

# ELECTRONIQUE PRATIQUE

JUILLET / AOUT 1999

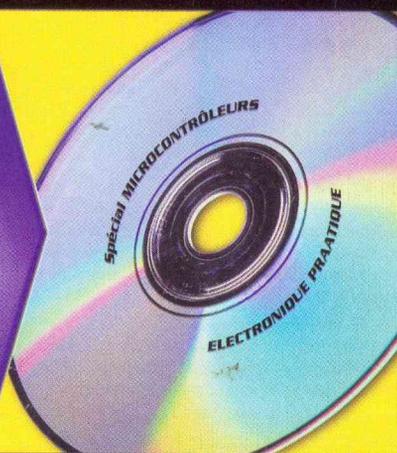
**Spécial**

## "MICROCONTRÔLEURS"

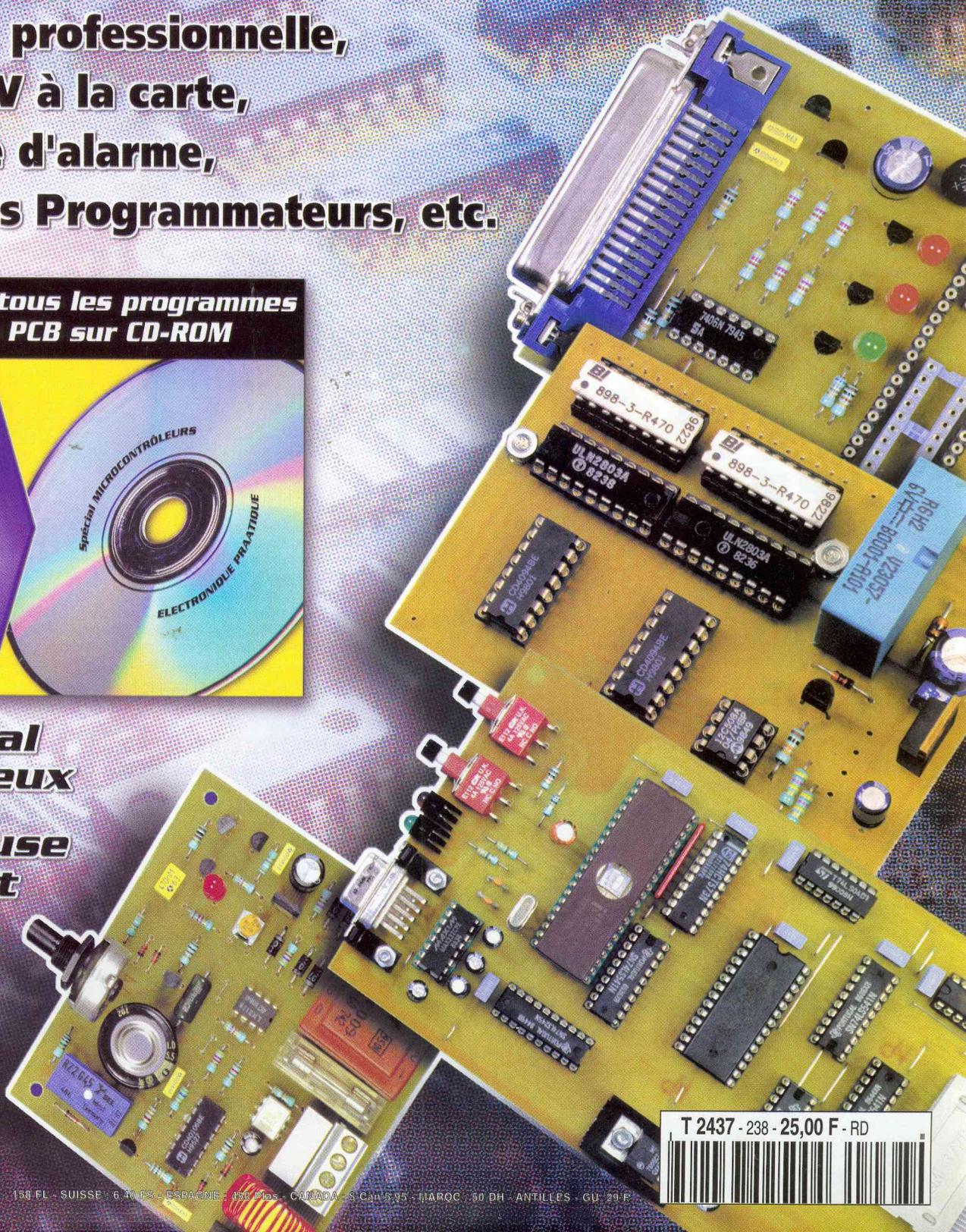
**Horloge professionnelle,  
Accès TV à la carte,  
Centrale d'alarme,  
plusieurs Programmeurs, etc.**

*Retrouvez tous les programmes  
et les PCB sur CD-ROM*

voir page  
**15**



- **Journal lumineux**
- **Veilleuse enfant**



T 2437 - 238 - 25,00 F - RD



# SOMMAIRE

## ELECTRONIQUE PRATIQUE

N° 238 - JUILLET-AOÛT 1999  
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  
Télex : 220 409 F

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

Directeur général adjoint : Jean-Louis PARBOT

Directeur graphique : Jacques MATON

Directeur de la rédaction : Bernard FIGHIERA (84.65)

Maquette : Jean-Pierre RAFINI

Avec la participation de : R. Amin, P. André,  
R. Basbug, U. Bouteville, H. Cadinot, O. Cotreau,  
A. Garrigou, M. Laury, L. Lellu, E. Lemery, Y. Mergy,  
P. Morin, P. Oguic, E. Quagliozi, A. Reboux,  
A. Sorokine, C. Tavernier.

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

Marketing : Corinne RILHAC Tél. : 01.44.84.84.52

Diffusion : Sylvain BERNARD Tél. : 01.44.84.84.54

Responsable des Ventes : Sylvain BERNARD

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)

Assisté de : Karine JEUFFRAULT (84.57)

Abonnement/VPC: Anne CORNET (85.16)

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 dernière 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, communiquez 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 (11 numéros) pour les USA est de 49 \$US et de 68 \$cnd pour le Canada.

**Electronique Pratique**, ISSN number 0243 4911, is published 11 issues per year by Publications Ventillard at P.O. Box 2769 Plattsburgh, N.Y. 12901-0239 for 49 \$US per year.

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



« Ce numéro  
a été tiré  
à 65 500  
exemplaires »

**BVP**  
Bureau de Vérification  
de la Publicité.

## Réalisez vous-même

106 Veilleuse pour enfant

## Spécial «Microcontrôleurs»

- 15 Le CD-ROM - 24 : Programmeur de PIC universel - 34 : Programmeur de PIC 16C84/16F84  
36 : Accès TV à la carte - 44 : Programmeur de 87C51/87C52 - 50 : Centrale d'alarme à microcontrôleur - 54 : Horloge professionnelle à LED  
62 : Programmeur de microcontrôleurs flash ATMEL - 70 : Télécommande IR 16 canaux  
80 - Communication série asynchrone avec un ST6230  
88 : Journal lumineux à persistance rétinienne  
96 : Interface RS232-I2C - 98 : émulateur d'EPROM.

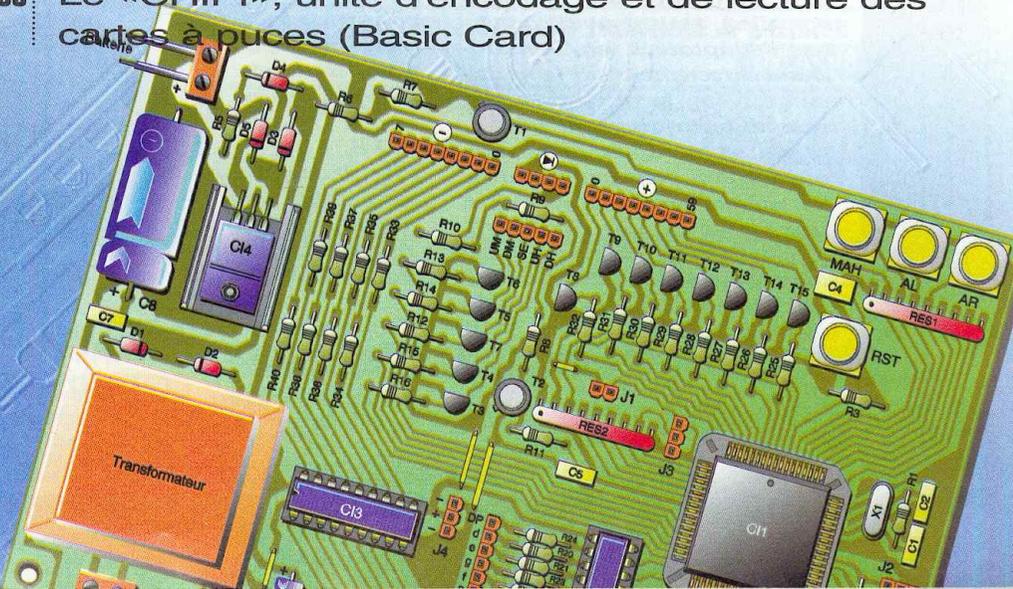
## Montages FLASH

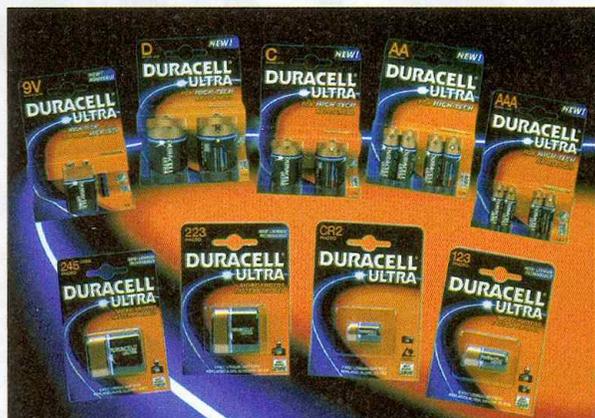
- 18 Émetteur codé 16 canaux  
20 Récepteur tout ou rien codé 16 canaux

## 04 Infos OPPORTUNITÉS

## DIVERS

- 10 Internet Pratique  
68 Le «CHIPI», unité d'encodage et de lecture des cartes à puces (Basic Card)





## DURACELL lance la gamme complète Duracell Ultra pour appareils fortement consommateurs d'énergie

D'une manière générale, les piles Duracell Ultra durent jusqu'à 50% plus longtemps que les piles alcalines standard. Duracell Ultra est spécialement conçue pour satisfaire les besoins accrus en énergie des appareils fortement consommateurs d'énergie comme les téléphones mobiles, les appareils photo numériques et avec flash, les baladeurs mini-discs et les caméscopes. La gamme complète Duracell Ultra et Duracell Ultra Lithium sera lancée avec un nouveau packaging, comportant des icônes d'appareils de haute technologie, pour une meilleure identification de la part des consommateurs.

En vente en hyper - GSB et détaillants spécialisés

**DURACELL France SNC**  
Tél. : 01.47.48.70.00



## Transformez tout multimètre numérique en thermomètre de précision

Les adaptateurs de température C.A. 801 et C.A. 803 sont des boîtiers compacts, capables de transformer tout multimètre numérique en véritable thermomètre professionnel (-40 à + 1000°C). Il suffit simplement de disposer d'un multimètre numérique à entrée mV DC, pour une lecture directe en °C ou °F, et d'impédance minimum 2 MΩ.

Mieux encore, si les bornes du multimètre sont distantes de 19 mm (ce qui correspond à l'entraxe standard d'un produit conforme aux normes actuelles), vous pouvez connecter l'adaptateur en l'embrochant directement sur le multimètre. Vous formez, ainsi, un ensemble parfaitement solide qui se manipule aussi aisément qu'un thermomètre, avec les mêmes performances et à prix nettement plus compétitif. Le C.A. 801 est livré d'origine avec un capteur filaire de 1,1 mm qui autorise des mesures de -40 à +200°C.

Avec ses deux entrées, le C.A. 803 permet, en plus, d'effectuer des mesures différentielles ( $\theta_1 - \theta_2$ ).

## Transformez tout multimètre numérique en luxmètre de précision

La cellule luxmétrique C.A. 808 est un capteur d'éclairage de précision. Équipée d'un capteur au sélénium, ses réponses spectrales et spatiales sont corrigées selon les normes (C.I.E.).

Elle se connecte sur tout multimètre numérique ou enregistreur à entrée tension mV DC et permet d'effec-

tuer des mesures d'éclairage jusqu'à 20000 lux.

Situé sous son boîtier, un insert au pas «Kodak» autorise une utilisation avec un

pied photo. Son cordon blindé de deux mètres, souple et solide, offre toutes facilités pour des manipulations plus aisées. Son cache imperdable, en élastomère, assure la protection de la partie sensible du capteur lors du stockage ou du transport de l'instrument.



**CHAUVIN ARNOUX**  
Tél. : 01.44.85.44.85

<http://www.chauvin-arnoux.com>

## Compo-PYRÉNÉES ouvre un département haut-parleurs



Spécialiste du composant, Compo-PYRÉNÉES étoffe son rayonnement par l'ouverture d'un département Audio/acoustique. Extension logique de la part de ce détaillant qui, de ce fait, propose dorénavant à sa clientèle de nombreuses solutions en Audio. Compo-PYRÉNÉES dispose d'un choix intéressant de haut-parleurs pour enceintes acoustiques,

de sonorisation et pour l'automobile. Les matériels électroniques sont aussi de la partie : amplificateurs de puissance de sonorisation et automobile complète la gamme. Il va sans dire que ce très jeune département évoluera compte tenu de la motivation et de la confiance que l'équipe de ce magasin place en lui.

A cette occasion, Compo-PYRÉNÉES nous propose une opportunité alléchante compte tenu de son rapport qualité/prix : Une paire de haut-parleurs pour automobile (de portière) 4Ω/30W, membrane papier, dôme tissu, suspension mousse, Ø 14 cm, au prix exceptionnel de 30 F. TTC la paire (dans la limite des stocks disponibles en magasin).

**Compo-PYRENEES • 312 rue des Pyrénées 75020 PARIS**  
**Tél. : 01.43.49.32.30**  
**Fax : 01.43.49.42.91**

## Multimètre mini prix ! maxi performances !



Le DVM 850 BL est un multimètre à commande manuelle 3 1/2 digit LCD. Ses caractéristiques sont riches compte tenu de sa petite taille :

- éclairage afficheur (backlight)
  - touche Hold (mémoire de la dernière valeur mesurée)
  - fiche 10A
  - mesures des tensions AC/CC et des courants CC,
  - test de continuité audible, diodes, transistors, résistances.
- Fourni avec coque de protection et jeu de cordons, son prix est de 119 F.TTC.

En vente sous blister dans le réseau de distributions :

**VELLEMAN Electronique**  
**Tél. : 03.20.15.86.15**

## Convertissez-vous

A l'approche des congés, bon nombre d'entre-vous goûteront aux joies du camping/caravaning avec les avantages et inconvénients que cela comporte. Le problème majeur, souvent, consiste en l'alimentation en courant 220V de nos appareillages électriques domestiques à partir du 12V. Electronique Diffusion nous réserve, ici, 3 modèles de convertisseurs :

- ITC300 : entrée 10/15V, sortie 220V/50 Hz/300 W, équipé d'une protection thermique. Son prix : 649 F TTC.

- ITC300+ : entrée 10/15V, sortie 220V/50 Hz/300 W, équipé d'une protection thermique et d'un fusible de protection 35A,

permet également de charger les batteries. Son prix : 829 F TTC.

- ITC600 : entrée 10/15V, sortie 220V/50 Hz/500 W, équipé d'une protection thermique. Son prix : 1225 F TTC.

Notamment disponibles chez :

**ELECTRONIQUE DIFFUSION**  
**Tél. : 03.20.70.23.42**



Le mois dernier, nous vous présentions une carte d'acquisition vidéo pour PC. Le signal vidéo-composite n'a donc plus de secret pour vous. Ce mois-ci, nous découvrirons un site dédié à l'acquisition et au traitement des signaux sonores. Nous nous dirigerons ensuite vers le site de la société DALLAS Semiconductor.

# internet PR@TIQUE

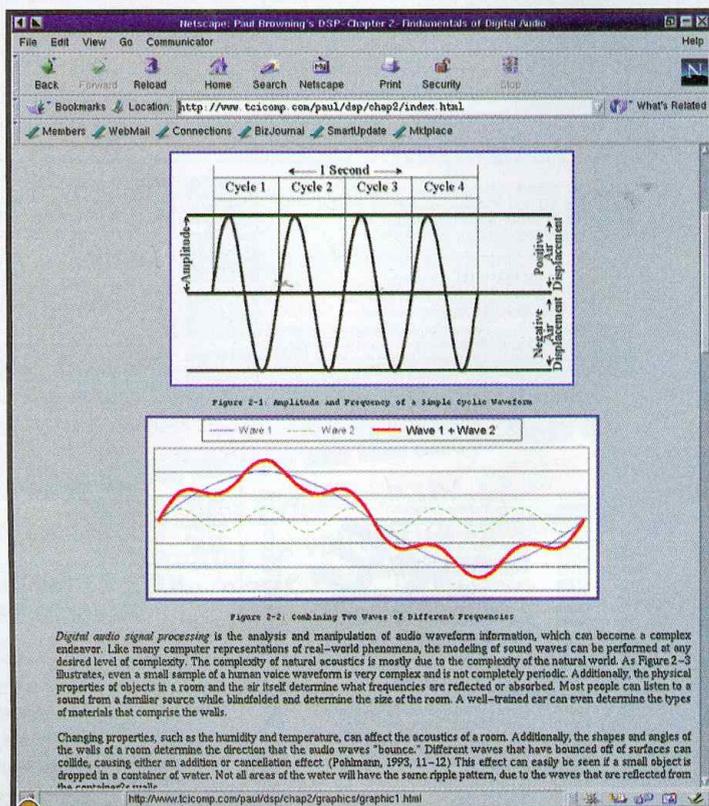
**A**vec l'introduction massive des ordinateurs multimédias dans les foyers, tout le monde ou presque dispose sans le savoir d'une puissance de calcul assez phénoménale. Les jeux vidéos sont d'ailleurs les applications

sonores temps réel comme l'écho, la distorsion, la réverbération et bien d'autres. Comme nous allons le voir, l'auteur ne se limite pas à proposer et à réaliser des algorithmes mais s'emploie à expliquer tous les phénomènes mis en jeu pour arriver au

nement interne des effets réalisés.

Il commence logiquement par décrire en détail le signal sonore (superpositions de signaux de différentes fréquences). Les explications sont agrémentées de schémas explicites donnés sous forme d'images (figure 1).

Il continue son cours par la description des systèmes d'acquisition des informations sonores. En effet, il faut d'abord transformer les données analogiques en informations numériques pour pouvoir les traiter par l'ordinateur. Ceci se fait par un convertisseur analogique/numérique. La fréquence d'acquisition est fixée par la bande passante que l'on veut conserver. Rappelons que l'oreille humaine est capable d'entendre les signaux dans la



1

## superpositions de signaux

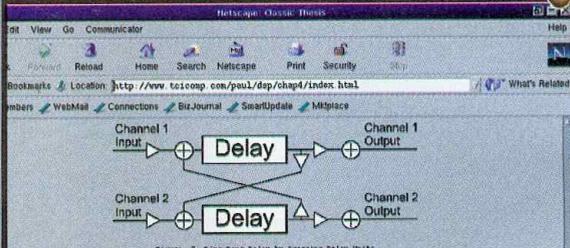
requérant le plus de ces possibilités. Le traitement du signal sonore temps réel sur ces machines est donc totalement envisageable. C'est ce que nous allons découvrir dans cette première partie en visitant le site disponible à l'adresse : <http://www.tciocomp.com/paul/dsp/index.html>.

Ce dernier nous présente des effets

résultat. Ce cours ne nécessite pas de pré-requis en traitement du signal et permettra donc à l'ensemble des lecteurs d'Électronique Pratique de comprendre le fonction-

## les différents algorithmes

2



le Reverb

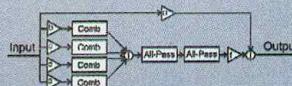
Acoustic hall, sound waves bounce off of the walls, seats, and floor, creating a torrent of echoes from these surfaces. Several people have tried to emulate this effect by studying the reflections caused by a single-source sound impulse in a room. In 1970, M.R. Schroeder did on several ways to simulate room reverberation by summing the outputs of multiple reverb units. (Meoer, 1979, 14) He first tried the chaining of several all-pass filters with a single feed-forward loop (Figure 4-8).

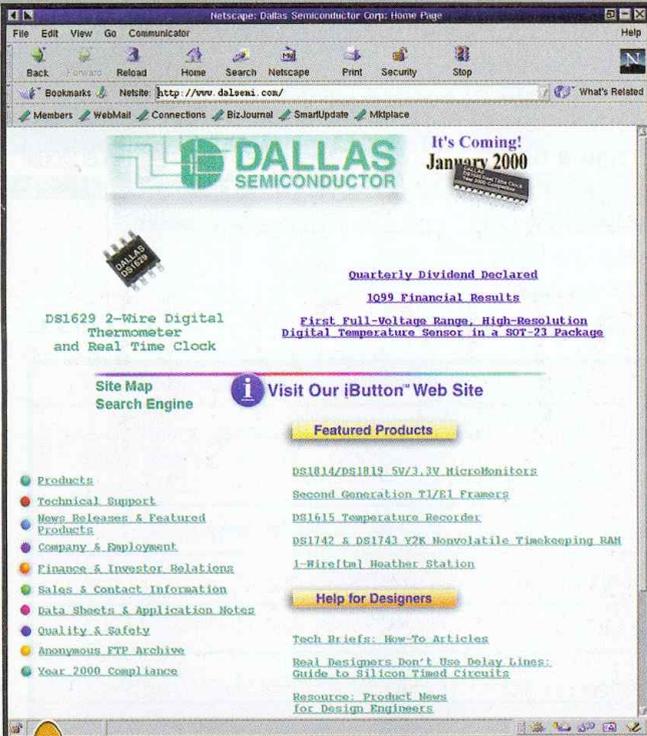


Meoer (1979, 15) points out the following problems with the results obtained by this configuration:

1. "The decay did not start with a dense sound and die out slowly in an exponential manner. In fact, the higher the order, the longer it took for the density to build up to a pleasing level. This produces the effect of a lag in the reverberation, as if the reverberation followed the sound by some hundreds of milliseconds."
2. "The smoothness of the decay seemed to be critically dependent on the choice of the parameters involved: the gains and delay lengths of the individual unit reverberators. Just changing one of the delay lengths from its prime number length to the next larger prime number, which could be a change as small as 2 samples, has been noticed to occasionally make the difference between a smooth-sounding decay and a ragged-sounding decay."
3. "The tail of the decay showed an annoying ringing, typically related to the frequencies implied by some of the delays. This produced a somewhat metallic sound that was generally found objectionable."

Another configuration suggested by Schroeder was a summation of three comb filters passed into a chain of all-pass filters, as shown in Figure 4-9.





