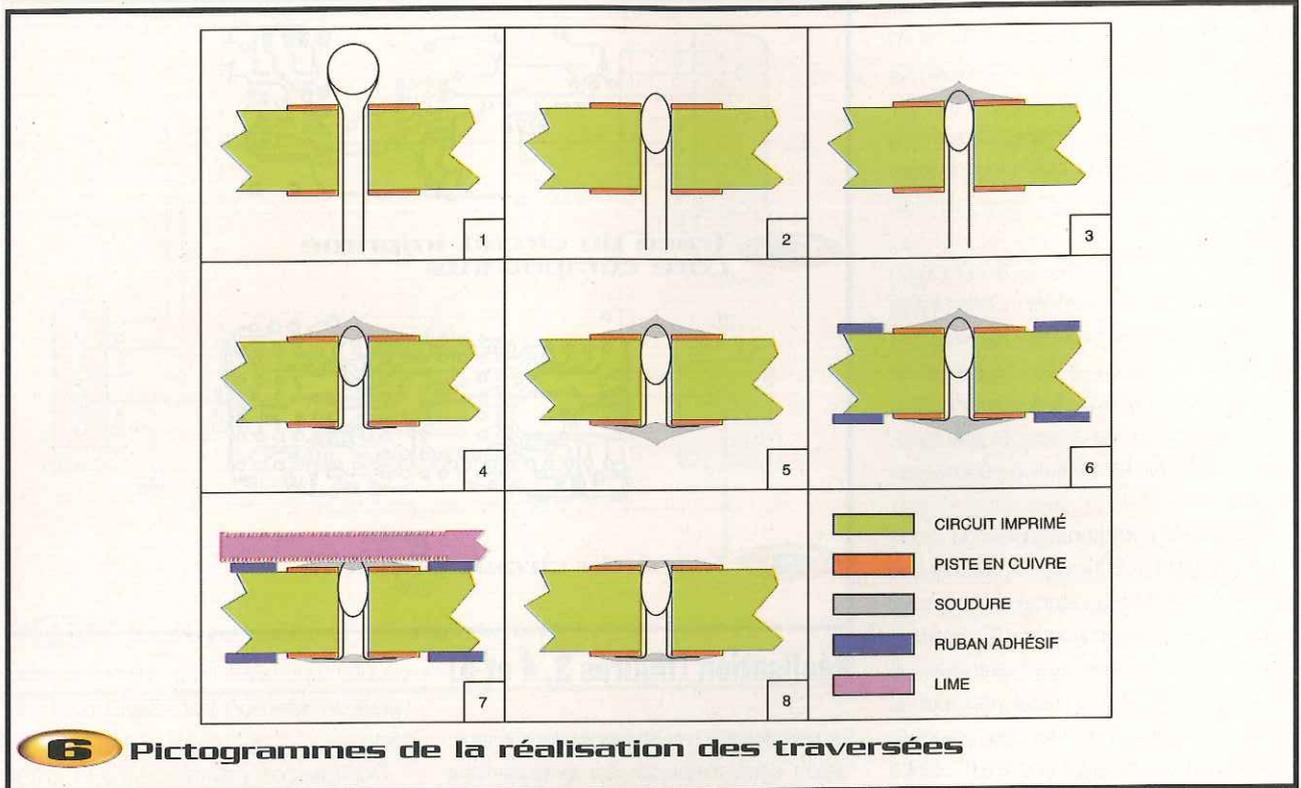
Ici, pas question de se lancer dans la réalisation sans respecter scrupuleusement l'ordre de soudage des composants sous peine d'aboutir à un échec. Après chaque étape, il faudra se discipliner et vérifier, soit

visuellement, soit à l'aide d'un ohmmètre, que la dernière opération effectuée est correcte.

- Commencez par reproduire le circuit imprimé double face, percez les trous au plus juste afin de faciliter l'étape de soudure. Percez l'ensemble des trous à un diamètre



**5** Implantation des éléments



de 0,5 mm y compris les traversées (via). Essayez de placer chaque composant (non CMS) et dans l'impossibilité, percez les trous au diamètre juste supérieur. Répétez l'opération jusqu'à l'engagement du composant aux plus grosses broches.

- Étamez le circuit, si possible à chaud (fer à souder) en vous aidant d'un flux décapant adapté. Certain préféreront la méthode chimique, le plus important n'est pas la méthode mais le résultat. Finissez par un nettoyage à l'aide d'un solvant de type acétone.

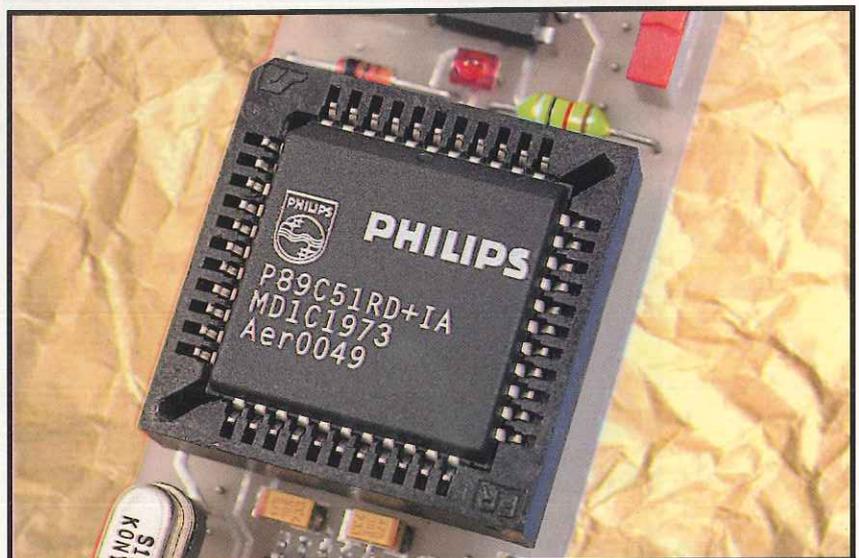
- Passez à la réalisation des 17 traversées. Le fil permettant de réaliser la métallisation entre les deux faces provient d'un conducteur multibrin étamé. Chaque brin d'une longueur de 5cm environ est enroulé sur une tige en métal (aiguille) d'un diamètre inférieur à 1mm. Une spire et demi est ainsi constituée. Les deux extrémités du fil sont introduites dans le trou à métalliser, puis par traction sur les deux brins, tentez de faire pénétrer la boucle dans l'épaisseur du circuit imprimé. Soudez le fil sur chacune des faces en veillant à former une soudure plate. Coupez l'excédant de fil (90% de la longueur initiale). L'opération est à renouveler seize fois encore mais vous serez soulagé de savoir que cette opération est la plus fastidieuse. Sous forme de picto-

grammes, la **figure 6** apporte un complément d'information sur le déroulement des opérations.

- A l'aide d'une lime plate extra fine (classique ou diamantée), arasez les dix sept soudures des traversées sur chaque face. Afin de ne pas rayer ou arracher les pistes, il est judicieux d'apposer une à deux couches de ruban adhésif sur la périphérie du circuit imprimé. En fin d'opération la lime viendra se poser sur le ruban garantissant l'épaisseur requise du 'via'.

- Nettoyez une fois de plus le circuit imprimé pour éliminer les traces de flux de soudure ainsi que les limailles produites par la dernière opération. A l'aide d'un ohmmètre, vérifiez la continuité entre chaque face normalement assurée par les traversées. A ce stade il est impératif de ne plus chauffer au fer les 'via' ainsi arasés qui ne tarderaient pas à attraper la grosse tête réduisant à néant vos efforts.

- Pour vous faire la main, soudez les résistances et condensateurs  $R_1, R_2, C_1, C_2, C_7,$



le microcontrôleur en boîtier 44 broches

C<sub>9</sub>. Nettoyez et contrôlez. Si tout se passe bien passez à la suite.

- Soudez U<sub>3</sub> dans le bon sens de préférence car toute opération de dessoudage peut être préjudiciable à la vie du dit composant. De plus une opération de sauvetage sous le support de U<sub>4</sub> due à une étourderie tardivement décelée est à proscrire. Héros s'abstenir ! Nettoyez et contrôlez ... (N.-C.)

- Sur le support DIL 40 broches, enfichez les deux barrettes sécables en insérant les broches de gros diamètre (l'inverse de ce qu'il conviendrait de faire) dans les contacts tulipes. Placer l'ensemble ainsi constitué (J<sub>1</sub>) sous le circuit imprimé et exécutez les

43 soudures (dont 3 sur la face supérieure). Les broches des barrettes sécables dépasseront tout au plus de 0,3 à 0,5 mm de la surface du circuit imprimé. (N.-C.)

- Implantez le support de U<sub>4</sub> (attention au sens) en le surélevant de manière à laisser un espace entre ce dernier et le circuit imprimé nécessaire à la réalisation de 3 soudures (broches 11, 22 et 44). Effectuez les 44 soudures de la face inférieure. (N.-C.)

- Implantez dans l'ordre R<sub>3</sub>, C<sub>10</sub>, C<sub>11</sub>, U<sub>1</sub>, C<sub>3</sub>, C<sub>6</sub>, C<sub>5</sub>, C<sub>4</sub>, D<sub>3</sub>, D<sub>2</sub>, D<sub>4</sub>, D<sub>1</sub> ou L<sub>1</sub> ou strap, U<sub>2</sub>, IT<sub>1</sub>, IT<sub>2</sub>, le support de XTAL<sub>1</sub> et enfin J<sub>2</sub>. (N.-C.)

- Confectionnez le câble de liaison RS232

comportant d'un côté le connecteur J<sub>2</sub> femelle et à son autre extrémité un connecteur DB9 femelle sous capot. Sur votre lan-cée, insérez sous le capot le régulateur et les composants connexes soudés en l'air. Vérifiez le bon fonctionnement de cette petite alimentation à l'aide d'un bloc secteur délivrant environ 9V. Passée cette étape, il ne vous reste plus qu'à insérer un quartz de 12 MHz (ou autre) ainsi qu'un 89C51RD+ dans les supports respectifs.

## Essais

Installez l'utilitaire WINISP disponible sur le site de PHILIPS à l'adresse :

<http://www.semiconductors.philips.com/mcu/download/80c51/flash/index.html>.

La documentation complète accompagnant ce produit permet à chacun d'utiliser le programmeur sans difficulté.

Le montage peut être alimenté de deux manières différentes, soit par le montage sur lequel on implante le module ou encore par le connecteur DB9 (à l'aide d'un bloc secteur débitant 9V). Après avoir lancé le programme d'application et rempli les différents champs de données, basculez l'interrupteur IT<sub>1</sub> en mode programmation, ce qui a pour conséquence visible l'allumage de la LED rouge D<sub>4</sub>, appuyez sur le PB de RESET et effectuez les commandes habituelles d'effacement et programmation.

Il est très important de remplir les champs 'Vector = FC' et 'Status = 0' lors de la première utilisation du module, d'activer la commande 'Write' du menu 'Misc' et ne plus jamais tenter d'y écrire autre chose sous peine de ne plus pouvoir communiquer avec WINISP. La seule solution serait de modifier le contenu de ces registres à l'aide d'un programmeur parallèle disposant de cette fonction. Après programmation, basculer IT<sub>1</sub> dans l'autre sens (LED D<sub>4</sub> éteinte), actionnez la commande de RESET et votre programme s'exécute. Le fichier TEST1.HEX fourni vous permettra de vérifier à l'aide d'une LED reliée en série à une résistance de 330 Ω accrochée au +5V que les 32 broches de port sont parfaitement fonctionnelles. Un clignotement basse fréquence témoignera du bon fonctionnement.

## Nomenclature

R<sub>1</sub> : 10 kΩ CMS boîtier 1206

R<sub>2</sub>, R<sub>3</sub> : 470 Ω CMS boîtier 1206

C<sub>1</sub>, C<sub>2</sub> : 27 pF CMS

C<sub>3</sub> à C<sub>6</sub>, C<sub>10</sub> : 1 µF Tantale 10V CMS

C<sub>7</sub>, C<sub>8</sub>, C<sub>11</sub> : 0.22 µF CMS

C<sub>9</sub> : 0,1 µF CMS

U<sub>1</sub> : MAX232 CMS

U<sub>2</sub> : MAX662A boîtier DIL (FARNELL)

U<sub>3</sub> : CD4053 CMS

U<sub>4</sub> : 89C51RD+IA boîtier PLCC 44 (FARNELL)

D<sub>1</sub> : 1N5819 ou self 10µH axiale ou strap

D<sub>2</sub> : 1N4148

D<sub>3</sub> : LED jaune 3 mm ou miniature

D<sub>4</sub> : LED rouge 3 mm ou miniature

IT<sub>1</sub> : inverseur DIL Dipswitch

(SELECTRONIC)

IT<sub>2</sub> : micro-poussoir 2 broches

J<sub>1</sub> : support DIL 40 contacts tulipe

J<sub>2</sub> : connecteur 5 points mâle au pas de 2 mm

X<sub>1</sub> : quartz bas profil 12MHz ou autre 2 barrettes sécables mâle/mâle 20 points

1 support PLCC 44

1 support de quartz (CONRAD)

### Pour alimentation

J<sub>3</sub> : SUBD9 femelle pour avec capot

J<sub>4</sub> : connecteur 5 points femelle au pas de 2 mm

J<sub>5</sub> : jack femelle mono miniature

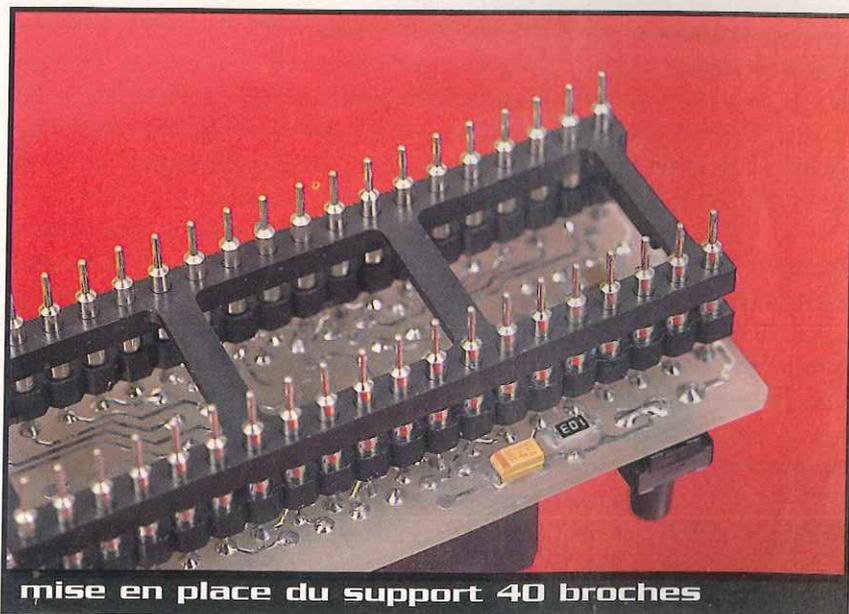
D<sub>5</sub>, D<sub>6</sub> : diodes 1N4007