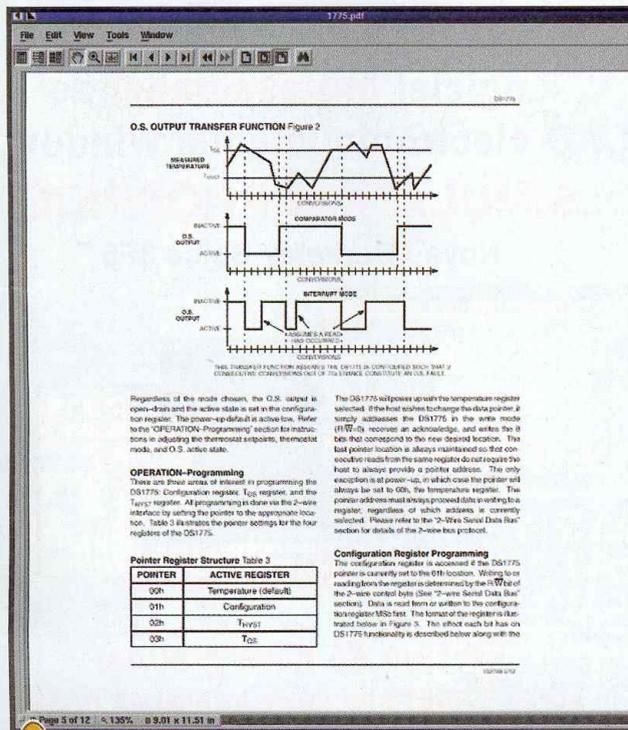
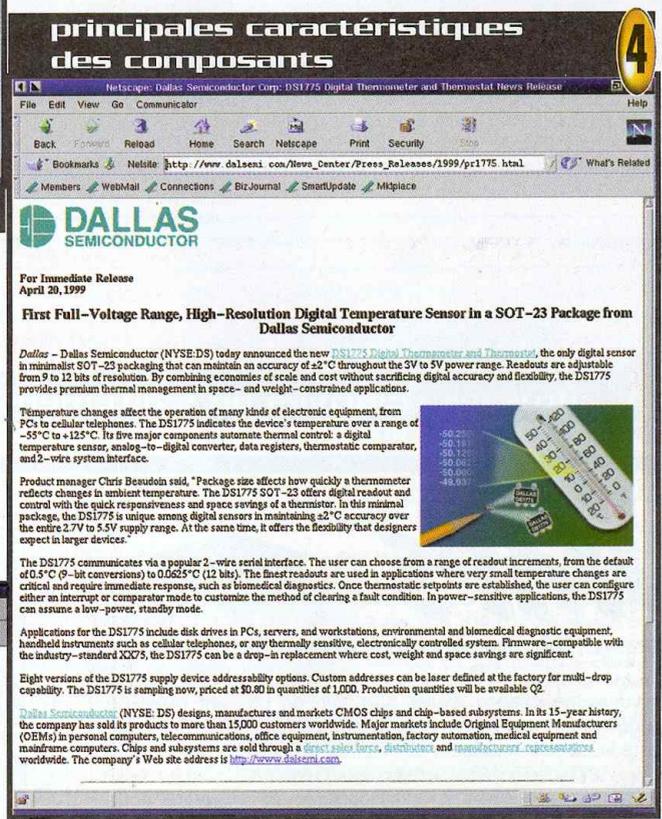
3 <http://www.dalsemin.com>

tranche 20 Hz-22 kHz. Le théorème de SHANON spécifiant qu'il faut échantillonner à une fréquence au moins 2 fois égale à la fréquence maximale du signal, la fréquence d'acquisition est fixée à 44 kHz (fréquence d'acquisition utilisée également pour les CD audio). Le cours se poursuit par la présentation de la chaîne de traitement. Le concepteur a

quasi-totalité des effets numériques. Nous trouvons ensuite les éléments de calcul et de traitement (mémoire et microprocesseur). La chaîne

choisi d'utiliser la carte sonore comme système d'acquisition. Il aurait très bien pu créer une carte personnelle mais à quoi bon ajouter des fonctionnalités déjà disponibles sur son ordinateur ? La chaîne se poursuit par un buffer d'entrée. Ceci est primordial pour la

représentés sous forme de diagrammes (figure 2). L'auteur termine son document par la description du programme qu'il a réalisé sous Windows et qui reprend tous les effets décrits préalablement. Ce dernier est téléchargeable gratuitement depuis le site. En conclusion, ce site présente tous les algorithmes de base de traitement du signal audio. Si vous n'avez pas le courage de lire tous les aspects théoriques, vous pourrez quand même utiliser le programme pour vous rendre compte de ce que l'on peut faire avec les ressources de son ordinateur. Nous terminons cette partie par une liste



5 DataSheet d'un circuit retenu

se termine par la diffusion du signal traité sur les haut-parleurs de la machine. Les 4 chapitres suivants présentent les différents algorithmes mis en jeu pour la création des effets. Ceux-ci sont expliqués en détail et

d'adresses relative au traitement audio (tableau 1). Comme nous vous l'annonçons dans notre introduction, la deuxième partie d'Internet Pratique sera consacré au site de DALLAS Semiconductor disponible à l'adresse <http://www.dalsemin.com> (figure 3). DALLAS Semiconductor s'est spécialisé dans des composants spécifiques qui concerne l'acquisition de température. Bien sûr, elle se positionne aussi sur des marchés plus importants comme celui des mémoires ou des microcontrôleurs. Le site a une architecture on ne peut plus classique : le sommaire se trouve dans une colonne à gauche de l'écran alors que la

partie utile se trouve au centre. On retrouve aussi les rubriques habituelles des sites américains (Products bien sûr mais aussi Finance & Investor Relations ou Employment). Pour chacun des composants de la marque, le site présente les principales caractéristiques sous forme d'une description HTML (**figure 4**).

Lorsqu'un circuit retient votre attention, vous pouvez télécharger sa DataSheet complète au format PDF (**figure 5**). Nous terminons cette brève description en attirant votre attention sur le fait que DAL-LAS Semiconductor réserve une rubrique entière au bug de l'an 2000 et certifie que tous ses composants sont compatibles

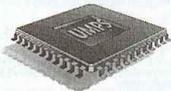
avec le nouveau millénaire. En vous souhaitant de longues et agréables visites sur Internet, nous vous donnons rendez-vous le mois prochain pour de nouvelles découvertes.

L. LELLU

<a href="http://arts.ucsc.edu/ems/music/tech_background/TE-16/teces_16.html">http://arts.ucsc.edu/ems/music/tech_background/TE-16/teces_16.html</a> - Basics of Digital Recording - digital audio introduction
<a href="http://www.digital-recordings.com/pub/pubneq.html">http://www.digital-recordings.com/pub/pubneq.html</a> - Consequences of Nyquist Theorem for Acoustic Signals Stored in Digital Format - published in proceedings from Acoustic Week in Canada 1991 - CAA Conference, Edmonton, Alberta, Canada, October 7 - 10, 1991, from <A HREF=>http://www.digital-recordings.com/ - Digital Recordings
<a href="http://www.bdti.com/faq/dsp_faq.htm">http://www.bdti.com/faq/dsp_faq.htm</a> - FAQs (Frequently asked questions with answers) on Digital Signal Processing
<a href="http://www.ticomp.com/paul/dsp/index.html">http://www.ticomp.com/paul/dsp/index.html</a> - Fundamentals of Digital Audio - and more
<a href="http://www.peavey.com/html/digital_audio_terms.html">http://www.peavey.com/html/digital_audio_terms.html</a> - Glossary of Digital Audio
<a href="http://www.pacificnet.net/~rexk/aeshome.htm">http://www.pacificnet.net/~rexk/aeshome.htm</a> - High Capacity Digital Audio - WWW site is for use of the AES SC-02-M Task Force
<a href="http://www.digital-recordings.com/pub/pubrec.html">http://www.digital-recordings.com/pub/pubrec.html</a> - Introduction to Digital Recording Techniques - paper was published in Canadian Acoustical Association Journal, 1989, from <A HREF=>http://www.digital-recordings.com/ - Digital Recordings
<a href="http://www.digido.com/Audiolevels.html">http://www.digido.com/Audiolevels.html</a> - Level Practices in Digital Audio
<a href="http://www.digido.com/morebits.html">http://www.digido.com/morebits.html</a> - More Bits ---Please! - how to pick right digital audio workstation (DAW)
<a href="http://tjev.tel.etf.hr/josip/DSP/sigproc.html">http://tjev.tel.etf.hr/josip/DSP/sigproc.html</a> - Signal Processing information page

**11 adresses relatives au traitement audio**

# UMPS



UNIVERSAL MICROPROCESSOR PROGRAM SIMULATOR

Donnez vous une chance d'aller plus loin en utilisant l'outil de développement par excellence : UMPS, un produit 100% Français, ouvert vers le monde extérieur.

UMPS est un simulateur de microcontrôleurs universel sous Windows (3.11, 95 ou NT4). UMPS simule le microcontrôleur associé à son environnement : panneau LCD, EEPROM (série et I2C), analyseur logique, convertisseurs A/N-N/A, convertisseur sur liaison série, LED, afficheurs, fonctions logiques ...

- Assembleur/Désassembleur universel intégré,
- Simulateur de CPU ultra-rapide,
- Simulateur de composants externes extensible...

UMPS offre un environnement de développement unique pour différentes familles de microcontrôleurs: Motorola HC05, HC08, HC11, PIC Microchip, 8051 et dérivés, SGS-Thomson ST6x, Atmel AVR, National Semiconductor COP8xx, SCENIX SX ... Un simple clic de souris suffit.

UMPS-démo est inclus dans le CD-Rom de ce numéro !

<http://www.vmdesign.com>

## Virtual Micro Design

Technopole Izarbel - 64210 BIDART

05.59.43.84.58 Fax 05.59.43.84.01

[umps@vmdesign.com](mailto:umps@vmdesign.com)

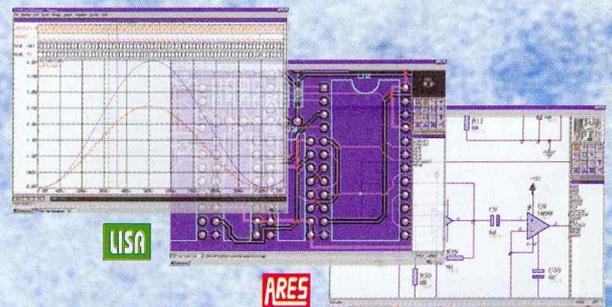


# PROTEUS 4.5

Logiciel professionnel de CAO électronique sous Windows™

## SIMULATION PROSPICE

Noyau Berkeley Spice 3F5



Version de base gratuite sur INTERNET <http://www.multipower-fr.com>

## Multipower

83-87, avenue d'Italie - 75013 Paris

Standard : 01 53 94 79 90 - Support technique : 01 53 94 79 95

Télécopie : 01 53 94 08 51 -

E-mail : [multipower@compuserve.com](mailto:multipower@compuserve.com)

# Émetteur codé

## 16 canaux

### ► A quoi ça sert ?

Un émetteur 16 canaux est capable de donner 16 ordres différents. Ils seront reçus puis interprétés par un décodeur muni de 16 sorties. Le tout travaille dans une bande largement autorisée... Quant à l'exploitation de ce système de commande, nous la laissons à votre imagination : commande domestique, modèle réduit de bateau, etc., le système pouvant être utilisé pour des accessoires en complément d'une télécommande à 2 voies.

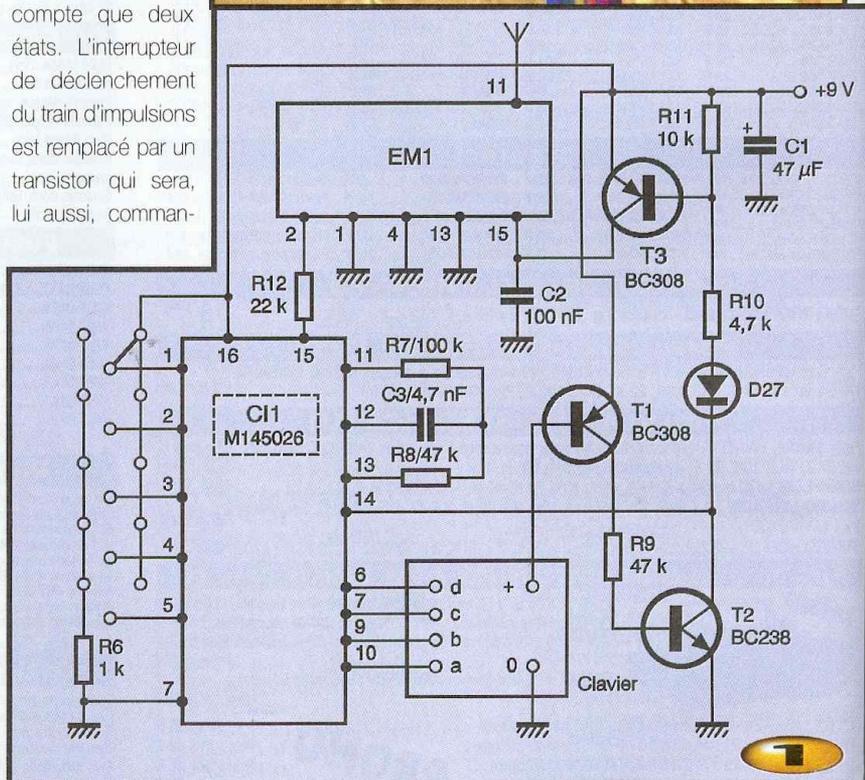
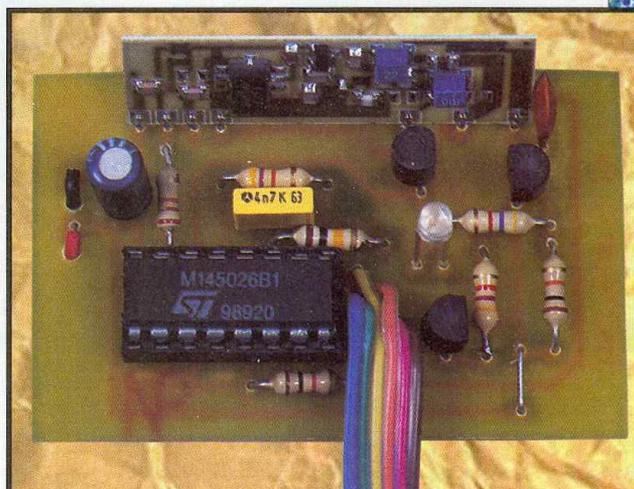
### Comment ça marche ?

La partie émission utilise un circuit très connu puisqu'il s'agit d'un module AUREL piloté par un résonateur à onde de surface travaillant à 433 MHz. La puissance de sortie, voisine d'une dizaine de mW, permet d'obtenir une portée d'une quarantaine de mètres, autrement dit une distance suffisante pour perdre de vue un modèle.

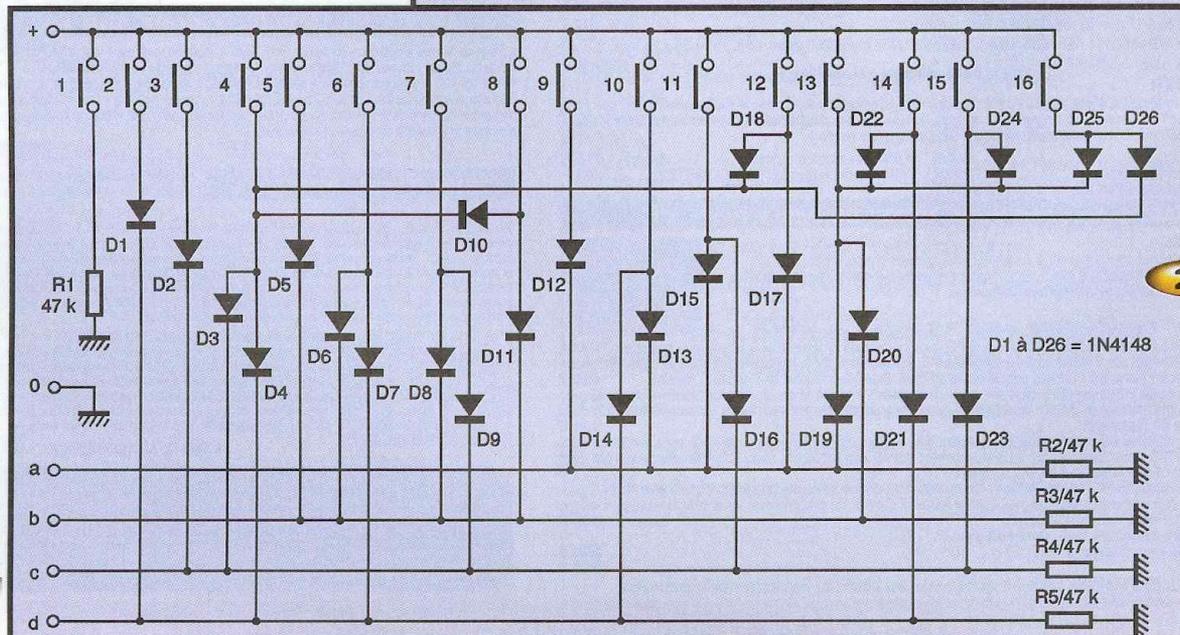
Pour limiter la consommation, nous avons installé un système de mise sous tension automatique, le module d'émission sera mis sous tension uniquement lors de la manipulation de l'une des touches.

Le codeur est un 145026, un circuit proposé par plusieurs fabricants, il s'agit d'un circuit de télécommande codé dont

on modifie les quatre derniers bits pour définir les 16 canaux. Le codage de ce circuit intégré se fait sur trois états mais dans le cas d'une transmission de données, ce que l'on fait ici, le décodeur ne prendra en compte que deux états. L'interrupteur de déclenchement du train d'impulsions est remplacé par un transistor qui sera, lui aussi, comman-



**1**



**2**

dé par n'importe laquelle des touches grâce au circuit que nous allons voir. Ce transistor,  $T_2$ , est relié à la broche 14 du circuit intégré.

En l'absence de signal de commande, les entrées de codage sont mises à la masse par des résistances de  $47\text{ k}\Omega$ . L'espace base/émetteur du transistor PNP  $T_1$  sera parcouru par le courant transitant par cette résistance chaque fois qu'une touche sera actionnée. Le courant de collecteur commande alors  $T_2$  et  $T_3$ .

Dans la série des 16 codes, il en est un particulier, celui qui correspond aux quatre zéros. Pour ce code, nous avons une touche qui se contente de faire débiter du courant dans une résistance, les quatre entrées restent au zéro mais nous avons émission du code par la mise à la masse de la broche 14.

Le circuit de codage dispose par ailleurs d'un codage de l'émission sur 5 bits, un codage cette fois permanent.

Les touches sont interconnectées par des diodes qui constituent des portes ET, en combinant certaines touches

grâce à deux diodes placées en série, il est possible de réduire d'environ 20% le nombre de diodes nécessaires.

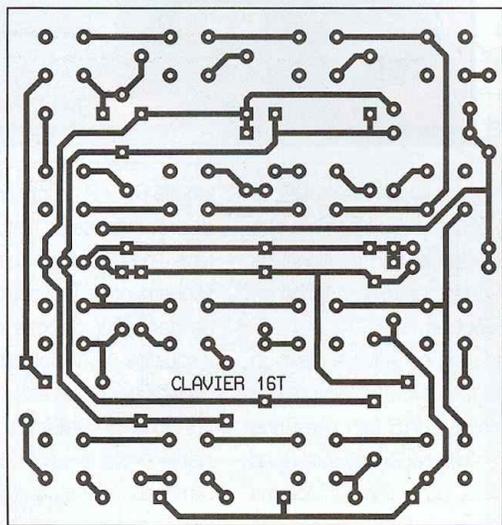
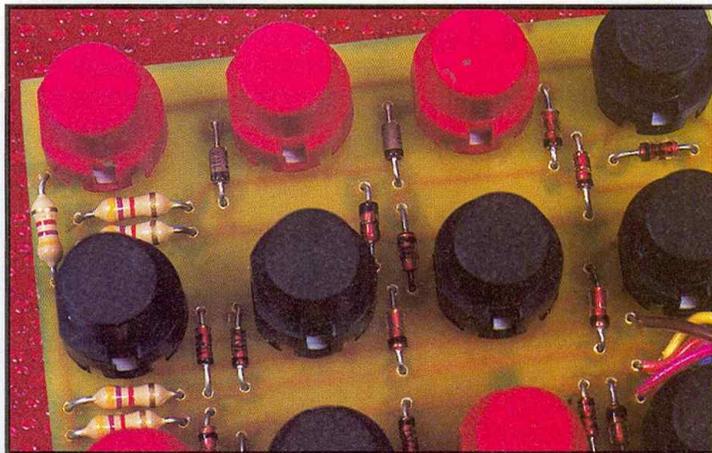
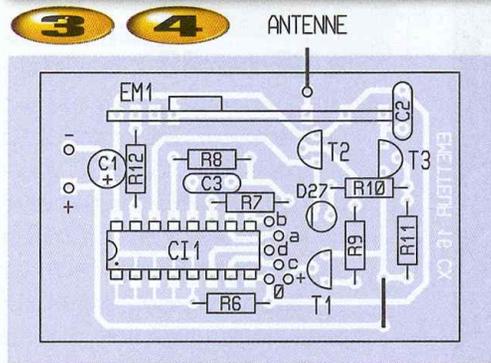
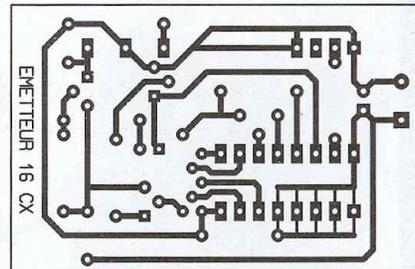
## Réalisation

L'émetteur est constitué de deux éléments, un clavier et un module de codage et d'émission. Si vous n'avez pas besoin des 16 touches, vous pourrez personnaliser votre clavier en conservant la partie codage et émission. Les diodes devront, bien sûr, être montées dans le bon sens, en cas d'inversion, le courant ne passe pas et il n'y

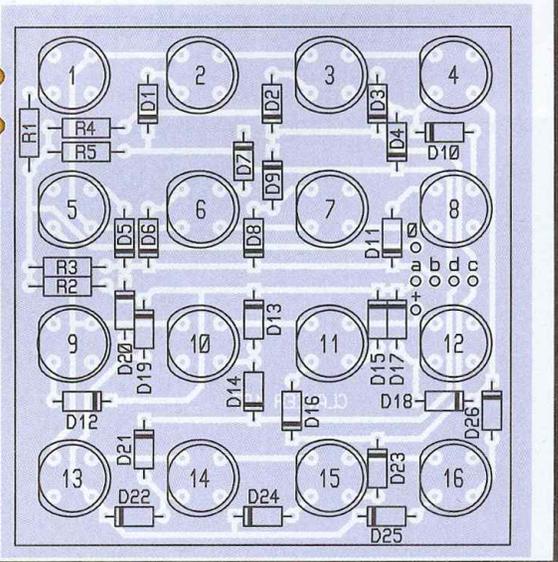
a pas d'émission. On devra aussi se méfier de la position des touches, leur orientation se base sur le méplat latéral. Une rotation de  $90^\circ$  risque d'entraîner une émission permanente. On aura intérêt à plier les pattes des diodes en utilisant un calibre, l'esthétique de la réali-

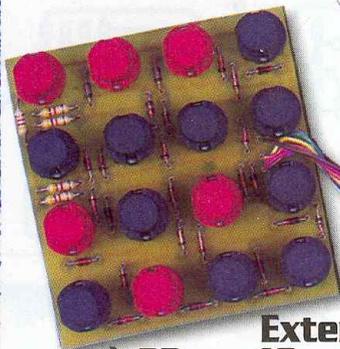
sation y gagnera. Sur l'implantation, nous avons marqué le centre du bouton, vous pourrez donc vous servir de ce repère pour usiner la façade de votre émetteur. Les dimensions sont prévues pour un coffret DIPTAL G1174, la diode électroluminescente sera collée en façade et reliée au circuit imprimé par fils.