C<sub>12</sub> : 1 µF 25V chimique

C<sub>13</sub> : 0,1 µF céramique

U<sub>5</sub> : 7805 (boîtier TO220 avec semelle coupée) ou 78L05

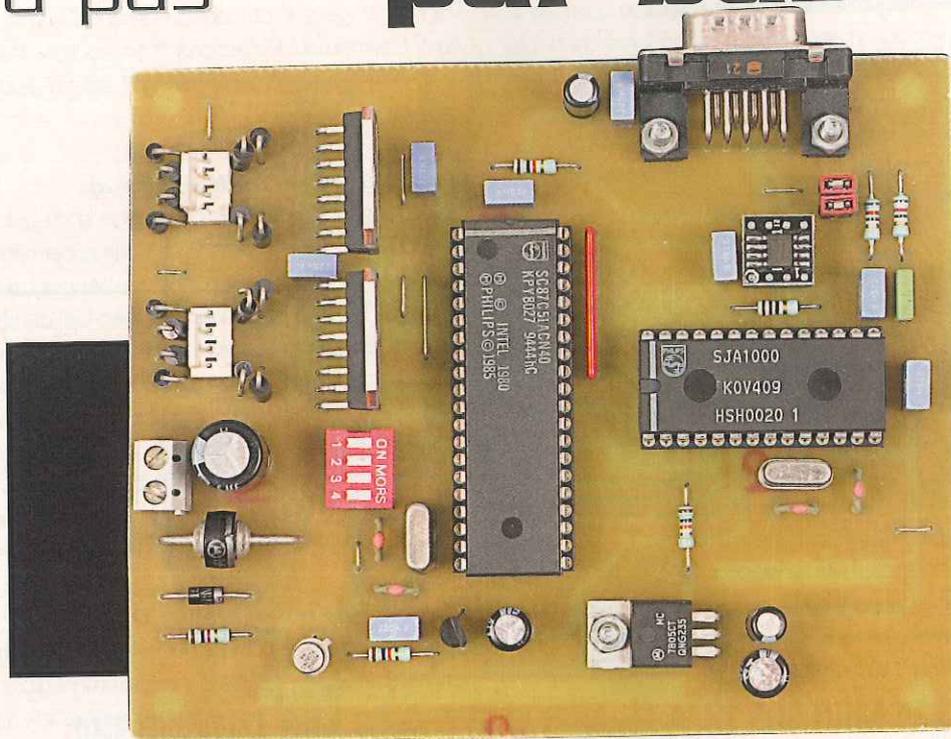
(régulateur 5V boîtier TO92)



mise en place du support 40 broches

J.-P. PITOLLET

# Commande de moteurs pas à pas par bus CAN



Pour continuer notre série de montages dédiés aux applications domotiques autour du bus CAN, nous vous proposons aujourd'hui de réaliser un module de commande intelligente pour moteurs pas à pas. Bien entendu, vous pourrez piloter ce montage à l'aide de l'interface RS232 / bus CAN que nous vous avons proposée dans ces pages.

Le montage que nous vous proposons ici est conçu pour piloter, de façon totalement indépendante, deux moteurs pas à pas bipolaires de petite et moyenne puissance.

Bien entendu, le montage gère complètement l'évolution des moteurs et propose des petites fonctions un peu inhabituelles et amusantes telle que la possibilité de faire sonner les moteurs pas à pas en guise de buzzer.

## Schéma

Les schémas de notre montage sont reproduits en figures 1 et 2. Comme cela apparaît sur la figure 1, le cœur de ce montage repose sur l'utilisation d'un microcontrôleur P89C51RD+ couplé avec un circuit SJA1000 (ou P82C200) spécialisé dans la gestion d'un bus CAN. La partie logique du montage se réduit donc à sa plus simple expression. Quant à la figure 2, elle dévoile les circuits de commande des moteurs pas à pas bipolaires.

Pour assurer l'accès aux registres du circuit SJA1000, l'espace mémoire externe adressé par le microcontrô-

leur est décodé de façon très sommaire. La logique du décodeur d'adresse est réduite à sa plus simple expression puisque nous avons simplement utilisé la ligne d'adresse A15 de  $U_4$  pour piloter la broche CS du circuit  $U_3$ . Notez, par ailleurs, que la broche 11 du circuit  $U_3$  impose le mode de fonctionnement des lignes ADO à AD7 du circuit  $U_3$ . Ici, c'est le mode "Intel" qui est utilisé. L'horloge interne du microcontrôleur est mise en œuvre au moyen du quartz  $QZ_2$  et des condensateurs  $C_8$  et  $C_9$  qui lui sont associés. Par souci de simplicité, nous avons préféré utiliser un quartz supplémentaire pour mettre en œuvre l'oscillateur du circuit  $U_3$ , plutôt que de partager le signal d'horloge.

Pour produire le signal de remise à zéro de ce montage, nous avons fait appel à un superviseur d'alimentation MC33164P ( $U_2$ ). Ce circuit permet de garantir le bon démarrage du montage même si les conditions de mise sous tension ne sont pas bien nettes (montée lente de la tension d'alimentation par exemple). Dans la mesure où les systèmes raccordés à un bus CAN peuvent être relativement éloi-