Le codage de l'émission s'effectue en coupant les liaisons fines installées entre les broches de codage de l'émission, broches 1 à 5. Ici, on coupera une liaison ou les deux, le codage se fait en mettant la broche au +, au - ou en le laissant en l'air. Nous avons installé une résistance pour mettre à la masse les broches, cette résistance est uniquement là pour éviter de mettre l'alimentation en court-circuit total.



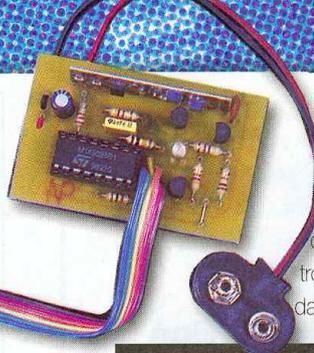
5  
6





### Extensions à 32 ou 48 canaux

Cette technique d'émission peut être utilisée pour une version à 32, voire 48 canaux. Pour ce faire, il suffit de relier la broche 5 du circuit intégré  $CI_1$  à un inverseur à trois positions, chacune des positions correspond à un état différent pour le bit de codage 5. Nous avons donc maintenant 48 codes soit un codage sur 5 bits



à notre disposition. Pour la réception, il suffit de mettre derrière le récepteur trois circuits de décodage dont le bit 5 sera

respectivement positionné sur le +, le - est laissé en l'air. Derrière chacun des décodeurs, on installera un décodeur 4/16.

E. LEMERY

### Nomenclature

$R_1$  à  $R_5$ ,  $R_8$ ,  $R_9$  : 47 k $\Omega$  1/4 W 5% (jaune, violet, orange)  
 $R_6$  : 1 k $\Omega$  1/4 W 5% (marron, noir, rouge)  
 $R_7$  : 100 k $\Omega$  1/4 W 5% (marron, noir, jaune)  
 $R_{10}$  : 4,7 k $\Omega$  1/4 W 5% (jaune, violet, rouge)  
 $R_{11}$  : 10 k $\Omega$  1/4 W 5% (marron, noir, orange)  
 $R_{12}$  : 22 k $\Omega$  1/4 W 5% (rouge, rouge, orange)  
 $C_1$  : 47  $\mu$ F/10V chimique radial  
 $C_2$  : 100 nF céramique  
 $C_3$  : 4,7 nF MKT 5mm  
 $T_1$ ,  $T_3$  : transistors PNP BC308

$T_2$  : transistor NPN BC548  
 $CI_1$  : Circuit intégré M145026  
 $D_1$  à  $D_{26}$  : diodes silicium 1N4148  
 $D_{27}$  : diode électroluminescente 3 mm  
 $EM_1$  : Module d'émission AUREL TX SAW433/s  
 Boutons 1 à 16 : boutons poussoirs ITT D6, couleurs variées  
 Coffret DIPTAL G1174  
 AUREL importé par : P2M, 8 Allée des Châtaigniers, ZA du Buisson de la Coudre, 78190 Trappes et distribué, entre autres, par SELECTRONIC

# Récepteur tout ou rien codé 16 canaux

## ► A quoi ça sert ?

Ce récepteur codé est capable de commander 16 fonctions différentes. Il fonctionne sur la fréquence autorisée de 433,92 MHz et ne coûte pas très cher compte tenu de ses prestations.

## Comment ça marche ?

Nous avons confié la réception à un module 433 MHz d'AUREL, une formule simple et économique. Le récepteur est un modèle à super réaction doté d'un préamplificateur qui évite de rayonner trop d'énergie vers l'extérieur. Il s'alimente sur pile et constitue une formule de réception simple et efficace. Le signal arrive sur la broche 3 du circuit hybride, est traité à l'intérieur du circuit et ressort démodulé sur la broche 14. Le circuit n'a que 8 broches mais elles se comptent de 1 à 15, on tient compte du pas de 2,54 et non du nombre de broches. Signalons aussi que pratiquement tous

les récepteurs de ce constructeur utilisent le même brochage, ce qui simplifie les échanges, si vous désirez une formule consommant moins d'énergie,

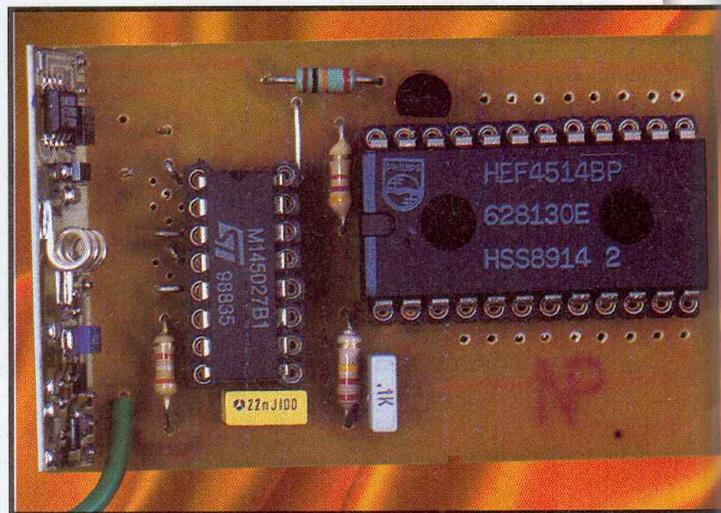
vous trouverez dans la gamme AUREL le récepteur NB-05L qui ne consomme que 600  $\mu$ A. La broche 13 sort un signal de test qui permet de contrôler le fonctionnement du récepteur.

Le signal démodulé est envoyé dans un circuit de décodage 145027. Un premier codage s'effectue sur 5 bits avec une entrée à trois niveaux. On codera donc les premiers bits du récepteur comme ceux de l'émetteur. Des composants stratégiques entourent le circuit, les résis-

tances  $R_1$  et  $R_2$  sont choisies en fonction des résistances de l'oscillateur de l'émetteur. Une différence entraîne un dysfonctionnement de la réception.

Le décodeur dispose de 4 bornes sur lesquelles on dispose des quatre bits du codage des 16 états, une borne distribue un signal de validation de la qualité du signal et de la similitude de code entre l'émetteur et le récepteur.

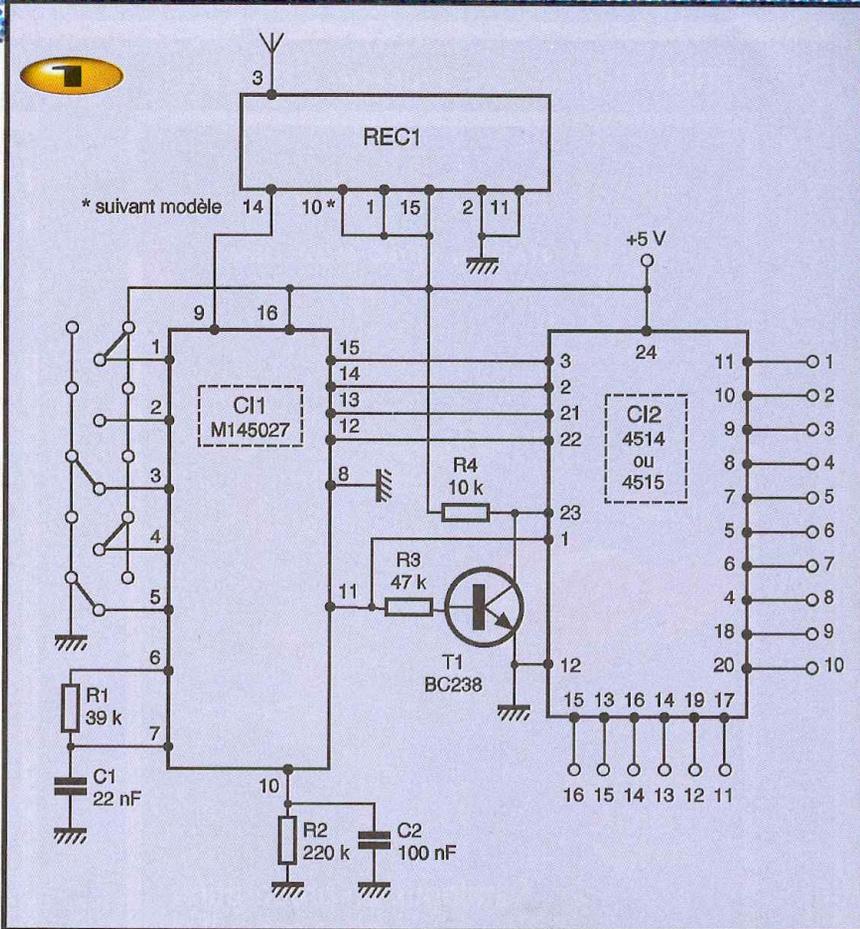
Les sorties du circuit  $CI_1$  sont reliées à un décodeur 4 entrées 16 sorties CMOS



type 4514 ou 4515. La différence entre les circuits réside dans la polarité du signal de sortie, le 4514 sort des impulsions positives, le 4515 des négatives. Ces circuits disposent d'une entrée d'inhibition. Le 145027 mémorise les informations qu'il a reçu. Si on relie directement ses sorties aux entrées du 4514 ou 15, on obtiendra un état permanent qui changera à chaque réception d'ordre nouveau. En associant l'information de validation, on doit inverser sa polarité avec l'entrée d'inhibition, on dispose de 16 sorties fugitives. Les sorties sont ici de type CMOS, pour obtenir une puissance plus importante, pour commander un relais ou un moteur, on installera une amplification à un ou plusieurs transistors. Si vous désirez faire suivre le décodeur de bascules RS réalisées à partir de 4011, le 4515 simplifie la réalisation et évite l'utilisation d'inverseurs. L'alimentation s'effectue par une tension voisine de 5V, par exemple 3 piles de 1,5V ou 4 éléments Ni-Cd.

## Réalisation

Les figures 2 et 3 donnent le circuit imprimé et l'implantation des composants. La réalisation est simple et ne pose pas de problème, le récepteur est un module qui ne demande aucun réglage, il suffit de le souder pour qu'il fonctionne. Les sorties ont été numérotées en fonction des touches de l'émetteur. Le 4514/15 a ses bornes réparties dans un ordre pas vraiment logique, par ailleurs, il n'est pas

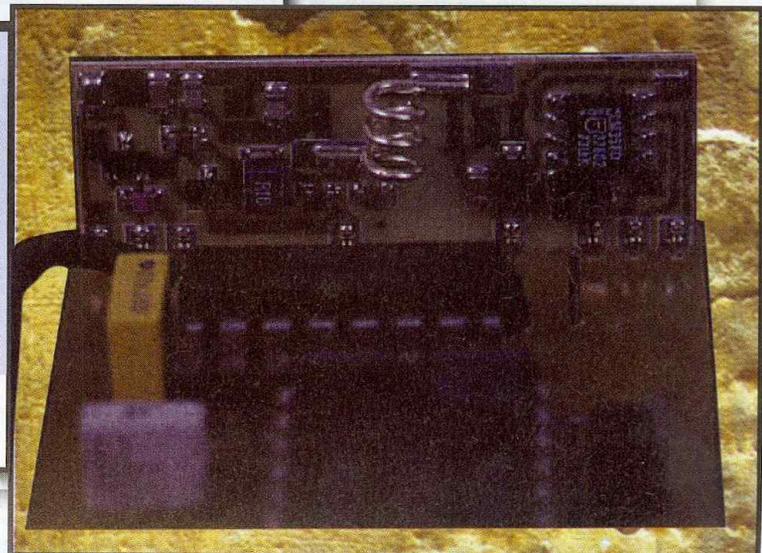
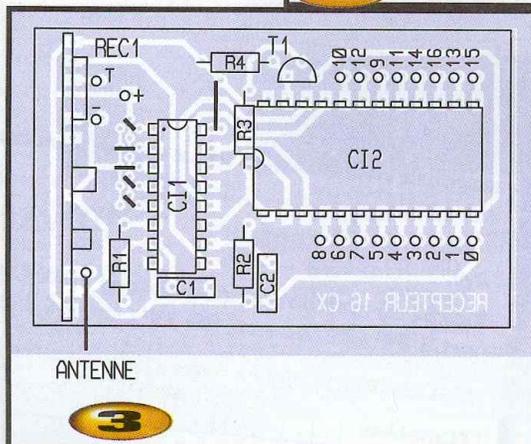
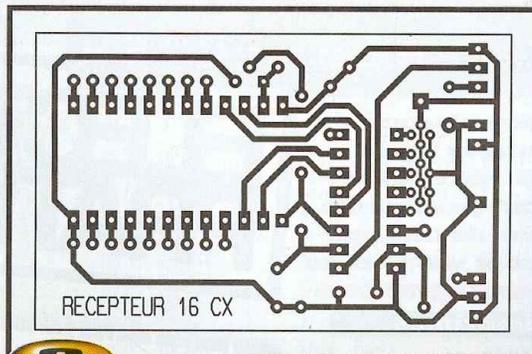


obligatoire de respecter l'ordre des bits, nous avons effectué une liaison directe entre la sortie du 145027 et le 4514/15. Une antenne de 18 cm environ constituera un brin de réception quart d'onde.

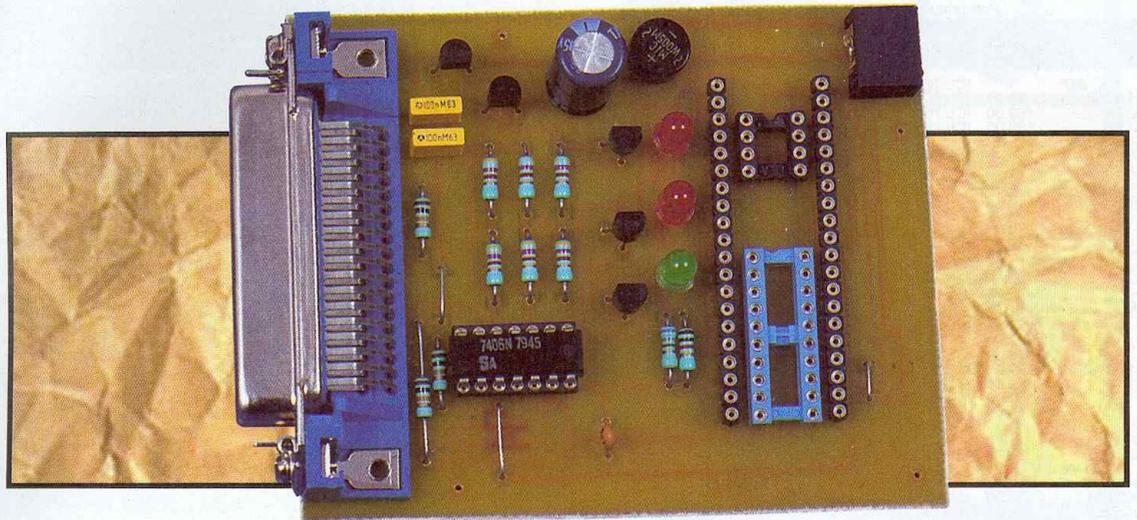
E. LEMERY

## Nomenclature

**R<sub>1</sub>** : 39 kΩ 1/4 W 5% (orange, blanc, orange)  
**R<sub>2</sub>** : 220 kΩ 1/4 W 5% (rouge, rouge, jaune)  
**R<sub>3</sub>** : 47 kΩ 1/4 W 5% (jaune, violet, orange)  
**R<sub>4</sub>** : 10 kΩ 1/4 W 5% (marron, noir, orange)  
**C<sub>1</sub>** : 22 nF MKT 5 mm  
**C<sub>2</sub>** : 100 nF MKT 5 mm  
**T<sub>1</sub>** : transistor NPN BC238 ou équivalent  
**CI<sub>1</sub>** : Circuit intégré M145027  
**CI<sub>2</sub>** : Circuit intégré CMOS HEF, CD4514 (sortie positive) ou 4515 (sortie négative).  
**REC<sub>1</sub>** : Module super réaction AUREL NB-05M, BC-NB, RF290A importé par P2M, 8 Allée des Châtaigniers, ZA du Buisson de la Couldre, 78190 Trappes et distribué, entre autres, par SELECTRONIC.



# Un programmeur de PIC universel



## Présentation

Si la réalisation de programmeurs pour certaines familles de microcontrôleurs est un travail délicat ou peu rentable, ce n'est pas le cas ici puisque vous allez voir dans un instant qu'avec une poignée de composants peu coûteux il est possible de

programmer, non pas un ou deux circuits, mais quasiment tous les microcontrôleurs PIC actuels à de très rares exceptions près.

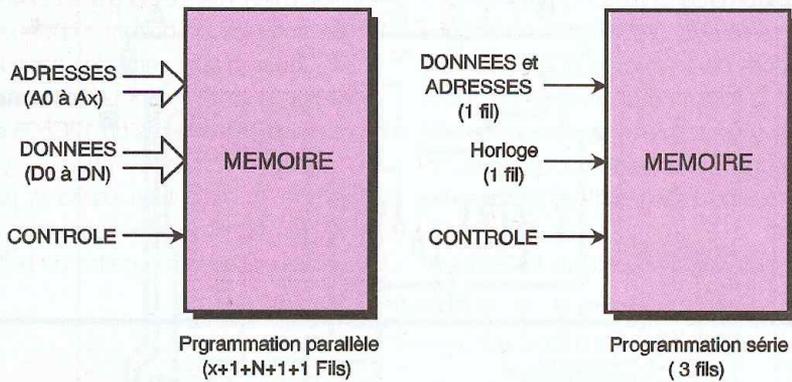
Même s'il ne peut recevoir le qualificatif de programmeur de production, notre montage respecte suffi-

## 1 Liste des circuits PIC

Device		
0: PIC12C508	17: PIC16C63	34: PIC16C74
1: *PIC12C508A	18: PIC16C64	35: PIC16C74A
2: PIC12C509	19: PIC16C64A	36: PIC16C76
3: *PIC12C509A	20: PIC16CR64	37: PIC16C77
4: *PIC16C506	21: PIC16C65	38: PIC16F83
5: PIC12C671	22: PIC16C65A	39: PIC16CR83
6: PIC12CE673	23: PIC16C66	40: PIC16C84
7: PIC12C672	24: PIC16C67	41: PIC16F84
8: PIC12CE674	25: PIC16C620	42: PIC16CR84
9: PIC14000	26: PIC16C621	43: PIC16C923
10: PIC16C554	27: PIC16C622	44: PIC16C924
11: PIC16C556	28: PIC16C710	45: *PIC16C642
12: PIC16C558	29: PIC16C71	46: *PIC16C662
13: PIC16C61	30: PIC16C711	47: *PIC16C715
14: PIC16C62	31: PIC16C72	48: PIC_NEW
15: PIC16C62A	32: PIC16C73	
16: PIC16CR62	33: PIC16C73A	

Device:  

Les microcontrôleurs PIC de MICROCHIP connaissent un succès croissant dans l'industrie ou auprès des amateurs. Les raisons en sont multiples mais, les plus importantes d'entre elles sont : vitesse d'exécution des programmes, prix de vente très compétitif, une excellente disponibilité et enfin des outils de développement gratuits ou d'un prix de revient dérisoire. Parmi ceux-ci, le programmeur occupe une place de choix car c'est le dernier maillon de la chaîne mais c'est aussi l'élément le plus important puisqu'il permet de programmer la mémoire du microcontrôleur avec le fruit de votre travail.



**2 Principe de la programmation parallèle**

N° de patte	Fonction en mode normal	Fonction en programmation
1 (VDD)	Alimentation positive	idem
2	GP5 ou OSC1 ou CLKIN	idem
3	GP4 ou OSC2	idem
4	GP3 ou MCLR	VPP
5	GP2 ou TOCKI	idem
6	GP1	Horloge de programmation
7	GP0	Données de programmation
8 (VSS)	Masse	idem

**Les fonctions des pattes changent selon que le circuit est ou non en mode programmation.**

samment les spécifications de MICROCHIP pour correspondre à la catégorie des programmeurs que ce fabricant appelle programmeurs de développement. Au moment où ces lignes sont écrites (fin mai), il peut programmer tous les circuits PIC dont la liste vous est fournie **figure 1** mais, compte tenu de son schéma et de l'évolution constante de son logiciel de commande, cette liste est susceptible d'évoluer au fur et à mesure de la mise sur le marché de nouveaux circuits.

Comme nombre de ses homologues, notre montage se connecte sur le port parallèle de tout compatible PC mais ici, point n'est besoin du dernier Pentium III à 450 MHz puisqu'un vieux PC sous DOS peut suffire, sans pour autant devoir renoncer à un excellent confort d'utilisation.

Notre programmeur nécessite en outre une alimentation qui peut être continue ou alternative, comprise entre 12 et 30V et qui n'a pas besoin d'être stabilisée. Un bloc secteur "prise de courant" ou n'importe quelle alimentation de laboratoire peut donc convenir d'autant que le débit nécessaire est inférieur à 100 mA.

Comme les programmeurs professionnels, notre montage est évidemment capable de lire, vérifier, programmer et comparer les PIC sans aucune restriction, de même qu'il sait lire et programmer leurs "fusibles" de configuration. Il sait aussi, bien sûr, effacer les circuits munis de mémoire de type EEPROM (voir ci-dessous) et permet l'accès à la mémoire de données des circuits dotés d'une telle mémoire lorsqu'elle est en technologie EEPROM également. Autant dire qu'il est vraiment complet et devrait satisfaire aussi bien le développeur occasionnel que l'utilisateur intensif de circuits PIC.

**Un peu de terminologie**

Les mémoires de programmes contenues dans tous les microcontrôleurs actuels, quelle qu'en soit la marque, peuvent être de différents types ce qui conditionne leurs méthodes de programmation. On distingue en effet :

- les ROM programmables par masque, que l'on rencontre sur les circuits destinés à de la production en grande série. Ces

mémoires ne peuvent en effet être programmées que par le fabricant lors de la phase de masquage du circuit intégré ce qui impose, pour des raisons évidentes de rentabilité, de produire un très grand nombre de pièces identiques (1000 est un minimum).

- Les EPROM qui sont des mémoires programmables électriquement grâce à une "haute" tension et à des chronogrammes particuliers. Ces mémoires se subdivisent en deux sous-familles avec : les UVEPROM ou mémoires effaçables aux ultraviolets et les OTPROM ou mémoires programmables "une fois" (One Time PROM). Dans les deux cas, la programmation est électrique mais, alors que les OTPROM sont programmées une fois pour toutes, les UVEPROM peuvent être effacées par exposition aux ultraviolets. Elles sont reconnaissables physiquement à leur boîtier muni d'une fenêtre en quartz laissant voir la puce.
- Les EEPROM ou E2PROM qui sont programmables et effaçables électriquement et qui représentent donc le nec plus ultra pour du développement car elles peuvent être programmées et effacées jusqu'à 1000 fois au minimum (10 000 ou 100 000 selon les fabricants et la technologie).

A la lecture de ce qui précède, vous vous demandez peut-être pourquoi on ne produit pas que des EEPROM puisque ce sont les plus pratiques d'emploi. La raison est simple et tient à leur technologie de fabrication qui est complexe et donc coûteuse, aussi bien en argent qu'en surface de puce. Ces mémoires sont donc hélas les moins répandues et les plus chères.

Les microcontrôleurs PIC de MICROCHIP existent, selon les références choisies, avec les quatre technologies que nous venons de décrire sommairement et notre programmeur est à même de traiter les trois dernières puisque, comme nous l'avons dit, la programmation par masque ne se conçoit qu'au stade de la fabrication du circuit.

**De la programmation parallèle à la programmation série**

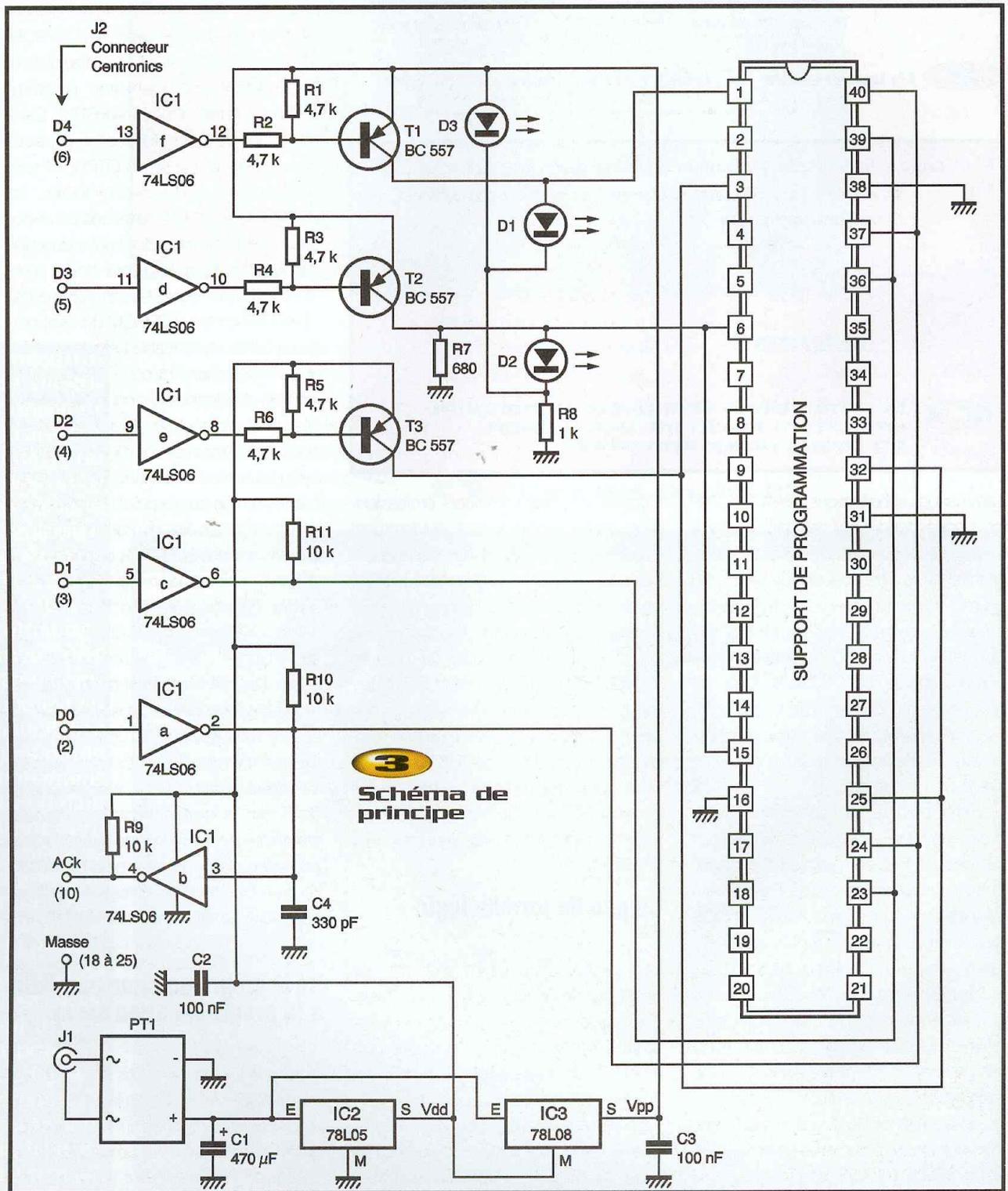
Alors que les mémoires les plus connues que sont les UVEPROM, de la "vieille" 2716 aux récentes 27512 ou 271024, se programment en parallèle ; c'est à dire en appliquant simultanément à la mémoire

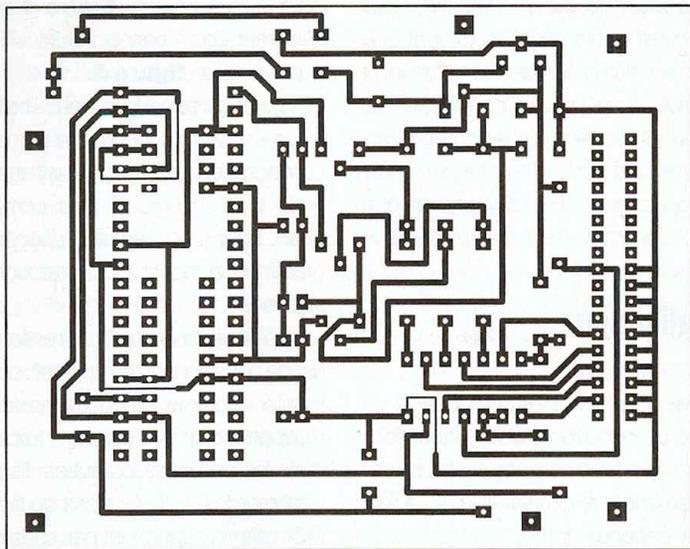
l'adresse à programmer et la donnée à placer à cette adresse ; de nombreuses mémoires récentes contenues dans les microcontrôleurs se programment en série. La **figure 2** montre la différence fondamentale entre ces deux procédés et l'on constate que, dans le cas de la programmation série, trois lignes de signaux seulement sont nécessaires contre plus d'une

dizaine en programmation parallèle (jusqu'à 26 même pour une 271024 qui est une mémoire de 1 M octets et qui nécessite donc 8 lignes de données et 17 lignes d'adresses). Le gain en encombrement est évident et la simplification du dessin du circuit imprimé qui en résulte est tout aussi flagrante.

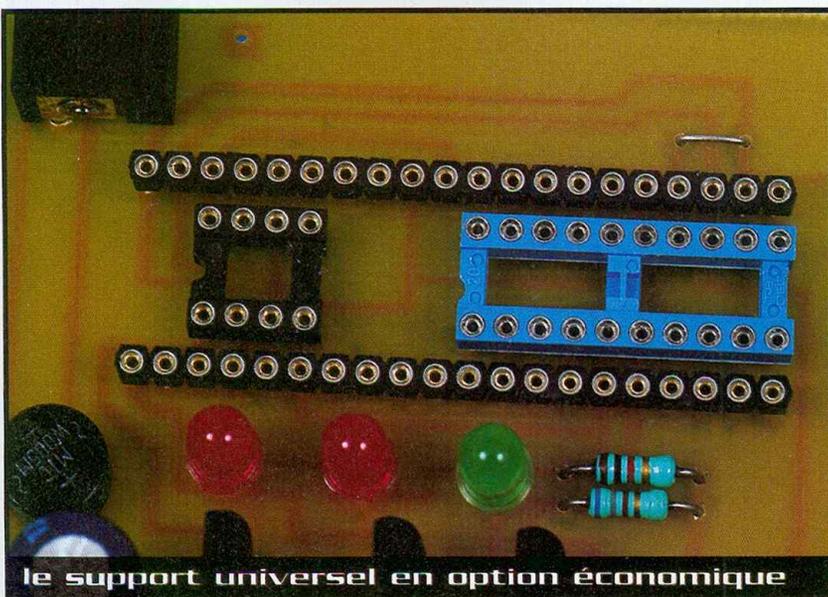
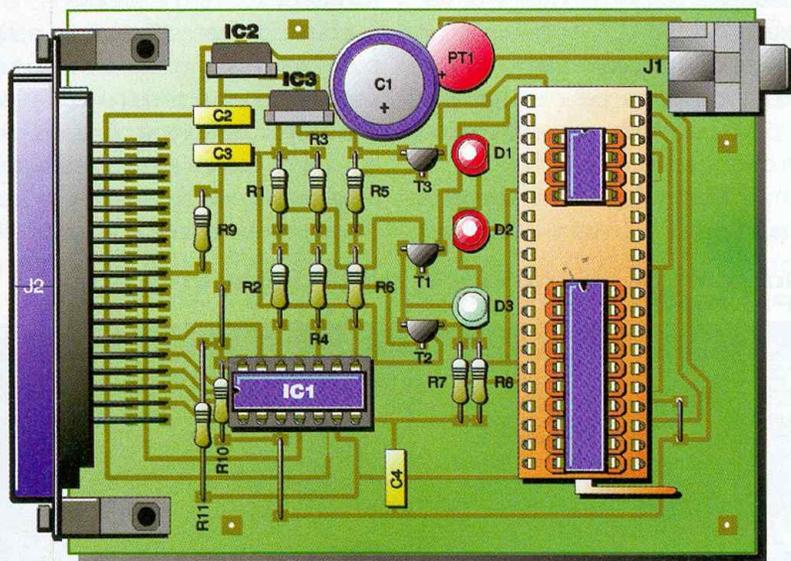
Les microcontrôleurs PIC de MICROCHIP

sont tous munis de mémoire à accès série et, comme certains boîtiers n'ont que 8 pattes, celles-ci sont partagées selon les modes de fonctionnement. Le **tableau 1** montre ainsi, dans le cas du 12C508 qui est un boîtier 8 pattes choisi à titre d'exemple, comment se passe la programmation au niveau des connexions du circuit. En pratique, trois pattes du boîtier





**4/5** Tracé du circuit imprimé et implantation des éléments



le support universel en option économique

changent momentanément de fonction pendant la phase de programmation pour donner accès à la mémoire de programme interne ; ce changement étant tout simplement déclenché par l'application de la "haute" tension de programmation sur la patte VPP.

**Le schéma de notre programmeur**

Même si les mémoires des PIC sont à programmation série, notre programmeur se connecte sur le port parallèle du PC. En effet, d'une part ce port se laisse très facilement piloter par logiciel, d'autre part il délivre des niveaux TTL directement exploitables. En outre, il nous faut disposer de quelques lignes de contrôle pour commuter les diverses alimentations du microcontrôleur en cours de programmation ce qui est beaucoup plus facile à réaliser sur un port parallèle que sur un port série.

Le schéma complet de notre programmeur est présenté **figure 3** et nous allons voir qu'il se laisse facilement analyser. Les signaux d'un port parallèle sont des signaux TTL et, de ce fait, sont assez "maltraités" par leur voyage sur des câbles de liaison un peu longs ou de piètre qualité. De ce fait, ils sont quelque peu remis en forme par les inverseurs contenus dans le circuit IC<sub>1</sub>. De plus, comme ce circuit dispose de sorties à collecteurs ouverts, il permet une commande facile des trois transistors T<sub>1</sub>, T<sub>2</sub> et T<sub>3</sub> qui font suite. T<sub>1</sub> et T<sub>2</sub> permettent l'application de la haute tension de programmation VPP aux pattes adéquates du support universel du programmeur ; pattes qui diffèrent selon le type de PIC programmé. On ne peut pas en effet espérer disposer du même brochage sur un boîtier DIL 8 pattes et sur un DIL 40 pattes !

Le transistor T<sub>3</sub>, quant à lui, commande la tension d'alimentation normale VDD, appliquée également au support universel. Ceci permet de n'alimenter le circuit à programmer que lorsqu'il est vraiment nécessaire d'y accéder et d'éviter ainsi tout problème lors de son insertion ou de son extraction du support de programmation.

Pour indiquer l'application ou non de ces tensions, deux LED rouges sont prévues, D<sub>1</sub> et D<sub>2</sub>, commandées par les deux tensions VPP. La diode D<sub>3</sub>, quant à elle, s'allume tout simplement lorsque le program-

mateur est sous tension afin de signaler le bon fonctionnement de l'alimentation.

Les données à programmer dans le circuit transitent par la porte IC<sub>1a</sub> tandis qu'elles passent par IC<sub>1b</sub> lors d'une relecture du circuit. L'horloge de programmation, quant à elle, passe par IC<sub>1c</sub>.

Le support destiné à recevoir les circuits à programmer est un modèle 40 pattes un peu spécial, comme nous le verrons lors de la réalisation pratique, de façon à pouvoir recevoir les PIC en boîtiers : 8, 18, 20, 28 et 40 pattes. Le câblage des différentes alimentations, de la ligne de données et de la ligne d'horloge est réalisé conformément aux brochages des différents circuits.

L'alimentation du programmeur est fort simple mais très tolérante. Deux tensions stabilisées sont en effet nécessaires : 5V pour l'alimentation normale ou VDD et 13V pour la haute tension de programmation ou VPP. IC<sub>2</sub> qui est un 78L05 se charge de la production du 5V tandis que IC<sub>3</sub>, qui est un 78L08, produit le 13V. Il n'est en effet pas référencé à la masse comme le veut la coutume mais à la sortie de IC<sub>2</sub> et produit ainsi 5 + 8 soit 13V !

Afin de s'accommoder de n'importe quelle source externe, nos régulateurs sont précédés d'un filtrage généreux et d'un pont redresseur. De ce fait, on peut appliquer sur l'entrée J1 toute tension alternative comprise entre 12 et 20V ou toute tension continue comprise entre 16 et 30V, sachant que le débit nécessaire n'est que de 100 mA environ.

### La réalisation

L'approvisionnement des composants ne doit vous poser aucun problème toutefois, avant de vous précipiter chez votre revendeur avec la liste à la main, lisez cet article en entier car deux options sont possibles au niveau du support de programmation, selon l'usage que vous ferez du programmeur et le budget que vous y consacrerez.

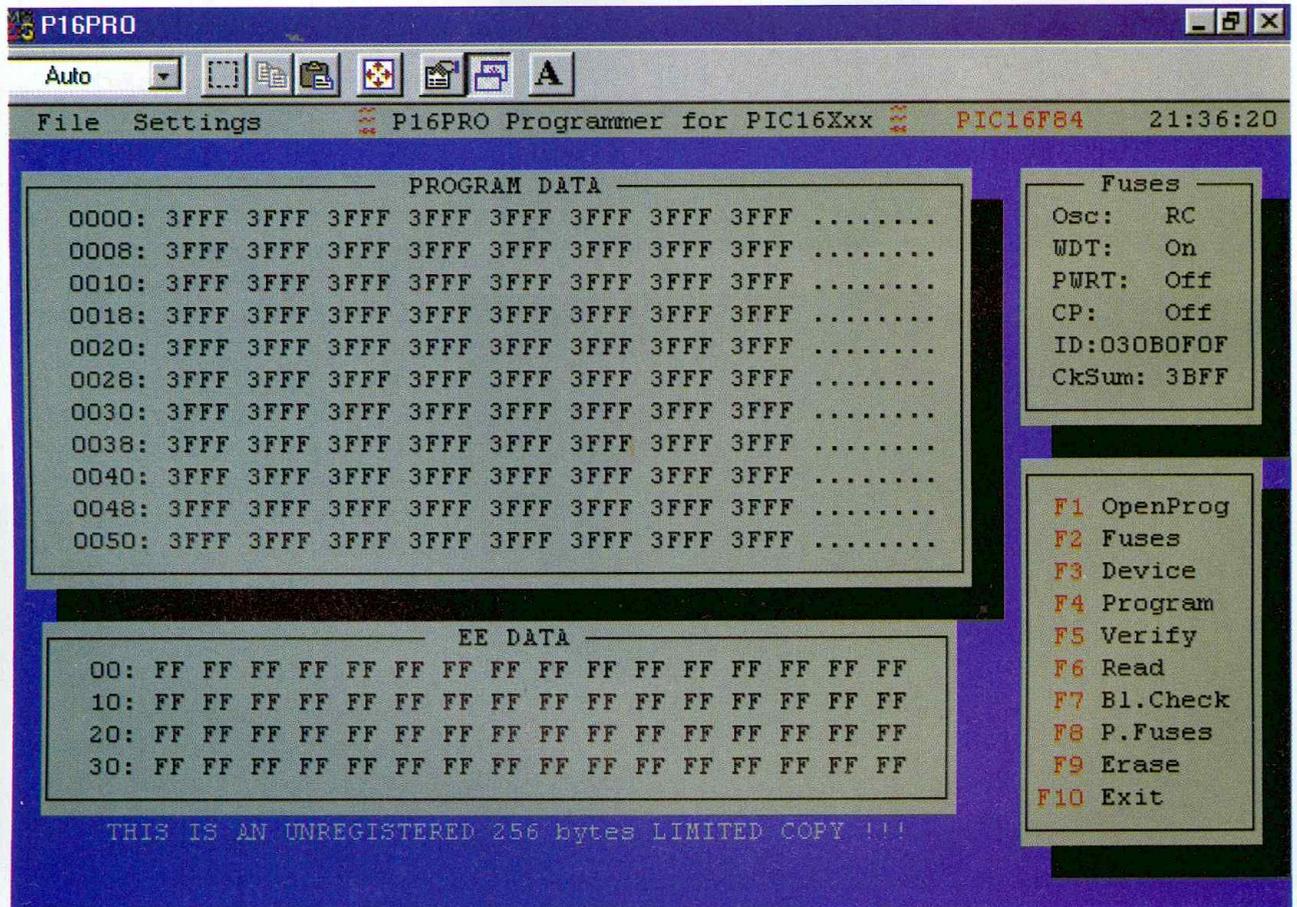
Le circuit imprimé que nous avons dessiné supporte tous les composants du schéma, connecteur Centronics normalisé compris, afin que vous puissiez brancher très simplement le montage en lieu et place de votre imprimante. Son tracé vous est pro-

posé à l'échelle 1 en **figure 4** et le plan d'implantation correspondant se trouve, quant à lui, en **figure 5**.

Le montage est à faire dans l'ordre classique à savoir : supports de composants, composants passifs, en faisant attention au sens des chimiques, puis composants actifs. Pour ce qui est du support de programmation, nous vous proposons deux solutions :

- la plus économique, à réserver à un usage occasionnel de l'appareil, consiste à monter, comme nous l'avons fait sur la maquette, plusieurs supports tulipes imbriqués les uns dans les autres. Ils peuvent ainsi recevoir tous les types de boîtiers de PIC mais ne supportent pas d'insertions et d'extractions de circuits répétées et fréquentes, d'où le qualificatif d'usage occasionnel que nous avons employé.
- La plus efficace consiste à utiliser un support à force d'insertion nulle, et non un support à insertion nulle, ce qui ne veut rien dire bien que nous le lisions régulièrement sous la plume de certains auteurs ! Ce support est encore appelé support ZIF pour Zéro Insertion Force. Mais attention, il ne faut pas

### **6** L'écran principal du logiciel de commande du programmeur, ici dans le cas d'un PIC 16F84.



utiliser n'importe quel modèle. En effet, les supports ZIF à 40 pattes standards ne peuvent pas recevoir les circuits DIL en boîtier 8, 18 et 20 pattes car ils sont trop "étroits" par rapport aux boîtiers à 28 et 40 pattes. Il faut donc utiliser un support ZIF de la marque 3M référence 240-6182-00-0605. Ce support coûte hélas à lui seul aussi cher que tous les autres composants du programmeur. Par contre, sa durée de vie très importante autorise un usage intensif de notre montage.

Si vous utilisez ce support ZIF, vous le soudez sur l'emplacement à 40 pattes prévu à cet effet. Dans le cas contraire, vous soudez un support tulipe 20 pattes en partie centrale basse de l'emplacement à 40 pattes ; un support tulipe 8 pattes dans la partie centrale haute de ce même emplacement et enfin deux bandes à souder à contacts tulipes de 20 pattes chacune de part et d'autre de ces deux supports. Vous obtiendrez alors la même disposition que celle que vous pouvez voir sur la photo de notre maquette.

Une fois le montage soigneusement contrôlé, vous pourrez passer aux essais et à la première utilisation ce qui nécessite évidemment de disposer du programme de pilotage adéquat.

## Logiciel et utilisation

Compte tenu du schéma que nous avons adopté et du succès des PIC évoqué en introduction, de nombreux logiciels sont utilisables avec notre programmeur et sont

disponibles sur Internet. Certains d'entre eux étaient encore disponibles en versions complètes il y a seulement quelques mois selon le principe du shareware mais, devant le nombre croissant de malhonnêtetés constatées, puisque certains distributeurs sont même allés jusqu'à revendre avec bénéfice de tels logiciels ; on ne trouve plus aujourd'hui que des versions "bridées".

Nous avons retenu pour notre part le programme P16PRO de Bojan Dobaj que vous trouverez sur notre site Internet. Ce logiciel est parfaitement adapté à notre montage et, la seule restriction de sa version bridée, est qu'elle ne peut programmer que les 256 premiers octets de la mémoire du circuit choisi. Le fait d'acquiescer la licence (pour 20 dollars soit environ 120 francs) permet de lever cette limitation et de disposer du produit complet et de ses mises à jour via Internet.

Le programme disponible s'appelle P16PR356.ZIP. Il doit être décompacté avec n'importe quel décompresseur ZIP dans le répertoire de votre choix. Un certain nombre de fichiers sont alors disponibles dont l'exécutable P16PRO.EXE et un fichier texte, REGISTER.TXT, qui explique comment acquiescer la licence. D'autres fichiers, que nous avons laissés en place conformément au principe du shareware, présentent des schémas de programmeurs similaires au nôtre et des dessins de circuits imprimés notablement différents. Ces fichiers sont au format PCX ou PDF selon le cas et peuvent être consultés avec n'importe quel logiciel de dessin supportant

le format PCX pour les premiers et avec Adobe Acrobat Reader pour les fichiers PDF.

Avant de lancer votre P16PRO fraîchement installé, commencez par raccorder votre programmeur au port parallèle de votre PC ; soit à la place de votre imprimante, soit sur un port parallèle supplémentaire (on trouve des cartes port parallèle supplémentaire pour 100 francs environ chez les revendeurs bien achalandés).

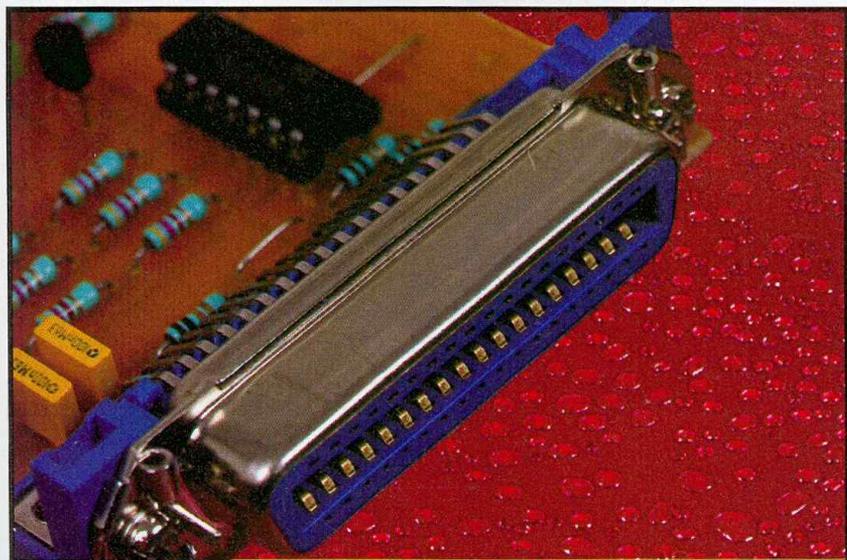
Reliez le programmeur à une alimentation conforme aux caractéristiques vues ci-dessus et vérifiez la présence de 5V en sortie de IC<sub>2</sub> et de 13V en sortie de IC<sub>3</sub>. La LED verte doit être allumée et les LED rouges peuvent être allumées ou éteintes selon ce que vous avez fait au préalable avec votre port parallèle.