gnés, l'utilisation d'un superviseur

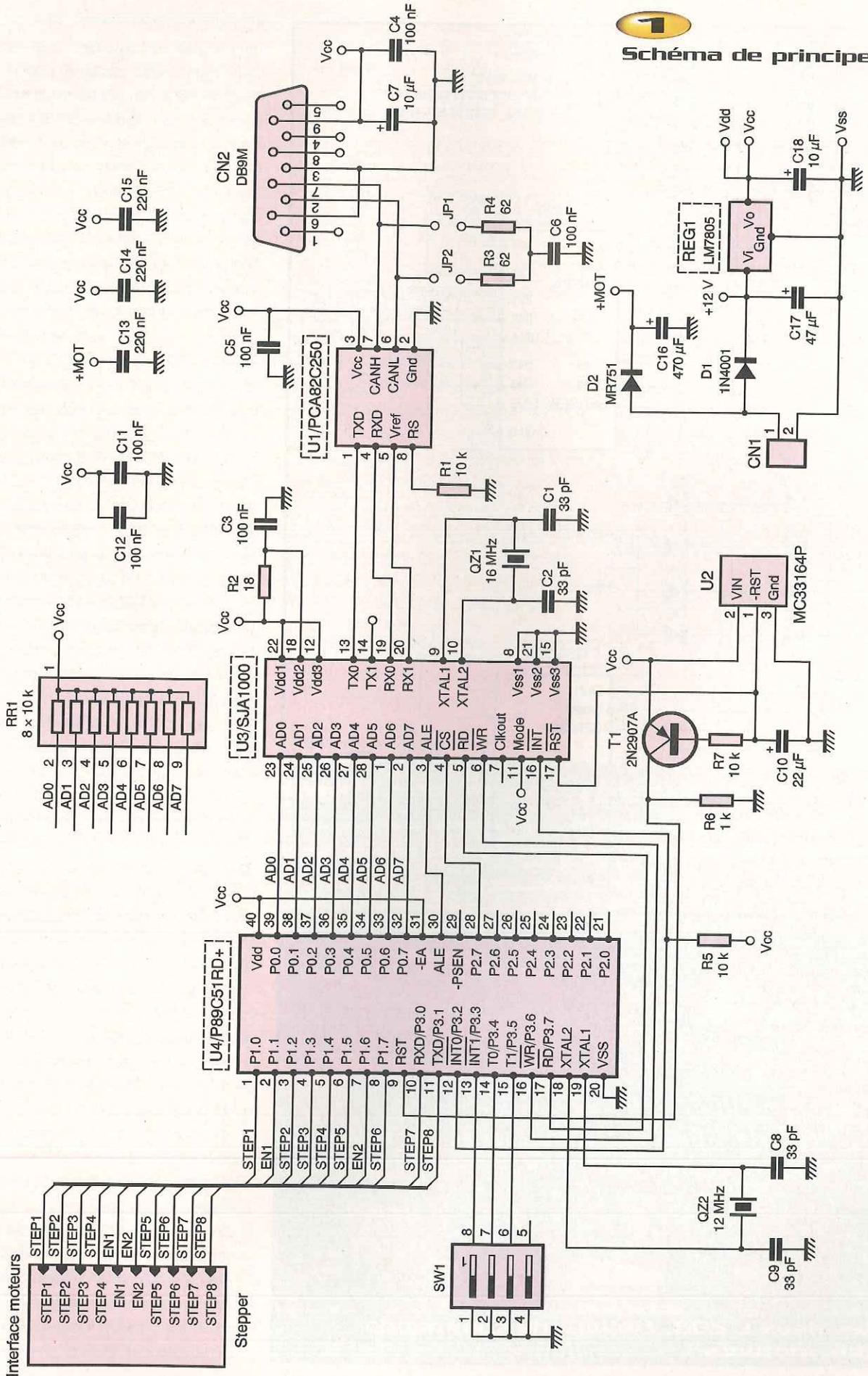
d'alimentation s'impose (avec la vitesse de bus retenue pour ce montage, il est théoriquement possible d'éloigner le montage à plus d'1km des autres équipements !). Dans un tel cas de figure, il n'est pas toujours simple de procéder à une remise à zéro manuelle du montage, si le microcontrôleur n'a pas correctement démarré à la mise sous tension (c'est ce qui arrive parfois lorsque l'on utilise un simple circuit R/C pour créer le signal RST).

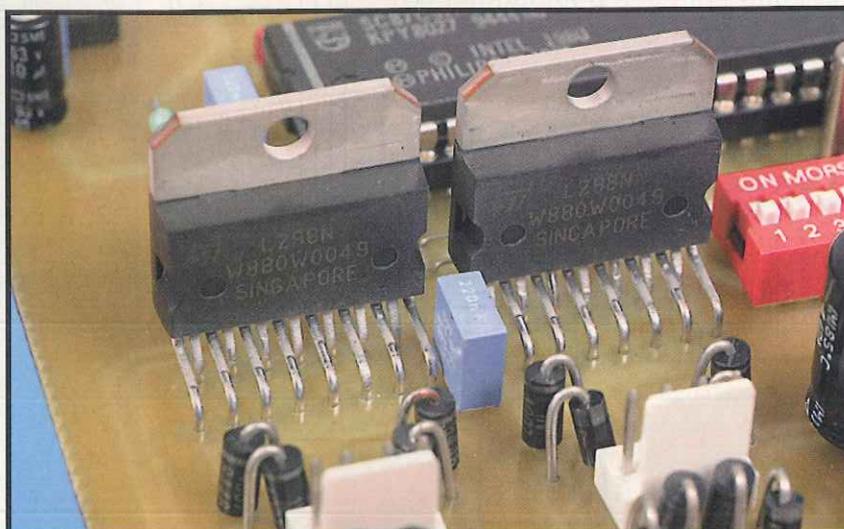
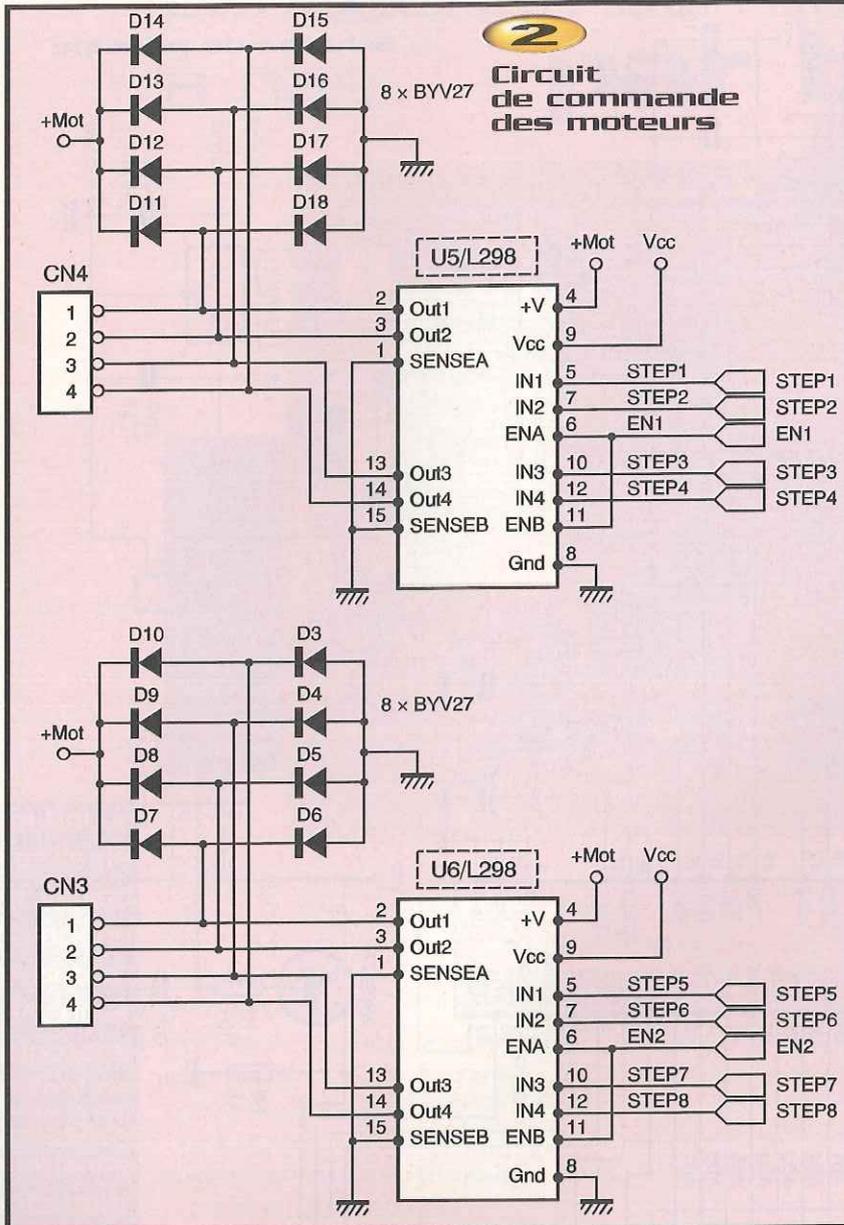
Sur notre schéma, le signal -RST issu du circuit  $U_2$  pilote directement la ligne de remise à zéro du circuit  $U_3$ , tandis que le transistor  $T_1$  inverse le niveau pour remettre à zéro le microcontrôleur. Le transistor  $T_1$  est nécessaire puisque l'état actif du signal de remise à zéro du microcontrôleur est un niveau haut.