Lancez alors le programme P16PRO. S'agissant d'un programme DOS, il se lance soit sous DOS, soit en mode DOS si vous travaillez avec Windows 95 ou 98, soit encore dans une fenêtre DOS toujours dans ce dernier cas.

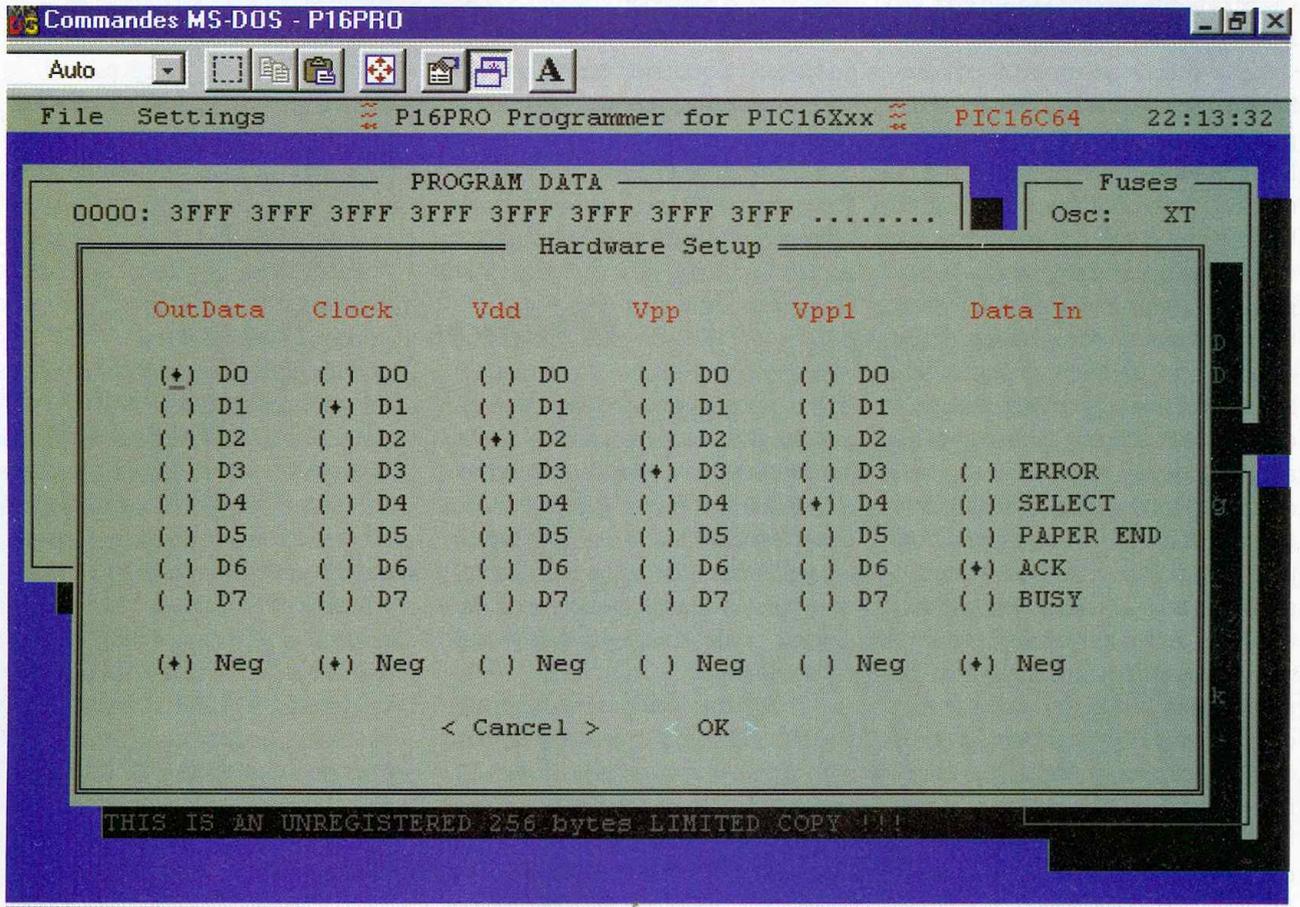
Quel que soit son mode de lancement, un écran s'affiche indiquant qu'il s'agit d'une version shareware (sauf après avoir acquis la licence) puis on accède à l'écran principal reproduit **figure 6**. Ce dernier affiche par défaut trois ou quatre fenêtres (selon le type de circuit PIC choisi). La fenêtre principale affiche le contenu de la mémoire de programme, la fenêtre secondaire l'état des "fusibles" de configuration du circuit et, selon le cas, une troisième fenêtre affiche le contenu de la mémoire de données pour les PIC munis d'une EEPROM de données. Une dernière fenêtre enfin résume les principales fonctions accessibles immédiatement au moyen des touches F1 à F10 du clavier.

Deux menus déroulants accessibles en haut et à gauche de l'écran en actionnant la touche ALT donnent accès à tout ce qui concerne les fichiers (menu "File") et à la configuration matérielle (menu "Settings"). Pour ouvrir l'un de ces menus, il suffit de frapper "Entrée" lorsque celui de votre choix est en vidéo inversée.

La première opération à réaliser consiste à paramétrer le logiciel en fonction de votre programmeur. Allez pour cela dans le menu "Settings" et descendez sur "Other" au moyen des touches fléchées de déplacement du curseur. Vous pouvez alors



le connecteur CENTRONICS femelle coudé à 90°



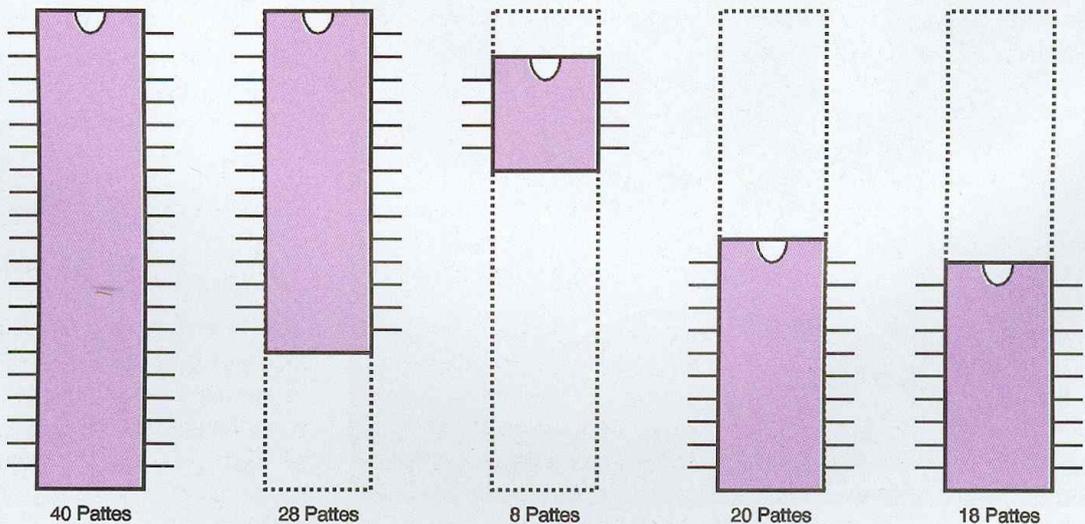
**7** L'écran de configuration du logiciel que vous devez obtenir.

sélectionner le port parallèle sur lequel se trouve connecté votre montage. Pour cela, déplacez le curseur clignotant sur le nom du port approprié et faites-le passer d'inactif à actif au moyen de la barre d'espace. Validez par OK puis revenez dans "Settings" mais allez cette fois-ci dans "Hardware" et vérifiez qu'une fenêtre identique à celle de

la **figure 7** s'affiche. Dans le cas contraire, modifiez votre fenêtre en conséquence (même principe que pour le choix du port ci-dessus) pour obtenir le même affichage que notre figure 7 faute de quoi votre programmeur ne pourra pas fonctionner. Vous pouvez ensuite placer un circuit PIC choisi parmi la liste des références suppor-

tées par le programmeur sur le support en respectant les indications de la **figure 8**. Sélectionnez la référence de votre circuit au moyen de la touche F3 puis essayez les diverses fonctions du programmeur. Si vous utilisez un circuit effaçable, un 16C84 ou 16F84 par exemple, vous pouvez même tenter une programmation "pour voir"

**8** Mode de mise en place des circuits en fonction de leur taille sur le support universel.



puisqu'il vous sera toujours possible de l'effacer ensuite.

Notez que pendant les phases où le circuit est alimenté, l'une au moins des LED rouges s'allume ce qui indique évidemment que le circuit ne doit pas être retiré de son support à cet instant sous peine de destruction.

L'encadré présente un résumé du mode

d'emploi de ce programmeur dont les principales fonctions sont toutefois suffisamment explicites pour que vous n'éprouviez aucune difficulté à les utiliser. Si tel était le cas, sachez qu'une notice en langue anglaise se trouve dans le fichier SMA-NUAL.ENG créé lors de la décompression du fichier initial.

Un problème de fonctionnement est très

improbable vu la simplicité du montage et les seules difficultés que vous pourriez éventuellement rencontrer sont des erreurs de programmation ou de lecture lors de l'utilisation de câbles de liaison de plus de 3 m entre le PC et le programmeur. Si donc vous constatez un tel phénomène, qui se manifeste par des changements plus ou moins aléatoires des données lues ou programmées, raccourcissez le câble de liaison ou utilisez un câble de meilleure qualité.

## Nomenclature

IC<sub>1</sub> : 7406 ou 74LS06

IC<sub>2</sub> : 78L05

IC<sub>3</sub> : 78L08

PT<sub>1</sub> : pont moulé 100V/1A

D<sub>1</sub>, D<sub>2</sub> : LED rouges 5 mm

D<sub>3</sub> : LED verte 5 mm

T<sub>1</sub> à T<sub>3</sub> : BC557, 558, 559

R<sub>1</sub> à R<sub>6</sub> : 4,7 kΩ 1/2 ou 1/4W 5%  
(jaune, violet, rouge)

R<sub>7</sub> : 680 Ω 1/2 ou 1/4W 5%  
(bleu, gris, marron)

R<sub>8</sub> : 1 kΩ 1/2 ou 1/4W 5%  
(marron, noir, rouge)

R<sub>9</sub> à R<sub>11</sub> : 10 kΩ 1/2 ou 1/4W 5%  
(marron, noir, orange)

C<sub>1</sub> : 470 µF/35V chimique radial

C<sub>2</sub>, C<sub>3</sub> : 100 nF mylar

C<sub>4</sub> : 330 pF céramique

J<sub>1</sub> : jack mâle 2,1 mm pour circuit imprimé

J<sub>2</sub> : connecteur Centronics femelle soudé à 90 ° pour circuit imprimé

1 support de CI 14 pattes

1 support de programmation 8 pattes, option économique, contacts tulipes

1 support de programmation 20 pattes, option économique, contacts tulipes

2 bandes à souder 40 pattes, option économique, contacts tulipes

1 support de programmation, option "pro" 3M (Textool) référence

240-6182-00-0605

## Conclusion

Pour un investissement très faible, vous voici équipé d'un programmeur capable de traiter la majorité des microcontrôleurs PIC actuels avec un maximum de confort. Qui plus est, il se contente d'un PC quelconque ; de quoi redonner vie par exemple à votre vieil AT 286 qui dort couvert de poussière dans votre grenier...

C. TAVERNIER

## Mode d'emploi résumé du programmeur

Les touches F1 à F10 donnent directement accès aux fonctions suivantes :

F1 : sélectionne le fichier à utiliser dans le répertoire courant.

F2 : donne accès à la fenêtre de définition des "fusibles" de configuration du circuit à programmer mais n'effectue pas leur programmation.

F3 : permet de choisir le type de circuit PIC à programmer.

F4 : programme le circuit PIC choisi par F3, dont le type est rappelé en haut à droite de l'écran, avec le fichier préalablement sélectionné par F1 ou par le menu "File".

F5 : compare le contenu du PIC placé sur le support de programmation avec le fichier préalablement sélectionné par F1 ou par le menu "File".

F6 : lit le contenu du PIC placé sur le support de programmation et l'affiche dans la ou les fenêtres correspondantes.

F7 : effectue un test de virginité du PIC.

F8 : programme seulement les "fusibles" de configuration du PIC conformément à l'état défini préalablement au moyen de F2 ; état qui est rappelé en permanence dans la fenêtre "Fuses".

F9 : efface le PIC si celui-ci est effaçable (versions munies de mémoire EEPROM). Si le circuit choisi au moyen de la touche F3 n'est pas effaçable électriquement, cette commande est grisée et inaccessible.

F10 : permet de sortir du programme.

Les deux menus "File" et "Settings" sont accessibles par action sur la touche ALT du clavier puis par déplacement au moyen des touches fléchées du curseur. Ils s'ouvrent par action sur la touche "Entrée". Les rubriques qui apparaissent alors sont accessibles soit en se déplaçant dessus au moyen des flèches montée et descente du curseur, soit directement en frappant la lettre repérée en surbrillance (ou vidéo inversée selon le mode d'affichage). Par exemple, la rubrique "Hardware" du menu "Settings" est accessible en frappant directement H lorsque ce menu est ouvert.

La sortie d'une fenêtre, ouverte par l'une de ces rubriques, peut se faire en déplaçant le curseur sur "Cancel" pour annuler les modifications éventuellement faites par erreur ; sur "OK" pour valider les choix faits dans la fenêtre ; ou bien encore sur "Escape" pour quitter la fenêtre sans intervenir dessus.

Lorsqu'une fenêtre contient des cases à cocher comme c'est le cas pour la fonction permettant de choisir le port parallèle par exemple ; il suffit de se déplacer sur la case de votre choix au moyen des touches fléchées de déplacement du curseur pour pouvoir mettre en place ou enlever la "coche" au moyen de la barre d'espace.

Ceci étant précisé, le menu "File" donne accès aux fonctions suivantes :

- Open Program : joue le même rôle que la touche F1 et permet de charger dans la mémoire de programme le contenu du fichier sélectionné. Le logiciel reconnaît les formats standards de l'assembleur MICROCHIP à savoir INH8M ou INH16.
  - Save Program : sauvegarde le contenu de la mémoire de programme dans le fichier de votre choix.
  - Open Data et Save Data : jouent le même rôle que les deux commandes que nous venons de voir mais pour les circuits contenant de la mémoire EEPROM de données. Dans le cas contraire, ces commandes sont en grisé et sont inaccessibles.
  - Edit Program : permet d'éditer le contenu de la mémoire de programme courante. L'adresse et le contenu désirés doivent être frappés en hexadécimal. Une fois qu'une adresse a été indiquée, le fait de frapper "Entrée" fait automatiquement passer à l'adresse suivante. Pour sortir de ce mode, il suffit de frapper sur la touche "Escape".
  - Fill Program : permet de remplir une zone de la mémoire de programme avec la donnée de votre choix. Toutes les entrées se font en hexadécimal.
  - Edit Data et Fill Data : jouent le même rôle que les deux commandes que nous venons de voir mais pour les circuits contenant de la mémoire EEPROM de données. Dans le cas contraire, ces commandes sont en grisé et sont inaccessibles.
  - Clear Buffer : remet à l'état vierge (00 ou FF selon le cas) toute la mémoire de programme ainsi que la mémoire EEPROM de données pour les circuits qui en contiennent.
  - About : affiche le copyright du programme et son numéro de version.
  - Exit : permet de sortir du programme comme la touche F10.
- Le menu "Settings", quant à lui, donne accès aux fonctions suivantes :
- Device : permet de choisir le type de microcontrôleur comme la touche F3.
  - Fuses : permet de définir l'état des fusibles comme la touche F2.
  - ID : permet de définir l'état du mot d'identification contenu dans les circuits PIC. Ce mot peut être utilisé comme somme de contrôle, comme numéro de série ou bien être ignoré, à votre choix.
  - Hardware : donne accès à la fenêtre de définition des différentes lignes de connexion du programmeur sur le PC. Cette fenêtre doit être rendue conforme à celle que nous vous présentons en figure 7 et ne doit pas être modifiée sous peine de ne pas pouvoir faire fonctionner le programmeur.
  - Other : permet de choisir le port parallèle utilisé ainsi que diverses options comme la sauvegarde de la configuration en quittant le programme, les fenêtres de visualisation affichées, etc.
  - Save : permet de sauvegarder toute la configuration courante, y compris le type de micro choisi, dans le fichier P16PRO.INI en quittant le programme afin de la retrouver au lancement suivant.

# 68HC11

## Assembleur, Basic, C, Débogueur

Basic11: compilateur croisé. Le programme compilé est rapide et petit. Pour tous les 68HC11 même avec peu de mémoire. Manuel, exemples. CC11: compilateur croisé ANSI C. Din: Débogueur pour les EEPROM internes 68HC11A1, E1, E2, F1... jusqu'à 8k. Dex: Débogueur non limité. Pour Windows 3.1/95/98/NT.

BADin	Basic11 + Assembleur + Débogueur interne	<b>540 F</b>
BADex	Basic11 + Assembleur + Débogueur	<b>1990 F</b>
CADin	CC11 + Assembleur + Débogueur interne	<b>980 F</b>
CADex	CC11 + Assembleur + Débogueur	<b>2380 F</b>
BCADin	Basic11 + CC11 + Ass + Débogueur intern	<b>1280 F</b>
BCADex	Basic11 + CC11 + Ass + Débogueur	<b>2680 F</b>

Documentation complète dans l'aide de Windows. Un an de mise à jour gratuite par Internet. Version Freeware limité à 100 octets avec documentation complète dans l'aide gratuit sur Internet.

## Controlboy - Starter Kit 68HC11

Carte montée, entrées numériques et analogiques, sorties numériques, câble PC, manuel d'installation. Programmation en assembleur et en prototypage rapide sous Windows 3.1/95/98/NT.

KitF1	Controlboy F1 12MHz 32k EEPROM, 32k RAM	<b>1688 F</b>
+opt	3 sorties anal., LCD, clavier, 8 entrées optoc.	<b>+482 F</b>
+Baboy	Basic11 pour Controlboy	<b>+300 F</b>
+CAboy	CC11 pour Controlboy	<b>+680 F</b>

CONTROLORD, 485. Av. Guils, 83210 La Farlède 0494487174

**www.controlord.fr**

Basic 11 5.3 Freeware et 68HC11 doc  
sur le CD-ROM Eprat 238

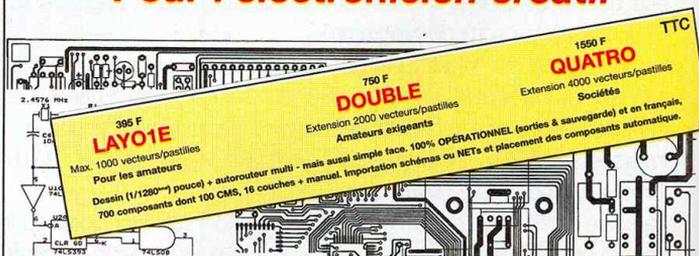
Un petit circuit en une demi-heure, un plus complexe en une matinée... à partir de 195 F TTC seulement

## L'AUTOROUTEUR LAYO... C'EST ÇA !

Comme le confirment 30 000 amateurs en France... quelques milliers de pros qui ne touchent plus que rarement leur superlogiciel précédent, ainsi que :

PRESIDENCE DE LA REPUBLIQUE, HOTEL MATIGNON, MINISTERES, PARLEMENT EUROPEEN, OTAN, LES TROIS ARMEES, DASSAULT, IBM, AEROSPATIALE, EDF, LES CENTRALES NUCLEAIRES, TELECOM, RATP, CITROEN, PEUGEOT, RENAULT, SAGEM, MOTOROLA, COMPAQ, PHILIPS, TEXAS INSTRUMENTS, CERN, CNRS, TEFAL, SOC, AUTOROUTES, INSTITUT PASTEUR, THOMSON CSF, CEA, COGEMA, SNCF, POSTE, ELF, RHONE-POULENC, ROCOCH, ROCKWELL, STAR, GRUNDIG, IFREMER, SATEL, ALCATEL, MATRA, 3M, AFPA, TDF, MERLIN, NUCLETUDE, COGETUDE, CANAL +, TF1, FR3, RMO, GENDARMERIE, AIR LIQUIDE, INSA, SEITA, TRANSPORTS, AEROPORTS, 90% DES UNIVERSITES et IUT, 85% DES LYCEES ET COLLEGES etc.

**Pour l'électronicien créatif**



## Layo visualiseur W 95/98

Visualiseur de tous les LMC et/ou PLY  
instantanément dans une deuxième fenêtre

Layo France Sarl, Château Garamache  
Sauvebonne 83400 Hyères

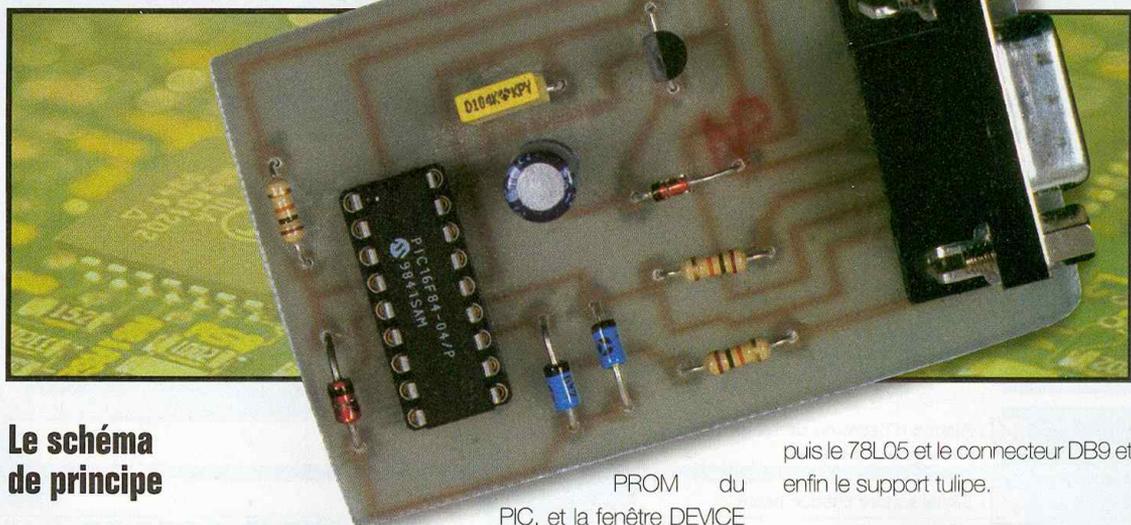
Tél.: 04.94.28.22.59 - Fax : 04.94.48.22.16

<http://www.layo.com>

[layo@layo.com](mailto:layo@layo.com)

# Programmateur de PIC16C84 et PIC16F84

par le port série  
du PC



**Le programmateur que nous vous proposons est sûrement le plus petit en taille et en prix que l'on puisse trouver ! En effet, il ne se compose que de 10 petits composants passifs, pas d'alimentation extérieure et permet de programmer le PIC16C84 et PIC16F84 en quelques secondes.**

## Le schéma de principe

Le montage est alimenté à l'aide des lignes 12V du port parallèle. Une diode 1N4148 ne laisse passer que les tensions entre 0/12V, puis la tension est ramenée à 5V grâce au 2 condensateurs et au 78L05 qui est un régulateur 5V.

Puis finalement, les 3 diodes zéner servent à arranger les signaux d'horloge et de données. Le reste est un problème de logiciel.

## Le logiciel

Avant tout, si vous décompressez le fichier sous DOS, utilisez le paramètre -d (PKUNZIP -d) afin que les sous-répertoires soient créés. C'est en effet lui qui se charge de tout. Il est très complet et très maniable. En effet, il permet de programmer, lire, vérifier et tester le PIC. De plus, il accepte les formats INHX8M, INHX16, INHX32, Raw binary. Il se compose de 2 buffers et d'une fenêtre Device.

Le premier buffer contient le programme, le second contient l'EE-

PROM du PIC, et la fenêtre DEVICE contient les informations telles que le type d'oscillateur choisi ou si le chien de garde doit être actif ou bien si le code doit être protégé...

Il est disponible gratuitement sur INTERNET et s'appelle PIP-02. Il est donc en téléchargement sur le site eprat.com

Il est nécessaire de charger préalablement un driver fourni dans le "zip" (com84.exe) avant de lancer le logiciel et après l'avoir lancé afin de le désactiver. Pour cela, il est conseillé d'utiliser un fichier .BAT.

## La réalisation

Il convient de percer tous les trous avec un foret de 0,8 mm exceptés les deux trous d'attaches du connecteur DB9 qui doivent être percés en 1 mm et agrandis en largeur afin de laisser passer les pattes d'attache du connecteur. Il est conseillé de prendre un support tulipe.

On soudera d'abord les résistances et les diodes, les condensateurs,

puis le 78L05 et le connecteur DB9 et enfin le support tulipe.

## Les essais

Branchez le programmateur muni d'un PIC16C84 au port parallèle du PC à l'aide d'un câble non croisé. Écrivez un fichier texte "PROGRAM.BAT" et inscrivez ce qui suit :

```
:LABEL1
COM84 COM1
IF ERRORLEVEL 200 GOTO LABEL1
:LABEL2
PIP-02
IF ERRORLEVEL 200 GOTO LABEL2
:LABEL3
COM84 REMOVE
IF ERRORLEVEL 200 GOTO LABEL3
```

Enfin, remplacez COM1 par le numéro de votre port si vous n'utilisez pas le COM1. Enregistrer et quittez l'éditeur.

Il est possible que sur les ordinateurs un peu rapides que l'on trouve de nos jours, apparaisse "ERROR 200" dû sûrement à une division par 0. C'est pourquoi des "IF ERRORLEVEL" ont été placés. Si le lancement se fait en

forme de boucle, une très bonne solution consiste à démarrer plusieurs applications (pourquoi pas 5 ou 10 fenêtres MS-DOS) en même temps afin de ralentir assez efficacement Windows.

Lancez PROGRAM.BAT. Si tout se passe bien, vous devez avoir un grand buffer bleu avec 3FFF partout. Allez dans SELECT puis DEVICE et choisissez PIC16C84 ou PIC16F84.

Faites ALT-F6 ou DEVICE>ERASE afin d'effacer la mémoire du PIC. Inscrivez des données dans le buffer ou éditez un fichier HEX pour PIC16C84.

Pour changer de fenêtre, il faut utiliser la combinaison de touche CTRL-F6

Faites F5 ou DEVICE>Program pour programmer le PIC.

Faites F6 ou DEVICE>Verify afin de vérifier si le programme présent dans le buffer est bien inscrit dans la mémoire du PIC. A ce propos, si le code est protégé en lecture et que vous tentez de faire une vérification, une erreur vous sera signalée étant donné que le PIC ne renverra pas les informations protégées en lecture.

Si aucune erreur n'est signalée, c'est donc que le montage fonctionne !

Si vous souhaitez changer manuellement le mode d'oscillation, allez dans SELECT>FUSE\_WORD et servez-vous de la touche TAB.

Les différentes possibilités sont :

- LP : Low Power avec un quartz de 32 kHz
- XT : Généralement le plus utilisé avec un quartz de 10 MHz
- HS : High Speed pour un quartz de 20 MHz
- RC : oscillateur RC

De plus, vous pouvez dans la même fenêtre désactiver ou activer :

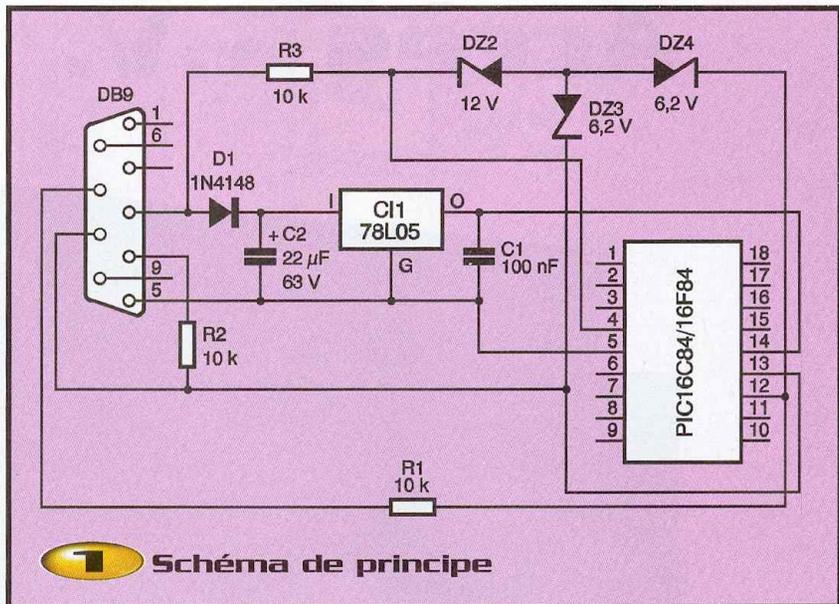
- Le chien de garde
- La mise en route du Timer
- La protection du code en lecture

Enfin, n'oubliez pas d'effacer la mémoire du PIC avant de le reprogrammer.

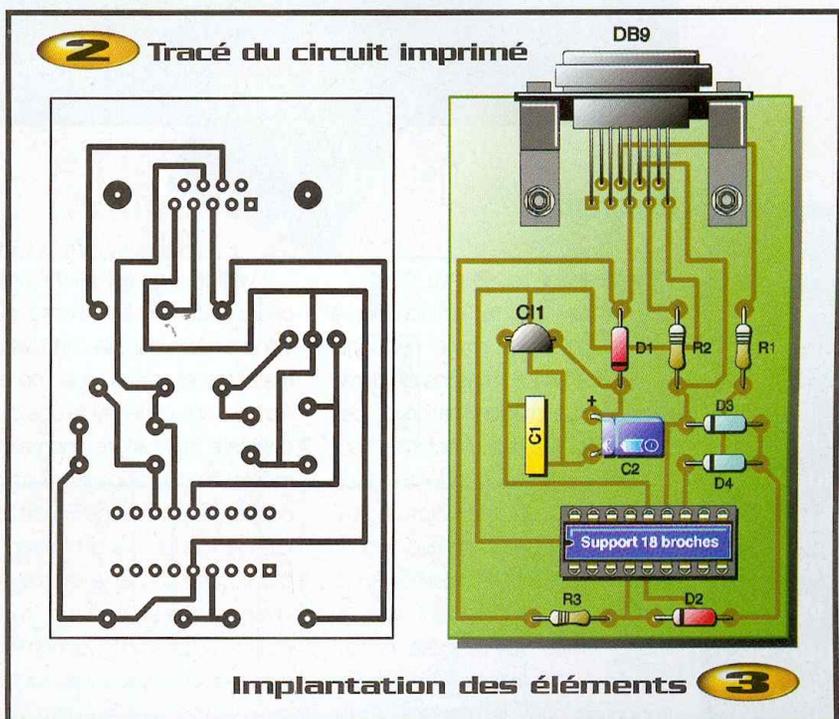
## Conclusion

Ce petit programmeur est de loin le plus simple et le plus économique à réaliser. Il pourra servir à programmer les microcontrôleurs PIC16C84 ou PIC16F84 qui sont utilisés dans beaucoup de montages.

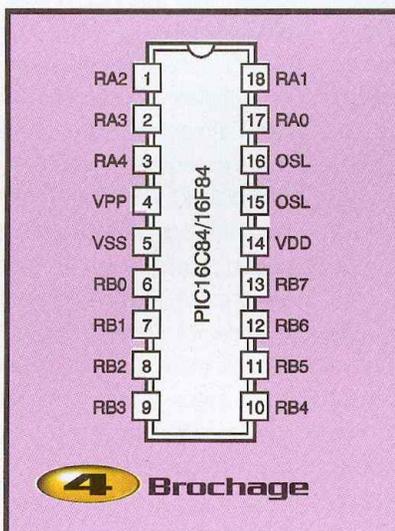
A. RAMTIN



**1** Schéma de principe



Implantation des éléments **3**



**4** Brochage

## Nomenclature

R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub> : 10 kΩ (marron, noir, orange)

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

D<sub>2</sub> : zéner 12V

D<sub>3</sub>, D<sub>4</sub> : zéner 6V2

C<sub>1</sub> : 100 nF céramique

C<sub>2</sub> : 22 µF/63V

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

1 support tulipe 18 broches

1 connecteur DB9

# Accès T.V.

## à la carte



### Description

La restriction d'accès va être déterminée par la distribution à chaque utilisateur d'une carte qui contiendra les informations de durée autorisée. La programmation de ces cartes sera réalisée directement sur le montage après introduction d'un code confidentiel. On pourra également générer une carte avec accès illimité permettant une utilisation continue de l'appareil restreint tout du moins temps que la carte restera insérée dans le montage. Lors de l'introduction dans le montage d'une carte à accès limité, l'appareil restreint sera alimenté et la carte sera décrétementée d'une unité toutes les minutes. A la fin du temps imparti, l'appareil restreint sera automatiquement mis hors tension. Pour poursuivre l'utilisation, il faudra remettre des unités sur la carte mais bien sûr au bon vouloir du détenteur du code confidentiel absolument indispensable pour programmer la carte. Le code d'accès, étant une séquence de touche, permet une sécurité tout à fait honorable surtout si l'on ne connaît pas la procédure complète. Toute tentative de fraude sera veine

puisqu'à chaque introduction, la carte est

vérifiée. Il en est de même après chaque décrétement d'unité où l'on viendra écrire dans la carte les unités restantes et vérifier que l'on a bien écrit sur une carte valide dès que l'on constate la moindre anomalie, l'alimentation de l'appareil restreint est coupée et le message Fin est affiché. Bien entendu, lors de l'introduction d'une carte neuve, le montage vient la reconnaître et l'écrit systématiquement avec zéro unité. Évidemment, la prise secteur équipant le montage ne sera pas d'un type standard mais plutôt une fiche type CEE femelle. L'appareil restreint se verra, lui aussi, équipé d'une fiche mâle car sinon notre limiteur d'accès ne servirait à rien. Le montage est réalisé autour d'un microcontrôleur MICROCHIP de la famille PIC, un 16C55. Ce microcontrôleur, compte tenu du fait que ce soit un produit de première génération chez MICROCHIP, présente l'intérêt de voir son prix plus qu'abordable (à peine quelques dizaines de francs mieux, moins de 6 Euros). Ce microcontrôleur va gérer la totalité des fonctions du montage.

### Fonctionnement du circuit

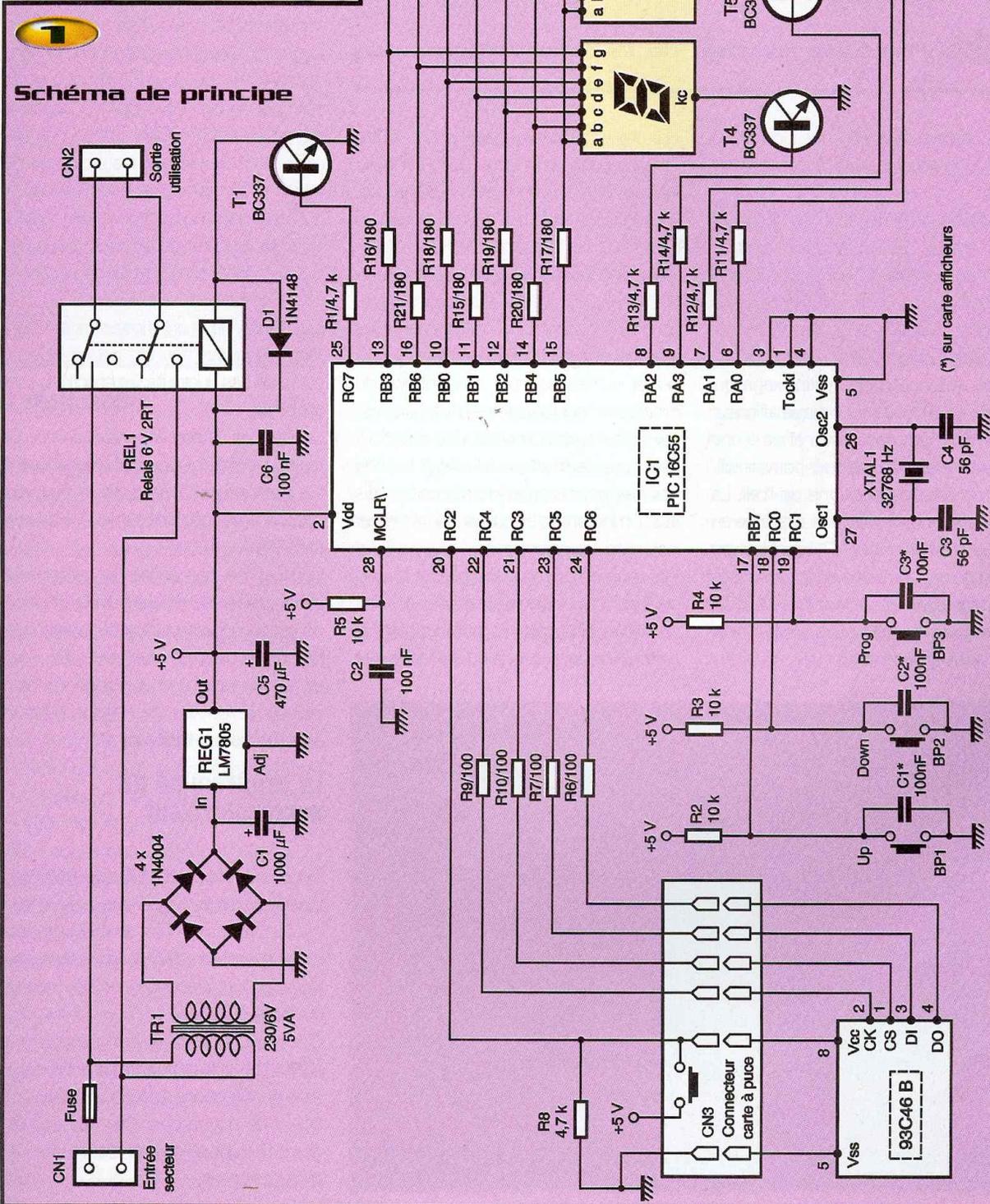
L'alimentation est réalisée grâce à un transformateur 230V/6V/5VA suivi d'un classique pont redresseur et d'un condensateur de filtrage. La régulation à la tension de 5V est confiée à un régulateur linéaire type 7805. Le choix du transformateur s'orientera vers un transformateur comportant une marque de conformité à la norme EN 60742 permettant de s'assurer d'une totale sécurité d'isolement primaire/secondaire qui rappelons-le est primordial pour la protection des personnes contre les risques d'électrocution, chose que certains ont tendance à oublier ou à laisser de côté.

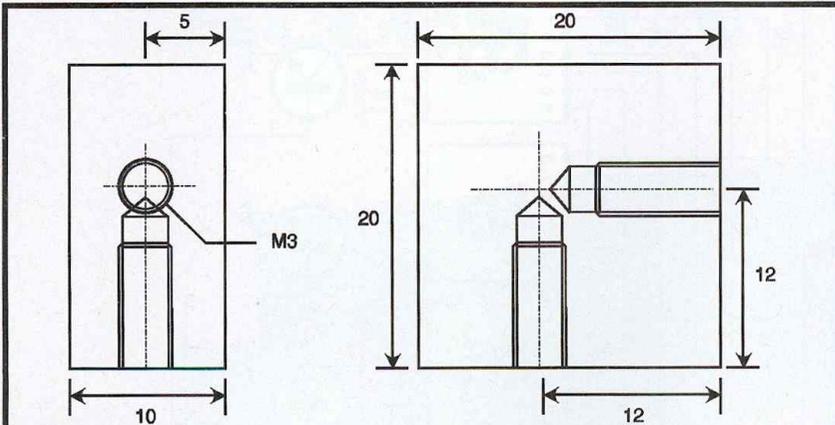
Le cœur du circuit est réalisé par un microcontrôleur PIC16C55 MICROCHIP comportant 20 entrées/sorties configurables à volonté. Son horloge est générée par un quartz de 32768 Hz. Le choix de cette fréquence est motivé par le fait que l'on a besoin de calculer ou plutôt de compter des temps en minutes donc multiple de la seconde donc égale à  $32768/2^{15}$  ou  $32768/(256 \times 128)$ . Le circuit de reset est simplement réalisé à l'aide d'un circuit

**Le montage proposé et intitulé «accès T.V. à la carte» est un moyen de restriction d'accès, soit total, soit limité dans le temps, de tout appareil relié au secteur. Il pourra non seulement servir à limiter le temps d'accès à la télévision ou jeu vidéo mais aussi servir lorsque l'on souhaitera limiter le temps de fonctionnement d'un quelconque appareil (par exemple, chargeur de batteries, etc.).**

RC permettant d'assurer la remise à zéro du microcontrôleur lors de la mise sous tension. Le microcontrôleur va se charger de l'affichage quatre digit permettant non seulement d'afficher le temps d'utilisation restant mais aussi de nous informer par des messages clairs de tout ce qui se passe, des boutons poussoirs nous permettant de dialoguer, de l'écriture et lecture de la carte utilisateur et enfin du relais de commande de l'appareil restreint.

## Schéma de principe





**2 Patte de fixation de la carte afficheurs**

Regardons maintenant chaque fonction un peut plus en détail en commençant par l'affichage.

- Les afficheurs utilisés sont des afficheurs 7 segments à LED à cathode commune et sont au nombre de 4. Pour piloter les 28 segments des quatre afficheurs, nous utiliserons 11 pattes de sorties du microcontrôleur. Pour ce faire, il est impératif d'utiliser la fonction de multiplexage qui consiste en fait à allumer chaque afficheur les uns à la suite des autres et ce à une fréquence assez élevée pour pouvoir utiliser la persistance rétinienne<sup>SM</sup> de l'œil. La cathode de chaque afficheur est reliée au collecteur d'un transistor NPN dont la base est pilotée par une sortie du microcontrôleur. Cette sortie sera donc mise à l'état haut lorsque l'on souhaitera allumer l'afficheur correspondant.

- Le clavier de commande, quant à lui, se compose de trois boutons poussoir directement montés sur circuit imprimé. On trouve un bouton pour la fonction UP, un bouton pour la fonction DOWN et un bouton pour la fonction PROG. Ces trois boutons sont reliés à trois entrées du microcontrôleur et sont actifs à l'état bas. Des résistances de Pull UP permettent de présenter un niveau haut aux entrées du microcontrôleur lorsque les boutons poussoir ne sont pas actionnés. Ces trois boutons nous permettrons de rentrer le code d'accès pour programmer la carte mais aussi d'incrémenter ou de décrémenter les unités à programmer.

- Le relais de commutation permet la mise sous ou hors tension de l'appareil restreint, son pouvoir de coupure sera donc choisi en fonction de la puissance de l'appareil à

commander. On utilisera un relais 2 RT de façon à couper la phase et le neutre. La bobine de ce relais est commandée par un transistor NPN dont la base est pilotée par une sortie du microcontrôleur. On n'oubliera pas la diode de roue libre sur la bobine du relais car on risquerait de voir la durée de vie du transistor de commande se réduire à peu de chose du fait de la surtension présente aux bornes de la bobine lors de la coupure d'alimentation de celle-ci.

- La carte d'accès est une partie importante du montage puisque se sera elle qui contiendra les droits d'accès à l'appareil restreint. Cette carte est simplement composée d'une mémoire EEPROM à accès série par un bus Microwire<sup>SM</sup>. Du fait de la technologie EEPROM cette mémoire conservera les informations programmées même si son alimentation est coupée ce qui, dans notre cas, est primordial puisque la carte utilisateur peut et doit être retirée lorsque l'on n'utilise pas l'appareil. Le bus Microwire<sup>SM</sup> permet d'accéder aux 64 mots de 16 bits de la mémoire simplement à l'aide de quatre fils. On pourra donc, à cause de ces quatre fils, lire et écrire la carte utilisateur.

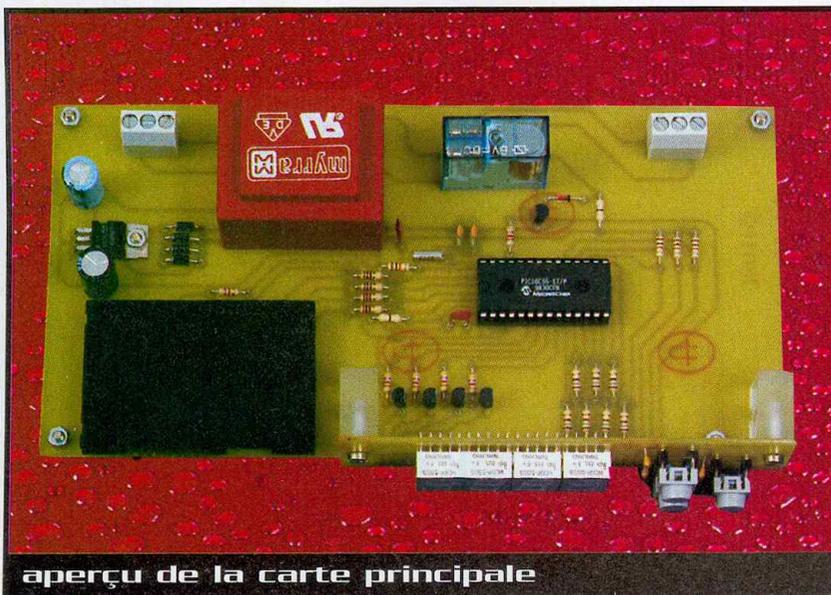
La mémoire utilisée est une 93C46B que l'on trouve chez plusieurs fabricants. Le suffixe B est important car il détermine l'organisation de la mémoire suffixe B = 64 x 16 et suffixe A = 128 x 8.