Les signaux TX0, RX0 et RX1 du circuit SJA1000 sont confiés au circuit  $U_1$  qui se chargera d'adapter les niveaux afin d'être compatibles avec un bus CAN "High Speed". Ce mode correspond à celui mis en œuvre par l'interface RS232/Bus CAN que nous vous avons proposé il y a quelques



Schéma de principe





**Les deux L 298, circuits de commande des moteurs**

temps dans ces pages. Comme nous l'avons déjà expliqué, bien que le mode CAN "High Speed" soit prévu pour un dialogue à des vitesses élevées, rien n'empêche de l'utiliser à des vitesses lentes. Pour atteindre de longues distances de communication entre les équipements du bus CAN, il est parfaitement possible d'utiliser le mode "High Speed" avec un débit aussi bas que 20kbits/s. Nous avons fait ce choix car le circuit PCA82C250 est l'un des "drivers" de ligne le plus répandu actuellement pour le bus CAN. Notez qu'il est d'ailleurs possible de remplacer le PCA82C250 par un PCA82C251 qui est, lui, complètement compatible broches à broches. Malheureusement, ces composants sont plus répandus en version CMS qu'en version DIP (on ne trouve presque plus de version en boîtier DIP). Heureusement, il est possible de faire appel à des supports DIP pour composants CMS. Nous en reparlerons un peu plus loin.

En ce qui concerne la connectique retenue pour le bus CAN (CN<sub>2</sub>), vous noterez qu'elle a changé par rapport à l'interface RS232/Bus CAN que nous vous avons proposée dans ces pages. Nous avons décidé d'abandonner les connecteurs RJ11 au profit d'un connecteur SubD 9 points mâle, pour être compatible avec les recommandations du groupement CiA (Can in Automation). Ceci permet de raccorder directement le montage à une carte CAN du commerce sans avoir besoin de faire appel à un cordon d'adaptation (par ailleurs, les connecteurs à souder sur le circuit imprimé sont moins chers). Les lecteurs qui ont déjà réalisé l'interface RS232/Bus CAN que nous vous avons proposée dans ces pages devront dessouder les connecteurs RJ11 et relier des fils directement du PCB vers un connecteur SubD à souder sur fil. C'est un peu moins pratique, mais si le montage est installé dans un boîtier cela n'est pas trop gênant.

Les jumpers JP<sub>1</sub> et JP<sub>2</sub> permettent de raccorder la terminaison de ligne constituée de R<sub>3</sub>, R<sub>4</sub> et C<sub>6</sub> pour réaliser l'adaptation d'impédance nécessaire au bon fonctionnement du bus CAN. Notez un point important : seuls les équipements qui sont placés aux extrémités du bus CAN doivent être équipés d'une terminaison de ligne. Nous insisterons une nouvelle fois (au risque d'être jugé un peu lourd par certains) : Les équi-

pelements qui viennent se raccorder le long du bus ne doivent pas être équipés d'une terminaison de ligne.

Sachez que si vous raccordez des équipements sur un bus CAN sans terminaison de ligne, cela a de fortes chances de fonctionner à faible vitesse et sur de courtes distances. Mais il ne faudra pas vous étonner si les performances du bus sont complètement dégradées (taux d'erreur des transmissions important). Le moindre changement dans la topologie des faisceaux (ou bien l'ajout d'une nouvelle station sur le bus) pourra alors avoir de fâcheuses conséquences. Il vaut donc mieux prendre quelques instants pour vérifier que seuls les équipements en bout de lignes sont équipés d'une terminaison.

L'alimentation fournie au montage via CN<sub>1</sub> sert en même temps à alimenter la partie logique et les moteurs pas à pas. Le montage sera alimenté, de préférence, sous une tension de 12VDC, mais vous devrez peut être adapter cette tension à la tension nominale nécessaire pour les moteurs pas à pas. Attention, car le régulateur REG<sub>1</sub> requiert 7V à 8V minimums pour fonctionner. Vous ne pourrez donc pas commander des moteurs pas à pas 5V à moins de séparer les alimentations (connectez l'alimentation des moteurs à la place de la diode D<sub>2</sub>).

L'alimentation fournie au montage n'a pas besoin d'être stabilisée, mais elle devra être correctement filtrée. Les condensateurs C<sub>16</sub> et C<sub>17</sub> viennent renforcer le filtrage. Ne perdez pas de vue que les moteurs pas à pas consomment généralement des courants assez importants. Il faut donc que le bloc d'alimentation retenu pour ce montage soit capable de fournir le courant nécessaire sans que la tension s'écroule. A titre d'exemple, avec les petits moteurs 9V que l'auteur a utilisés, le courant consommé par le montage atteint facilement 600mA. Il ne faut donc pas lésiner sur le choix du bloc d'alimentation. Ajoutons que les diodes D<sub>1</sub> et D<sub>2</sub> protègent le montage contre les inversions de polarité, ce qui est parfois utile. Le schéma de la figure 2 dévoile les circuits de commande des moteurs pas à pas. Pour piloter un moteur pas à pas de type bipolaire, il faut faire appel à un double pont en H. Si on emploie des composants discrets pour réaliser cette fonction, il faut alors faire appel à 8 transistors de forte ou

moyenne puissance auxquels il faut ajouter de nombreux composants pour polariser correctement les transistors et réaliser l'adaptation de niveau des signaux de commande (généralement des signaux TTL). Pour commander des moteurs de puissance raisonnable, il est donc préférable de faire appel un circuit spécialisé. Pour ce montage, nous avons fait appel à deux circuits L298 qui intègrent les ponts en H ainsi que les étages pour une commande directe par un microcontrôleur. Les circuits L298 sont conçus pour fournir 2A ou 3A par enroulement, ce qui permet de piloter des moteurs relativement puissants. Malheureusement, la mise en œuvre du circuit L298, sur un substrat époxy simple face (35µm d'épaisseur de cuivre pour les circuits standards), ne permet pas de faire passer des courants aussi élevés.