Voilà en ce qui concerne le détail de chaque élément. Voyons maintenant, avant de passer à la description du programme, la connexion du microcontrôleur. Pour cela un simple tableau nous permet de nous indiquer la fonction de chaque patte du microcontrôleur (**tableau 1**).

**Le programme du microcontrôleur**

Nous ne rentrerons pas dans le détail car il faudrait plusieurs pages et cela deviendrait vite fastidieux mais il est à noter que celui-ci a été écrit directement en assembleur dans le but d'économiser de l'espace mémoire.

Comme on pourra le constater dans le fichier .LST, il ne reste plus de place disponible dans l'espace mémoire du microcontrôleur ce qui ne permettra pas une très grande latitude si l'on souhaite modifier certaines fonctions.



**aperçu de la carte principale**

Port	Bit	N° Pin	Description	E/S
PORT A	D0	6	Commande de la base du transistor de la cathode de l'afficheur 4	S
PORT A	D1	7	Commande de la base du transistor de la cathode de l'afficheur 3	S
PORT A	D2	8	Commande de la base du transistor de la cathode de l'afficheur 1	S
PORT A	D3	9	Commande de la base du transistor de la cathode de l'afficheur 2	S
PORT B	D0	10	Anode du segment e des afficheurs	S
PORT B	D1	11	Anode du segment d des afficheurs	S
PORT B	D2	12	Anode du segment c des afficheurs	S
PORT B	D3	13	Anode du segment g des afficheurs	S
PORT B	D4	14	Anode du segment b des afficheurs	S
PORT B	D5	15	Anode du segment a des afficheurs	S
PORT B	D6	16	Anode du segment f des afficheurs	S
PORT B	D7	17	Bouton poussoir UP	E
PORT C	D0	18	Bouton poussoir DOWN	E
PORT C	D1	19	Bouton poussoir PROG	E
PORT C	D2	20	Switch connecteur carte à puce	E
PORT C	D3	21	CS de l'EEPROM 93C46	S
PORT C	D4	22	CK de l'EEPROM 93C46	S
PORT C	D5	23	DI de l'EEPROM 93C46	S
PORT C	D6	24	DO de l'EEPROM 93C46	E
PORT C	D7	25	Commande de la base du transistor du relais d'utilisation	S

## Tableau de fonctions

### La réalisation

Le montage se compose de trois cartes distinctes, une carte unité centrale qui supportera le microcontrôleur, le connecteur de cartes à puce, l'alimentation et le relais d'utilisation. Une carte afficheur qui, comme son nom le laisse supposer, comprendra les afficheurs sept segments mais également les boutons poussoirs de commande qui en fait sont des micro-contacts au format DIP6. Enfin un troisième circuit au format carte de crédit qui supportera la mémoire EEPROM contenant les informations liées aux conditions d'utilisation de l'appareil restreint par son porteur. Le nombre de ces cartes supportant l'EEPROM n'est pas limité et il faudra envisager d'en réaliser une pour chaque personne ayant accès à l'appareil restreint.

La réalisation des circuits imprimés ne pose aucun problème mais il faudra tout de même vérifier que les distances entre les pistes reliées au secteur et celles du reste du circuit ne sont pas inférieures à 7 mm dans le but de garantir une totale sécurité pour les utilisateurs.

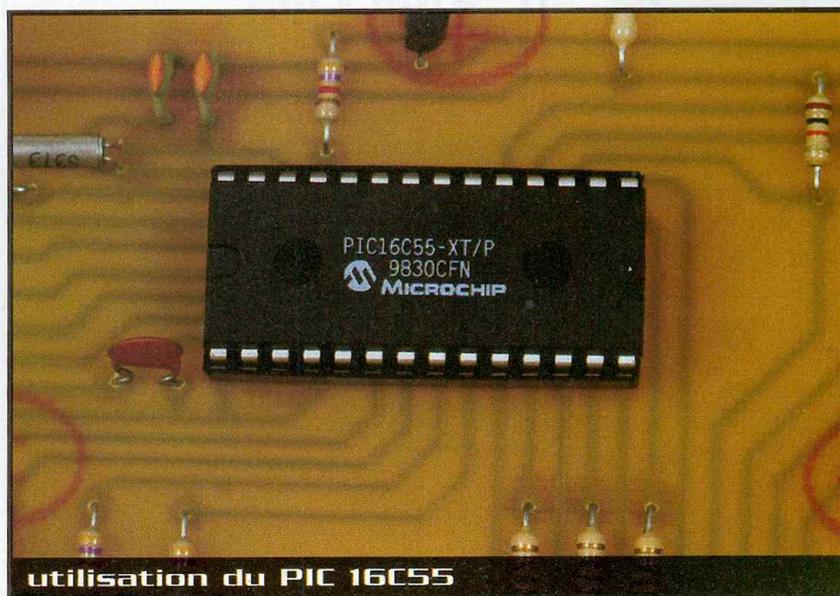
Commencer par câbler la totalité des composants en vérifiant la polarité de chaque

composant où cela s'avère nécessaire.

Concernant la carte utilisateur, on veillera à étamer les pistes faisant office de contact entre la carte elle-même et le connecteur. Pour ce faire, il suffit d'en un premier temps de procéder à l'étamage par lui-même des parties concernées puis de retirer l'excédent d'étain à l'aide de tresse à dessouder garantissant ainsi une parfaite planéité des contacts. Terminer l'opération par le net-

toyage à l'alcool.

Après une dernière vérification et avant de câbler entre elles les cartes unité centrale et afficheur, procéder, sans monter le microcontrôleur sur son support, à la mise sous tension de la carte unité centrale. Vérifier que l'on trouve bien la tension de 5V entre les broches 4 et 2 du support du microcontrôleur. De même, on pourra s'assurer du bon fonctionnement du relais d'uti-



utilisation du PIC 16C55

lisation en reliant la broche 2 et 25 du support du microcontrôleur. Insérer la carte utilisateur dans son connecteur de façon à ce que l'époxy soit du côté des contacts du connecteur et vérifier qu'une tension de 5V

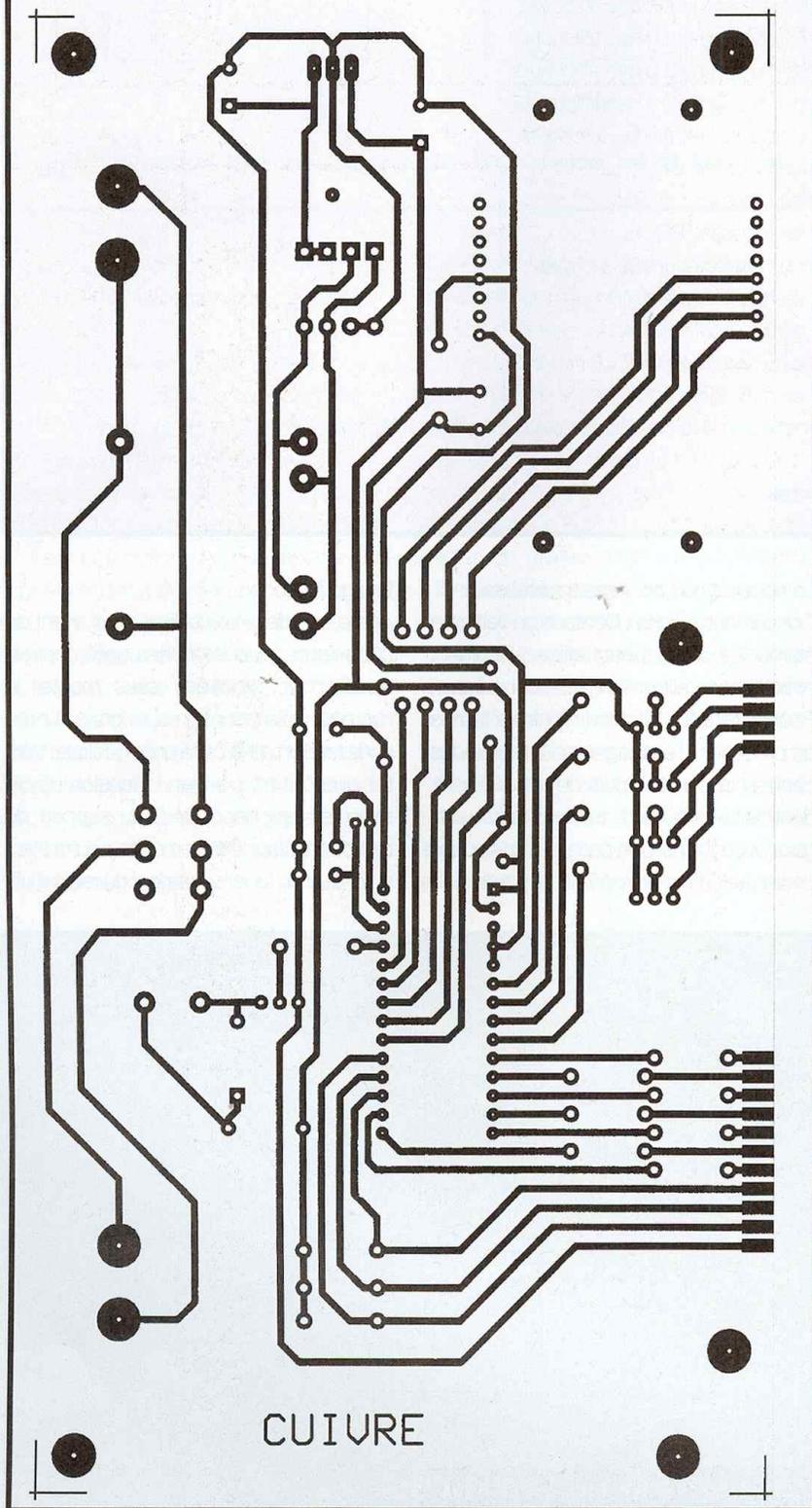
est présente sur la broche 20 du support du micro validant ainsi le fonctionnement du contact de fin de course du support de cartes à puce. Après cette vérification, retirer l'alimentation.

Maintenant relier entre elles les cartes afficheurs et unité centrale. Pour ce faire, on utilisera soit des petites pièces réalisées conformément au plan, soit on utilisera des équerres quelconques, voir des pièces de Meccano pourvu que la distance entre la face contre le circuit imprimé et les trous soit de 12 mm. Une fois relier entre elles mécaniquement, procéder aux liaisons électriques côté pistes en soudant entre elles les pastilles de la carte unité centrale et afficheur.

Procéder à la programmation du microcontrôleur. Dans le cas où le programmeur utilisé ne reconnaîtrait pas la configuration des fusibles, il faudra lui indiquer : Oscillateur = LP, WDT = OFF, Code protect = ON ou OFF peu importe.

Brocher le microcontrôleur sur son support, s'assurer qu'aucune carte utilisateur n'est insérée dans le connecteur puis mettre sous tension. Le message "CArd" doit apparaître, si cela est le cas passer au chapitre utilisation, si ce n'est pas le cas procéder à la vérification complète du montage en vérifiant également chaque piste.

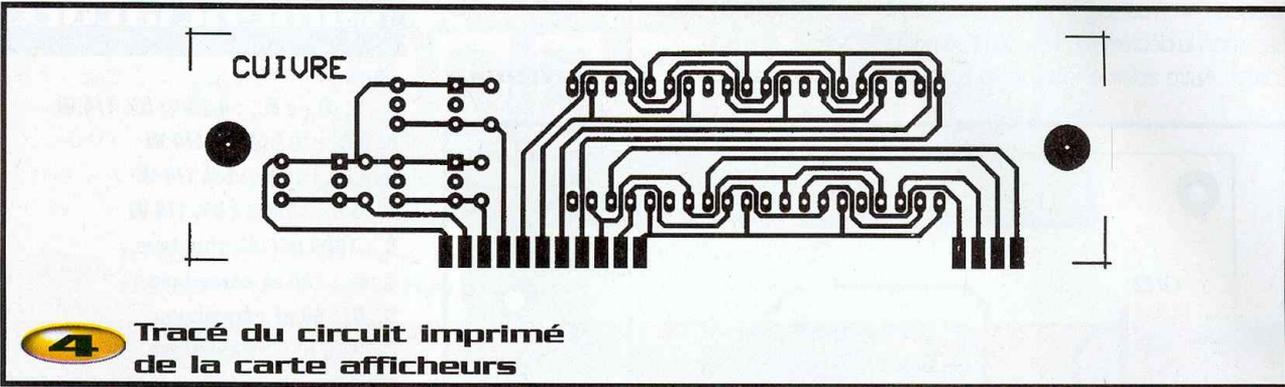
### 3 Tracé du circuit imprimé de la carte cpu



### Utilisation

Après mise sous tension sans la présence de carte utilisateur, le montage nous indique "CArd" nous demandant d'insérer une carte dans le connecteur de cartes à puce. Insérer une des cartes réalisées avec les pistes vers le bas. A ce moment, on peut lire "Fin" sur l'affichage ce qui est normal car notre carte était neuve. Elle a donc été vérifiée et écrite à 0.

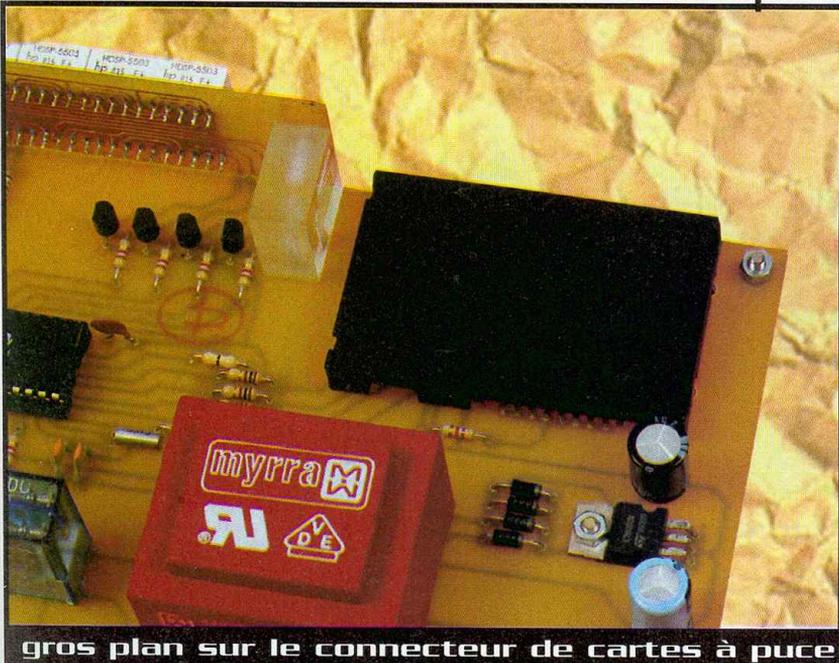
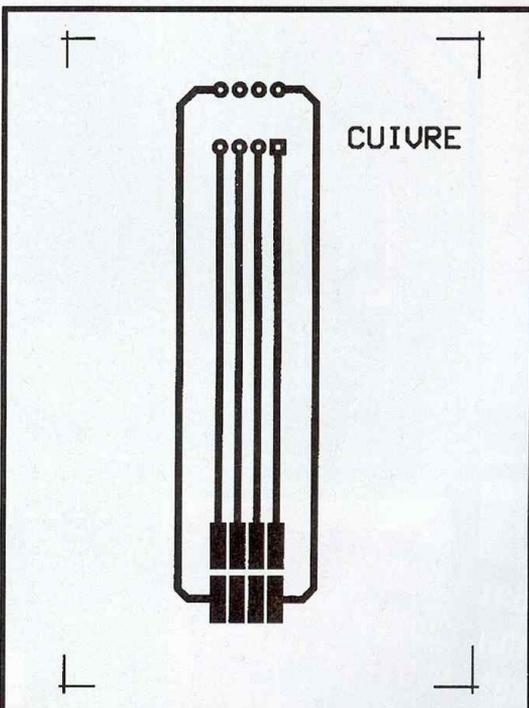
Maintenant il va falloir créditer cette carte. Appuyer sur PROG, le montage nous demande "Code". C'est maintenant que l'on va entrer notre code confidentiel par l'intermédiaire d'une séquence de touches. La séquence de touches valide est la suivante : DOWN, UP, UP, DOWN, UP, DOWN, DOWN, UP, PROG. Si la séquence de touches a été entrée correctement, on doit avoir sur l'affichage "0000". Si la séquence de touches n'a pas été entrée correctement, le message "FAIL" apparaît. Il faut alors appuyer sur PROG pour entrer une nouvelle séquence. Comme nous venons de le voir, la séquence de touches est composée de 8 UP ou DOWN puis d'un PROG, c'est l'appui sur la touche PROG en fin de séquence qui déclenche la vérifica-



tion avec le code contenu en mémoire. Donc, dans le cas où l'on sait qu'il faut 8 appuis sur les touches UP et DOWN puis 1 appui sur la touche PROG, cela nous donne 256 combinaisons possibles. Par contre, si l'on ne connaît ni le nombre d'impulsions ni sur quelles touches elles doivent être effectuées, cela nous donne une infinité de combinaisons donc une très grande sécurité. La séquence que l'on vient de donner est bien évidemment à garder en lieu sûr. Revenons à notre utilisation, nous avons entré le bon code et les afficheurs nous indiquent "0000". Nous allons maintenant incrémenter ou décrémenter les unités à l'aide des touches UP ou DOWN jusqu'à la valeur souhaitée. Rappelons qu'une unité correspond à une minute d'utilisation de l'appareil restreint. Une fois le nombre d'unités atteint qui peut se situer entre "0001" et "9998", nous allons programmer ces unités dans la carte. Pour cela il suffit

d'appuyer sur PROG. A cet instant, le relais d'utilisation se colle et le décompte commence (pour une carte à utilisation restreinte). Maintenant, nous allons programmer une carte à accès illimité. Cette carte pourra évidemment être la même que celle que nous venons de programmer puisqu'elles sont reprogrammables à volonté. Insérer la carte et recommencer la procédure comme précédemment en entrant le code puis, pour le nombre d'unités, nous allons entrer "9999" puis PROG. L'afficheur nous indique alors "FULL" confirmant que la carte est en accès permanent. De même, le relais sera collé et restera collé tant

que la carte restera insérée dans le

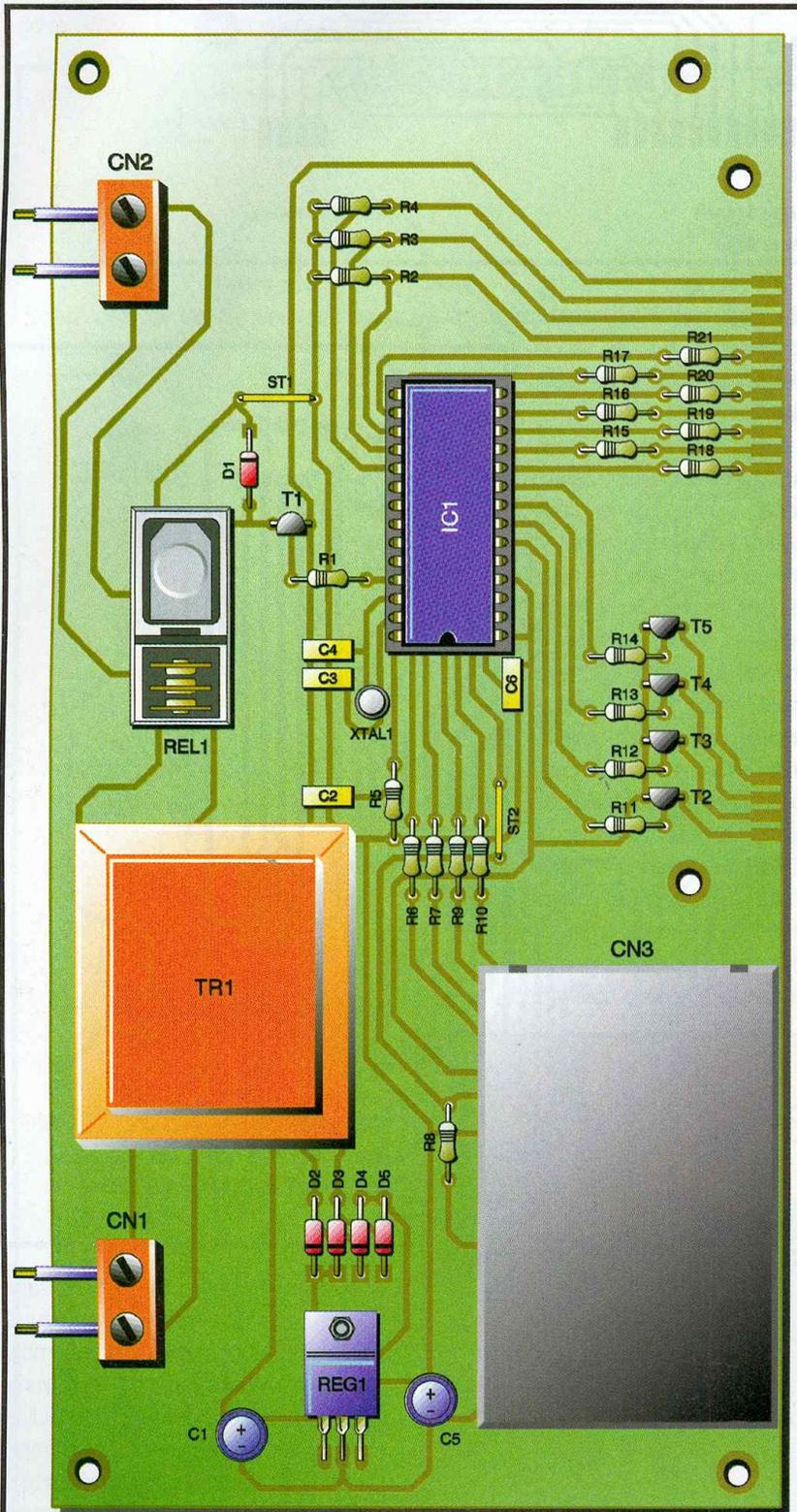


connecteur. Retirer la carte, le message "CArd" apparaît, insérer de nouveau la carte, l'afficheur indiquera soit le nombre d'unités de la carte si elle a été programmée en utilisation restreinte soit "FULL" dans le cas d'une carte à accès illimité. Dans le cas de l'utilisation d'une carte à accès restreint, dès que l'afficheur passera à "0000" donc que la carte sera vide, le relais s'ouvrira et l'afficheur nous indiquera "Fin". A ce moment, il faudra reprogrammer la carte si on le souhaite, bien entendu. Dernier conseil, retirer la carte du montage

lorsque l'on n'utilise pas l'appareil restreint car sinon le décrétement de la carte se poursuivra. Autre solution, équiper le montage

d'un interrupteur marche/arrêt.