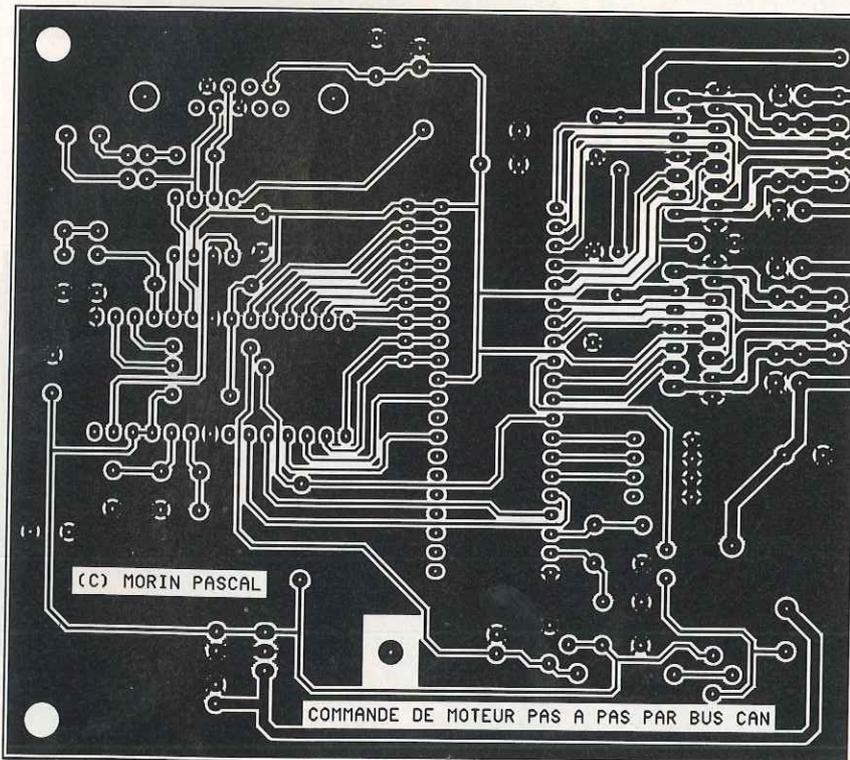
Pour utiliser toute la puissance que peuvent délivrer les circuits L298, les pistes d'alimentation (pistes reliant les broches +V ou GND) devraient pouvoir faire passer 4A à 6A selon l'état de commande des enroulements. Étant donné que notre montage est conçu pour piloter deux moteurs en même temps, cela impliquerait que le connecteur CN<sub>1</sub> fournisse des courants allant de 8A à 12A. Pour pouvoir passer des courants

aussi élevés, il faudrait des pistes de 3 à 4 mm de large. Or, si vous regardez bien le dessin du circuit imprimé reproduit en **figure 3**, vous pouvez apercevoir que certaines pistes sont obligées de passer entre les broches des circuits L298 (c'est le prix à payer pour conserver un circuit imprimé simple face). Cela limite beaucoup la largeur des pistes et, par la même, le courant qui peut y circuler.

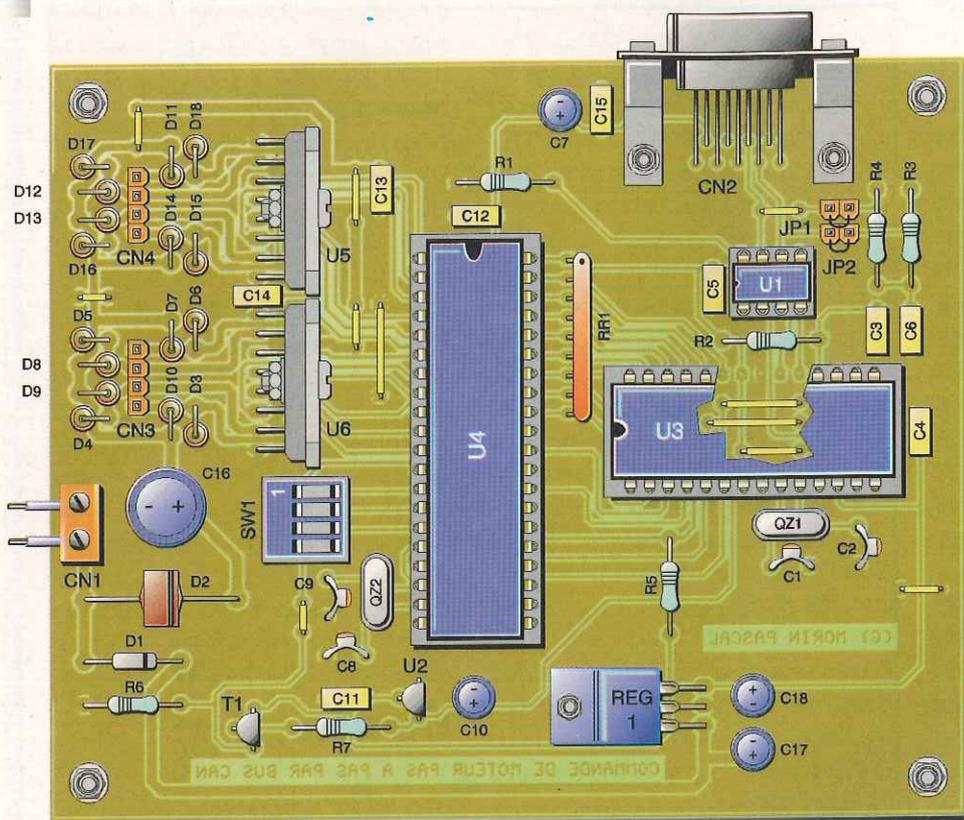
Il faudra donc utiliser ce montage pour piloter uniquement des petits moteurs pas à pas. Le courant n'étant pas limité, ce sera à vous de rester raisonnable. Sinon, vous ferez fumer les pistes du circuit imprimé (avouez que ce serait dommage). A titre d'exemple, nous avons fait nos tests avec des petits moteurs pas à pas bipolaires portant la référence KP4CM2-203. Ces moteurs fonctionnent sous 9V et sont diffusés par les annonceurs de la revue.

## Réalisation

Le dessin du circuit imprimé est visible en figure 3 tandis que la vue d'implantation associée est reproduite en **figure 4**. Les pastilles seront percées à l'aide d'un foret de 0,8 mm de diamètre, pour la plupart. En ce qui concerne CN<sub>1</sub>, CN<sub>3</sub>, CN<sub>4</sub>,



### 3 Tracé du circuit imprimé



#### 4 Implantation des éléments

REG<sub>1</sub>, D<sub>1</sub>, U<sub>5</sub> et U<sub>6</sub>, il faudra percer les pastilles avec un foret de 1mm de diamètre. Pour le connecteur SubD et le régulateur, n'oubliez pas de percer le passage nécessaire aux vis de fixations avec un foret de 3,5mm de diamètre. En ce qui concerne la diode D<sub>2</sub>, il faudra percer les pastilles avec un foret de 1,5mm de diamètre.

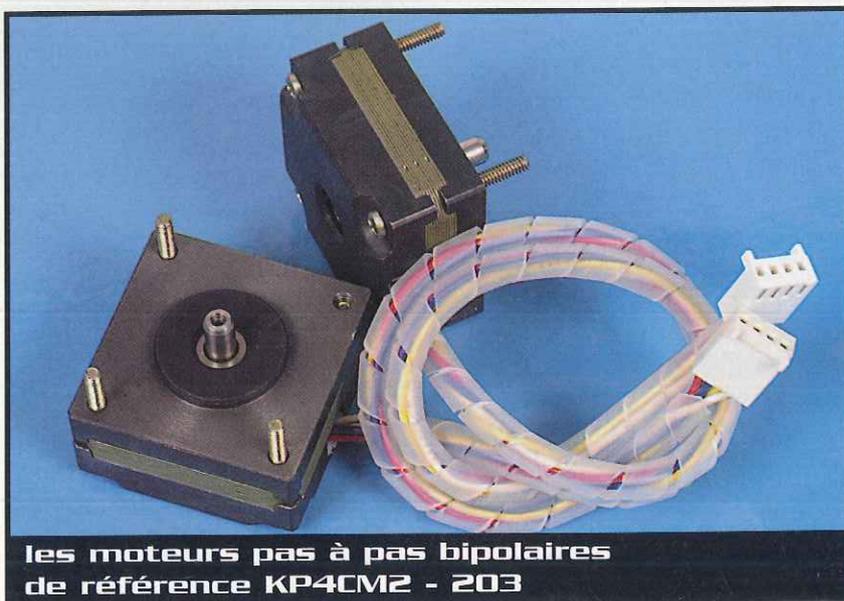
Avant de réaliser le circuit imprimé, il est préférable de vous procurer les composants pour vous assurer qu'ils s'implanteront correctement. Cette remarque concerne particulièrement le circuit PCA82C250 (ou PCA82C251). Ces composants sont très répandus en version CMS, mais ils ne sont pas toujours disponibles en boîtier DIP. Si vous avez du

mal à trouver un modèle DIP, vous pourrez utiliser le modèle CMS en faisant appel à un adaptateur pour boîtier SO8. Pour le reste, l'implantation des composants est relativement simple. Respectez scrupuleusement le découplage des lignes d'alimentations si vous voulez éviter les mauvaises surprises. Vous noterez la présence de 11 straps qu'il est préférable d'implanter en premier pour des raisons de commodité (cela concerne particulièrement les strap situés sous U<sub>3</sub>). Pensez à étamer abondamment les pistes dans lesquelles circulera un courant important, en fonction des moteurs que vous aurez choisis.

Le microcontrôleur U<sub>2</sub> sera programmé avec le contenu du fichier que vous pourrez vous procurer par téléchargement sur le serveur Internet de la revue (<http://www.eprat.com>). Le fichier "CANSTEP.BIN" est le reflet binaire du programme du micro-

contrôleur tandis que le fichier "CANS-TEP.HEX" correspond au format HEXA INTEL. Selon le modèle de programmeur dont vous disposez, vous utiliserez l'un ou l'autre des fichiers. Si vous voulez utiliser le programme Winstp de PHILIPS, pour programmer un microcontrôleur P89C51RD+ ou P89C51RD2, c'est le fichier CANS-TEP.HEX qui vous sera utile (voir l'article proposé par E. MIGOT dans le n°8 d'Interfaces PC, publié en juin 2001). Le fichier CANS-TEP.BIN pourra vous servir si vous préférez utiliser un microcontrôleur 87C51, car certains programmeurs pour cette famille de microcontrôleurs ne reconnaissent pas les fichiers Hexa (un comble pour ce type de produits). Si vous n'avez pas la possibilité de télécharger les fichiers, vous pourrez adresser une demande à la rédaction en joignant une disquette formatée accompagnée d'une enveloppe self-adressée convenablement affranchie (tenir compte du poids de la disquette).

Notez que le P89C51RD+ dispose de 64 k de mémoire flash, ce qui est beaucoup trop pour notre application (les 4 ko d'un simple 87C51 ne sont même pas remplis). Du coup, vous n'aurez que l'embaras du choix pour choisir votre microcontrôleur. A partir du moment où vous êtes équipés pour les programmer, les 87C51, 87C52,



les moteurs pas à pas bipolaires de référence KP4CM2 - 203

P89C51RC+ et P89C51RD+ conviennent tous pour notre montage. Le choix final n'est plus qu'une question de coût (en tenant compte du fait que les 87C51, 87C52 ne peuvent pas être reprogrammés s'ils sont en boîtier OTP)

A vrai dire, vous pouvez utiliser aussi les microcontrôleurs P89C51RB2, P89C51RC2 ou P89C51RD2 à condition de remplacer le quartz QZ<sub>1</sub> par un quartz de 6 MHz.

En effet, en dehors du fait que ces microcontrôleurs se programment sous 5V seulement (12V pour le P89C51RD+), ces microcontrôleurs sont cadencés par une horloge interne deux fois plus rapide. Pour pouvoir les utiliser avec un programme identique à celui conçu pour les microcontrôleurs habituels, il suffit donc de choisir un quartz dont la fréquence est diminuée de moitié. Notez qu'à terme les P89C51RC+ et P89C51RD+ ne seront plus disponibles puisqu'ils sont d'ores et déjà remplacés par les P89C51RB2, P89C51RC2 et P89C51RD2. Si vous avez le choix, et étant donnée la relative simplicité d'adaptation que cela demande, préférez l'utilisation des P89C51RB2.

Pour brancher les moteurs pas à pas au montage, commencez par repérer les fils de chacun des enroulements du moteur. Aidez-vous d'un ohmmètre pour repérer la résistance de chaque bobine qui est relativement faible. Une fois que vous avez identifié les fils, connectez le premier enroulement du moteur aux broches 1 et 2 (CN<sub>3</sub> pour le premier moteur et CN<sub>4</sub> pour le second) et le deuxième enroulement broches 3 et 4. Le sens de rotation des moteurs est arbitraire puisque cela dépend de l'ordre dans lequel vous connectez les fils. Si vous utilisez deux moteurs identiques, pensez à connecter les fils dans le même ordre.

Si vous utilisez des moteurs différents et que l'un d'eux tourne dans le sens opposé à ce que vous souhaitez, vous n'aurez qu'à intervertir les fils de l'un des enroulements. Ne changez pas les fils des deux enroulements en même temps sinon le sens de rotation du moteur sera identique.

Le montage est relativement simple à utiliser, en particulier avec l'interface RS232/Bus CAN publiée dans ces pages. La vitesse de transmission du montage est fixée par le logiciel du microcontrôleur à

SW1-1	SW1-2	SW1-3	SW1-4	Valeur de l'identificateur de base
ON	ON	ON	X	0F8H
OFF	ON	ON	X	1F8H
ON	OFF	ON	X	2F8H
OFF	OFF	ON	X	3F8H
ON	ON	OFF	X	4F8H
OFF	ON	OFF	X	5F8H
ON	OFF	OFF	X	6F8H
OFF	OFF	OFF	X	7F8H

**5** (Identificateur de base en fonction de SW1)

20 kbits/s afin de pouvoir dialoguer sur un bus CAN de plusieurs centaines de mètres de longueur. Cette valeur ne peut pas être modifiée pour ce montage, ce qui ne devrait pas être vraiment gênant dans le cadre d'une utilisation domotique, telle que nous l'avons conçue (peu de données à échanger).

Selon la position des interrupteurs de SW<sub>1</sub>, le montage répond à différents identificateurs sur le bus CAN. Le tableau de la **figure 5** indique les 8 possibilités offertes par SW<sub>1</sub>, tandis que le tableau de la **figure 6** indique quels sont les messages qui sont interprétés par le montage. Pour la progression des moteurs, le moteur tient à jour un compteur pour chaque moteur qui détermine le nombre de pas qu'il lui reste à faire. L'étendue des compteurs est de -32767 à + 32767, mais les commandes peuvent seulement demander l'addition de ±127 pas. Pour des déplacements plus importants, il faut alors émettre plusieurs messages suc-

cessifs sur le bus CAN. A l'aide d'un seul message, vous pouvez faire progresser les deux moteurs de façon synchrone. Mais vous pouvez aussi ne faire tourner qu'un seul moteur en spécifiant un déplacement de 0 pas supplémentaire pour le deuxième moteur.

Bien entendu, vous pouvez définir le temps qui s'écoule entre chaque pas (de 1 ms à 255 ms), pour régler la vitesse de progression des moteurs. Vous pouvez stopper la progression des moteurs de façon individuelle et choisir le mode de repos (couple bloqué ou rotation libre). Et enfin, comme nous l'avons indiqué en introduction, nous avons ajouté une fonction "buzzer" qui utilise l'inertie mécanique des moteurs pour qu'ils émettent un signal audible de 500 Hz sur demande (de façon indépendante). Cette fonction peut être utilisée à n'importe quel moment, sans rien changer à la position des moteurs ni au nombre de pas en cours. Le seul changement perceptible à l'issue de cette

Identificateur de la trame	Émission / Réception	Action du montage	Nb d'octets de donnée (DLC)
ID = Base	Réception	Progression des moteurs	2
ID = Base+1	Réception	Réglage de la vitesse des moteurs	2
ID = Base+2	Réception	Relâche / maintient moteur n°1	1
ID = Base+3	Réception	Relâche / maintient moteur n°2	1
ID = Base+4	Réception	"Buzzer" sur le moteur n°1	0
ID = Base+5	Réception	"Buzzer" sur le moteur n°2	0
ID = Base+	Réception (RTR) + Émission	Lecture des variables internes (status, nb de pas restants, vitesse)	7
ID = Base+7	Réception	Réinitialisation complète (arrêt des moteurs, valeurs par défaut)	0

**6** (Action en fonction de l'identificateur des messages CAN)

commande est que les moteurs restent en couple bloqué s'il n'y a plus de pas à faire. Si les moteurs étaient déjà placés en rota-

tion libre, il faudrait penser à lancer à nouveau la fonction après l'appel à la fonction "buzzer". Pour ne pas allonger inutilement

le texte de cet article, la signification exacte des octets contenus dans les messages interprétés par le montage est décrite dans un fichier qui vous sera remis avec les fichiers qui permettent de programmer le microcontrôleur.

Les lecteurs qui souhaiteraient modifier le comportement du logiciel (en particulier pour modifier la vitesse du bus ou bien la valeur de base des identificateurs) pourront retoucher les fichiers sources qui accompagnent ce montage. Les lecteurs intéressés pourront compiler le programme à l'aide du compilateur C développé par la société KEIL et qui est disponible en version de démonstration sur leur site Internet à l'adresse <http://www.keil.com>. La taille du code généré pour notre programme est légèrement en dessous des limitations de la version de démonstration du compilateur, ce qui est une véritable aubaine pour ceux qui souhaitent développer des petites applications autour du P89C51RB2.

P. MORIN

## Nomenclature

**CN<sub>1</sub>** : bornier de connexion à vis 2 plots, au pas de 5,08mm, à souder sur circuit imprimé, profil bas

**CN<sub>2</sub>** : connecteur Sub-D, 9 points, mâle, sorties coudées, à souder sur circuit imprimé (ref. HARTING 09 66 122 7801)

**CN<sub>3</sub>, CN<sub>4</sub>** : barrettes mini-KK 4 contacts, sorties droites, à souder sur circuit imprimé (ref. MOLEX 22-27-2041)

**C<sub>1</sub>, C<sub>2</sub>, C<sub>8</sub>, C<sub>9</sub>** : 33 pF céramique, au pas de 5,08mm

**C<sub>3</sub> à C<sub>6</sub>, C<sub>11</sub>, C<sub>12</sub>** : 100 nF

**C<sub>7</sub>, C<sub>18</sub>** : 10 µF/25V sorties radiales

**C<sub>10</sub>** : 22 µF/25V sorties radiales

**C<sub>13</sub> à C<sub>15</sub>** : 220 nF

**C<sub>16</sub>** : 470 µF/25V sorties radiales

**C<sub>17</sub>** : 47 µF/25V sorties radiales

**D<sub>1</sub>** : 1N4001 (diode de redressement 1A/100V)

**D<sub>2</sub>** : diode de puissance MR751 ou équivalent (4A à 5A, 100V)

**D<sub>3</sub> à D<sub>18</sub>** : BYV27

**JP<sub>1</sub>, JP<sub>2</sub>** : Jumpers au pas de 2,54mm

**QZ<sub>1</sub>** : quartz 16 MHz en boîtier HC49/U

**QZ<sub>2</sub>** : quartz 12 MHz en boîtier HC49/U (voir le texte à propos du P89C51RD2)

**REG<sub>1</sub>** : régulateur LM7805 (5V) en boîtier TO220

**RR<sub>1</sub>** : réseau résistif 8x10 kΩ en boîtier SIL

**R<sub>1</sub>, R<sub>5</sub>, R<sub>7</sub>** : 10 kΩ 1/4W 5% (marron, noir, orange)

**R<sub>2</sub>** : 18 Ω 1/4W 5% (marron, gris, noir)

**R<sub>3</sub>, R<sub>4</sub>** : 62 Ω 1/2W 5% (bleu, rouge, noir)

**R<sub>6</sub>** : 1 kΩ 1/4W 5% (marron, noir, rouge)

**SW<sub>1</sub>** : bloc de 4 micro-interrupteurs en boîtier DIL

**T<sub>1</sub>** : 2N2907A

**U<sub>1</sub>** : PCA82C250 ou PCA82C251 (+ adaptateur S08/DIP8 si disponible uniquement en version CMS)

**U<sub>2</sub>** : MC33164P

**U<sub>3</sub>** : SJA1000 ou 82C200 (gestionnaire de protocole CAN Philips)

**U<sub>4</sub>** : P89C51RD+ (ou P89C51RD2 : voir le texte pour QZ<sub>1</sub>)

**U<sub>5</sub>, U<sub>6</sub>** : L298 (circuit de commande de moteur pas à pas)

## Acoustique des salles Le guide de référence du praticien

Il existe beaucoup d'ouvrages spécialisés traitant de façon approfondie des divers domaines évoqués de l'acoustique. En revanche, rares à ce jour, pour ne pas dire inexistantes, sont ceux qui embrassent l'ensemble des sujets non superficiellement et établissent les relations entre eux. On ne trouve que de simples opuscules de vulgarisation.

*Le présent ouvrage se veut pragmatique, réaliste et concret ; il apportera une vue exhaustive, claire et précise aux chefs de projet, architectes, décideurs, décorateurs, entrepreneurs, techniciens de l'audiovisuel, ingénieurs du son, sonorisateurs, musiciens, comédiens et, aussi, aux étudiants concernés de près ou de loin par un des nombreux sujets traités ici. Les différentes facettes de l'acoustique des salles y sont exposées, illustrées d'exemples réels et concrets, assorties de nombreuses références bibliographiques. De ce fait, cet ouvrage constitue une solide base de départ, un guide ou un excellent moteur de recherche.*

292 pages - 270 F (41,16E) + 28 F de frais d'envoi soit au total 298 F (45,43E)

Thierry MALET - SONO Magazine

PGV, 2 à 12 rue de Bellevue Paris 75019  
Tél. : 01.44.84.84.84