O. COTREAU



**6** Implantation des éléments de la carte CPU

**7** Implantation des éléments de la carte utilisateurs

## Nomenclature

### Carte CPU

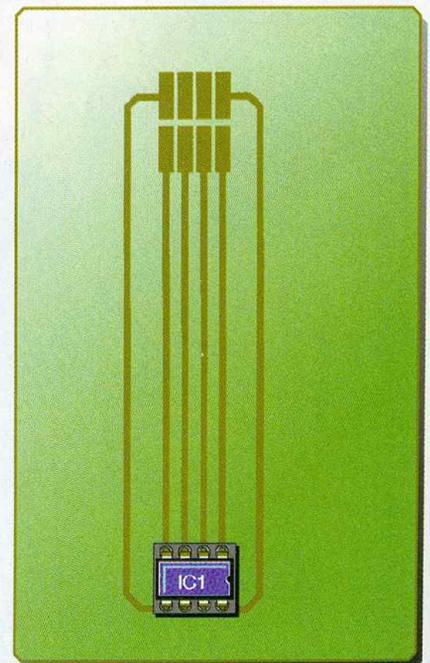
- R<sub>1</sub>, R<sub>8</sub>, R<sub>11</sub> à R<sub>14</sub> : 4,7 kΩ 5% 1/4 W
- R<sub>2</sub> à R<sub>5</sub> : 10 kΩ 5% 1/4 W
- R<sub>6</sub> à R<sub>10</sub> : 100 Ω 5% 1/4 W
- R<sub>15</sub> à R<sub>21</sub> : 180 Ω 5% 1/4 W
- C<sub>1</sub> : 1000 µF/16V chimique
- C<sub>2</sub>, C<sub>6</sub> : 100 nF céramique
- C<sub>3</sub>, C<sub>4</sub> : 56 pF céramique
- C<sub>5</sub> : 470 µF/25V chimique
- TR<sub>1</sub> : transfo 6V/5VA ex. MYRRA 44235
- CN<sub>1</sub>, CN<sub>2</sub> : borniers 2 points
- CN<sub>3</sub> : connecteur cartes à puce
- SUP<sub>1</sub> : support CI 28 broches
- T<sub>1</sub> à T<sub>5</sub> : transistors BC337
- D<sub>1</sub> : diode 1N4148
- D<sub>2</sub> à D<sub>5</sub> : diodes 1N4004
- XTAL<sub>1</sub> : quartz 32,768 kHz
- IC<sub>1</sub> : PIC 16C55
- REL<sub>1</sub> : relais 2RT/6V
- REG<sub>1</sub> : régulateur LM7805
- ST<sub>1</sub>, ST<sub>2</sub> : straps

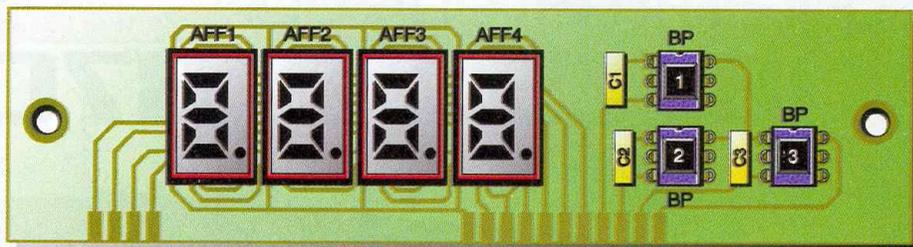
### Carte Afficheurs

- AFF<sub>1</sub> à AFF<sub>4</sub> : afficheurs 7 segments HDSP - 5503
- C<sub>1</sub> à C<sub>3</sub> : 100 nF céramique
- BP<sub>1</sub> à BP<sub>3</sub> : touches contact DIP6 ITT
- SUP<sub>1</sub> à SUP<sub>3</sub> : supports tulipe 6 broches

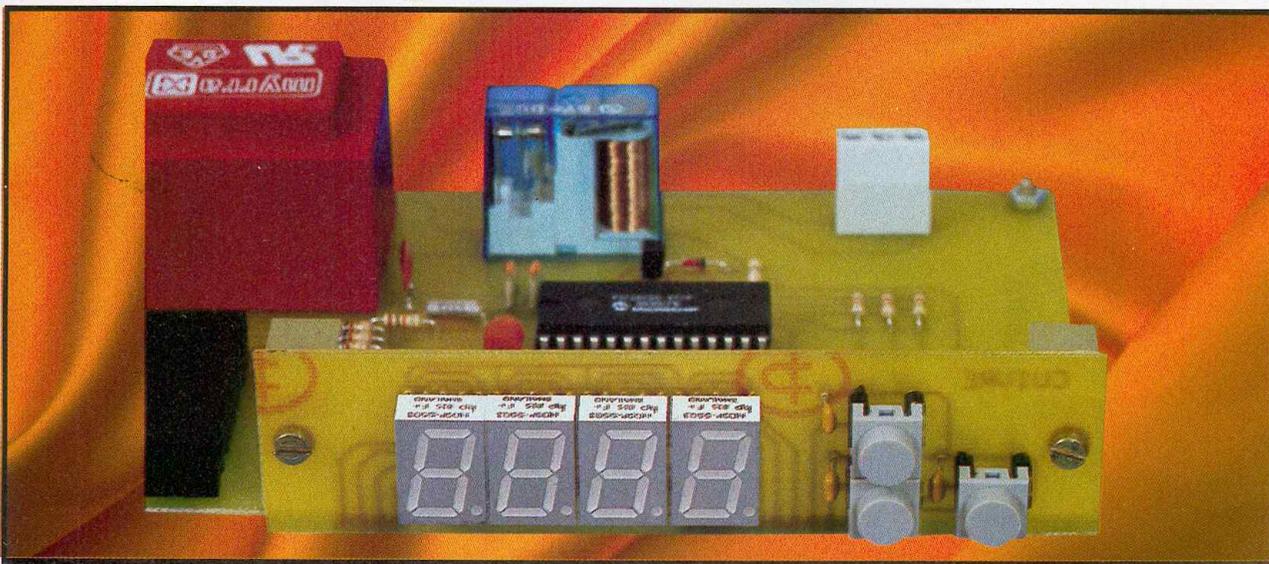
### Carte utilisateur

- IC<sub>1</sub> : EEPROM 93C46 B





**8** Implantation des éléments de la carte afficheurs



l'afficheur avec ses trois touches DIP6

# L'ENCYCLOPÉDIE DES CIRCUITS ÉLECTRONIQUES DATA-NET

10 CDs, 180.000 circuits,  
300.000 pages d'infos  
pour 395 Frs TTC seulement

Que vous soyez électronicien débutant ou confirmé, cette encyclopédie est une véritable mine d'information et vous fera gagner des centaines d'heures de recherche.

Les dix premiers CD-ROM de l'encyclopédie contiennent les fiches techniques de plus de **180.000 circuits** répartis sur **61 fabricants**, soit plus de **300.000 pages** d'information au format PDF !

C'est comme si vous disposiez chez vous, de plus de **460 data-books** et que vous puissiez retrouver une fiche technique de composant en un clin d'œil grâce à un moteur de recherche ultra performant.

De plus, les dix CD-ROM de l'encyclopédie Data-Net, sont disponibles au prix de **395 Frs TTC seulement !... (60,22 €)**

Transistors, Diodes, Thyristors, Mosfets,  
CIs, Mémoires,  $\mu$ processeurs,  $\mu$ contrôleurs, etc...

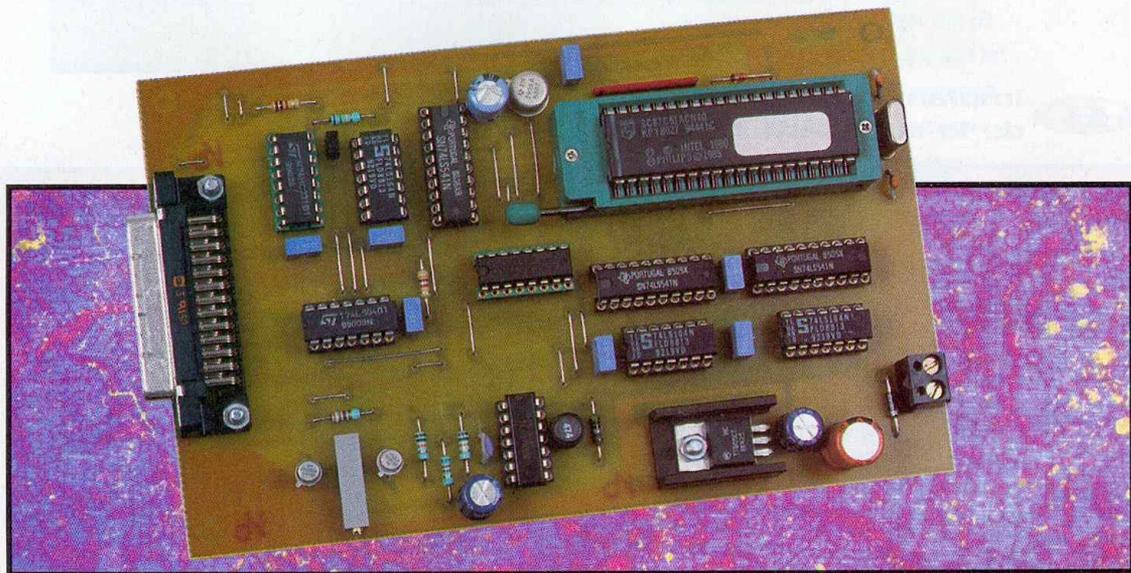


Data-Net fonctionne sur Windows® 3.1/95/98/NT3.51 et NT 4.0  
Pour recevoir Data-Net, veuillez rajouter 15 Frs au prix indiqué pour participation aux frais d'expédition (35 Frs hors France métropolitaine) et adresser votre règlement par chèque ou carte bancaire à :

**Technical Data Systems**  
501 Av. de Guigon - BP 32  
83180 SIX FOURS cedex  
Tél 04 94 34 45 31 - Fax 04 94 34 29 78

Pour commander par carte bancaire, veuillez nous communiquer vos numéros de carte et date d'expiration. Le prix de 395 Frs TTC est valable pour toute commande accompagnée d'un règlement par chèque ou carte bancaire. Pour les paiements administratifs, veuillez rajouter 50 Frs à ce prix.

# Programmeur de 87C51 et 87C52



Les microcontrôleurs 87C51 et 87C52 sont très répandus et très pratiques du fait qu'ils intègrent l'EPROM nécessaire au programme. Ils permettent ainsi de réaliser des petits montages sans avoir à ajouter de la logique autour du microcontrôleur. Les lecteurs assidus auront sûrement déjà trouvé des montages faisant appel à ces microcontrôleurs dans ces pages. Malheureusement, pour programmer ces microcontrôleurs, il faut faire appel à un équipement spécifique, souvent coûteux.

Bien entendu, avec la démocratisation des PC, sont apparues des cartes spécialisées qui permettent de programmer les EPROM et les microcontrôleurs les plus courants. Mais ce type de système, en plus d'être relativement coûteux, requiert un «slot» de type «ISA» de libre dans le PC. Selon les équipements dont on dispose déjà, il n'est pas toujours possible de trouver un emplacement de libre dans le PC. Pour résoudre ces petits problèmes nous vous proposons ce mois-ci de réaliser un programmeur de microcontrôleurs 87C51 et 87C52 qui se connectera à un port d'imprimante standard, sur un PC fonctionnant sous Windows 95/98.

## Schéma

Avant d'aborder les schémas de ce montage, précisons quelques remarque sur les choix qui ont été effectués lors de la conception de cet appareil. Nous souhaitons réaliser un système pouvant être connecté facilement à un PC, sans avoir à l'ouvrir pour installer une carte de plus dans un hypothétique emplacement d'extension libre. Le choix de la connexion

au port imprimante s'est imposé de lui-même car il procure plusieurs avantages : connexion simple et rapide permettant l'utilisation de l'appareil sur plusieurs PC. L'accès au port imprimante, d'un point de vue logiciel, est relativement simple, ce qui a permis d'envisager d'écrire le programme associé à l'appareil sous WINDOWS (3.1 et 95). En revanche, nous avons choisi d'utiliser un port imprimante standard pour ne pas pénaliser les lecteurs qui n'ont pas un PC récent (qui ne dispose pas d'un port EPP/ECP). En conséquence, les données à transmettre au montage sont sérialisées, ce qui ralentit considérablement les échanges. Le résultat est donc un appareil agréable et simple d'emploi, mais qui ne rivalisera pas en terme de performance avec les équipements du commerce (plus destiné à la production).

Après ce préambule venons-en aux schémas. La **figure 1** dévoile le cœur du système et la connexion au port imprimante. Les données à transmettre aux EPROM et les adresses associées sont transmises en série aux registres à décalage  $U_3$ ,  $U_4$  et  $U_5$ . A la fin du transfert, le registre  $U_5$  contiendra les données et

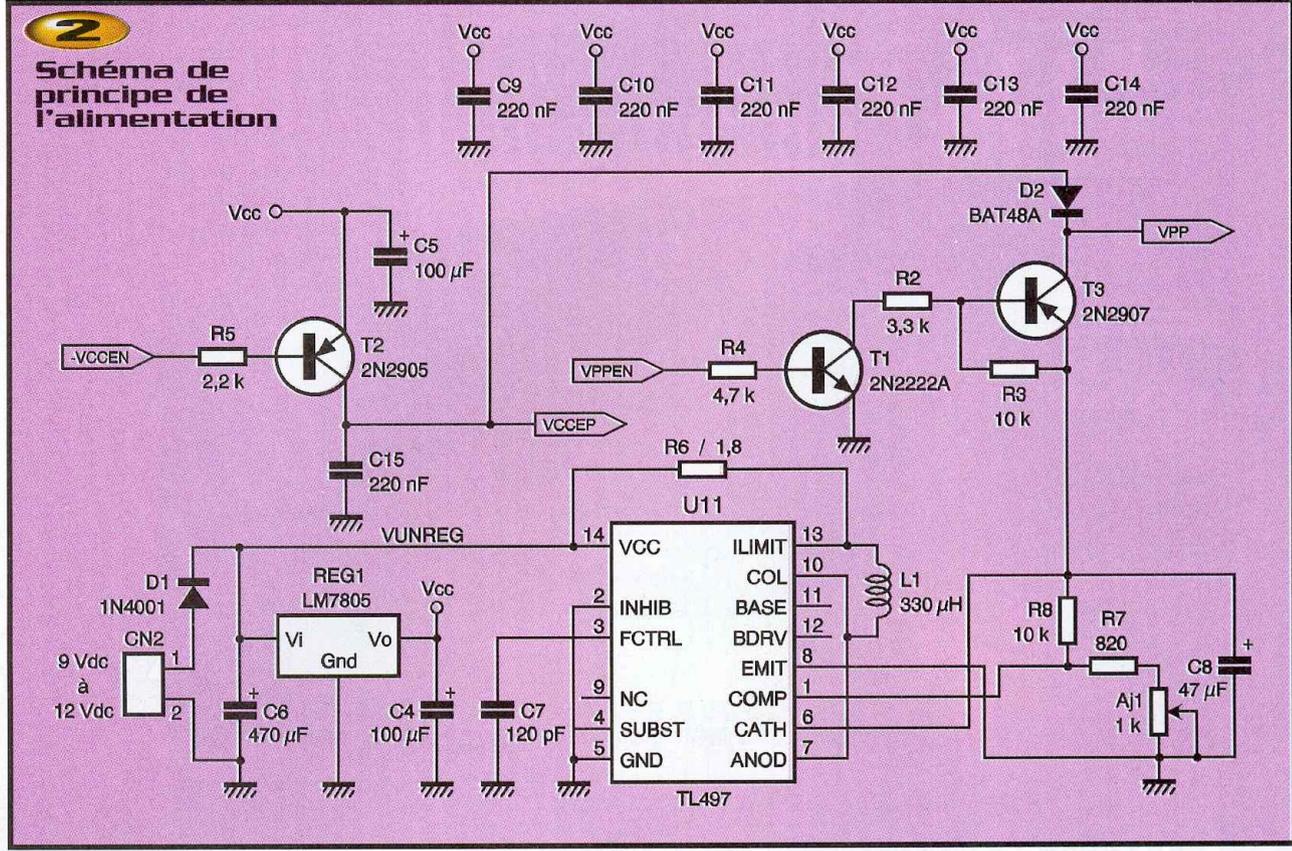
les registres  $U_3$  et  $U_4$  contiendront l'adresse à appliquer à l'EPROM.

Les sorties des registres à décalage sont «tamponnées» par des buffers à sorties 3 états pour éviter les perturbations lors du transfert des données. En ce qui concerne  $U_7$ , les sorties du buffer sont commandées lorsque les sorties du microcontrôleur sont inactives, grâce à la présence de  $U_{20}$ . Pour relire les données de l'EPROM, le circuit  $U_6$  permettra de sélectionner tour à tour chacun des bits du bus de donnée. Cela augmente la charge de travail du programme du PC (ce qui ralentit un peu le programme), mais c'est la seule solution pour utiliser un port imprimante standard.

Pour générer l'impulsion de programmation, nous avons fait appel à un monostable pour garantir la durée de l'état bas. Nous n'avons pas fait appel à une temporisation par logiciel à cause des contraintes imposées par WINDOWS 95/98. En effet, WINDOWS 95 est un système d'exploitation multitâche, de sorte qu'il est très difficile de contrôler des durées de l'ordre de la milliseconde. Or pour programmer les microcontrôleurs 87C51, il faut pouvoir générer 5 impulsions de 100 ms à l'état



**2**  
Schéma de principe de l'alimentation



bas sur la broche -PGM. Ceci est très compliqué à obtenir avec un PC sous Windows, d'autant plus que le programme aurait à s'adapter à la vitesse de chaque PC. Nous avons donc préféré confier cette tâche à un simple monostable.

Le schéma de la **figure 2** dévoile l'alimentation du montage. En plus de la tension +5VDC nécessaire aux circuits logiques, le montage doit produire les

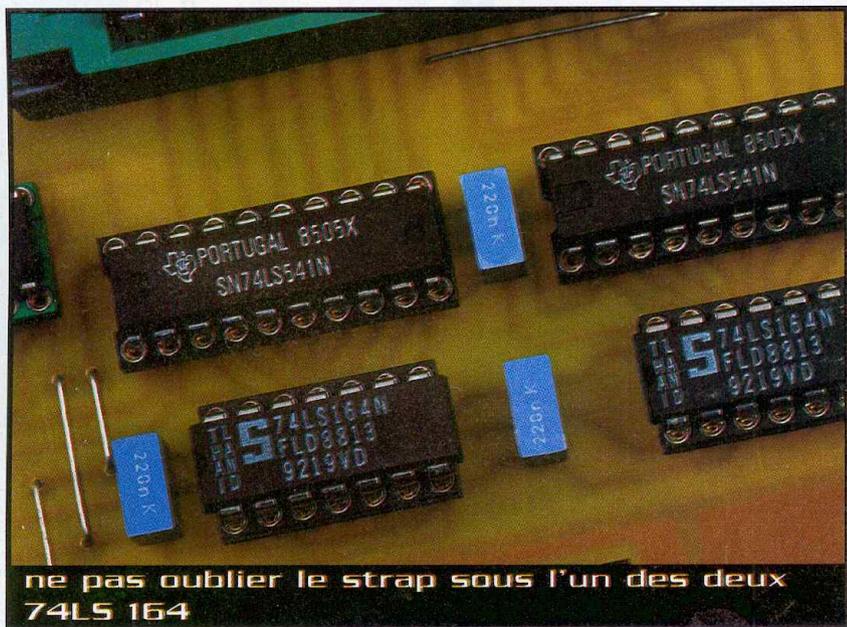
tensions VPP et VCCEP à appliquer au microcontrôleur à programmer. La tension VCCEP sera produite par un simple transistor monté en commutation sur le +5V. Par contre, pour produire les 12,75V nécessaires pour VPP, nous avons fait appel à un régulateur à découpage monté en «STEP UP». Pour cette fonction, nous avons fait appel au circuit TL497 en raison de sa simplicité de mise en œuvre. L'ajustable AJ, permet-

tra un réglage précis de la tension de programmation des microcontrôleurs. Le programme associé au montage dispose d'une procédure de réglage.

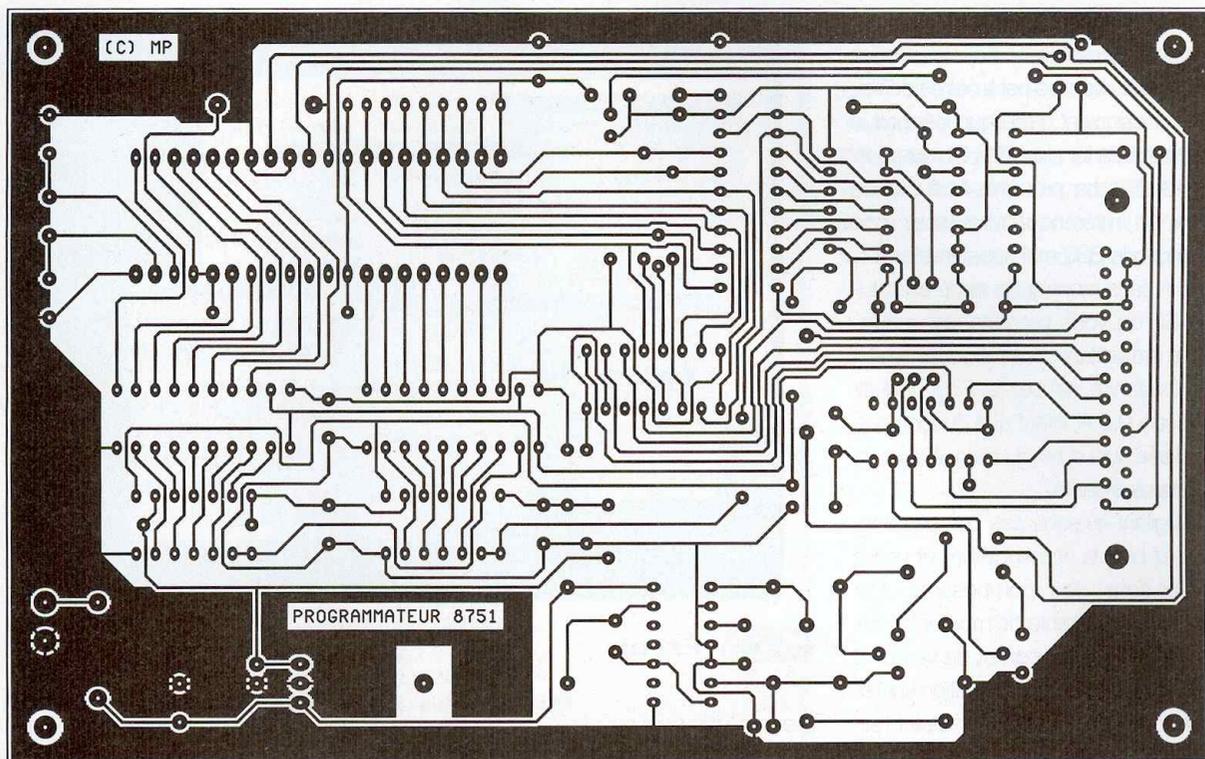
Les tensions VPP et VCC appliquées à l'EPROM doivent pouvoir être commandées rapidement à un instant précis ou être totalement absente du support. Pour cela nous avons fait appel aux transistors T<sub>2</sub> et T<sub>3</sub> qui seront pilotés directement par le port imprimante. L'alimentation de la logique du montage est articulée autour du régulateur LM7805 (REG.). La carte sera alimentée par une tension de 12VDC qui n'a pas besoin d'être stabilisée. Une tension correctement filtrée fera très bien l'affaire à condition de pouvoir fournir au moins 500 mA. La diode D<sub>1</sub> permet de protéger le montage en cas d'inversion du connecteur d'alimentation.

**Réalisation**

Le dessin du circuit imprimé est visible en **figure 3**. 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 raison de la taille réduite de certaines pastilles, il vaudra mieux utiliser des forets de bonne



ne pas oublier le strap sous l'un des deux 74LS 164



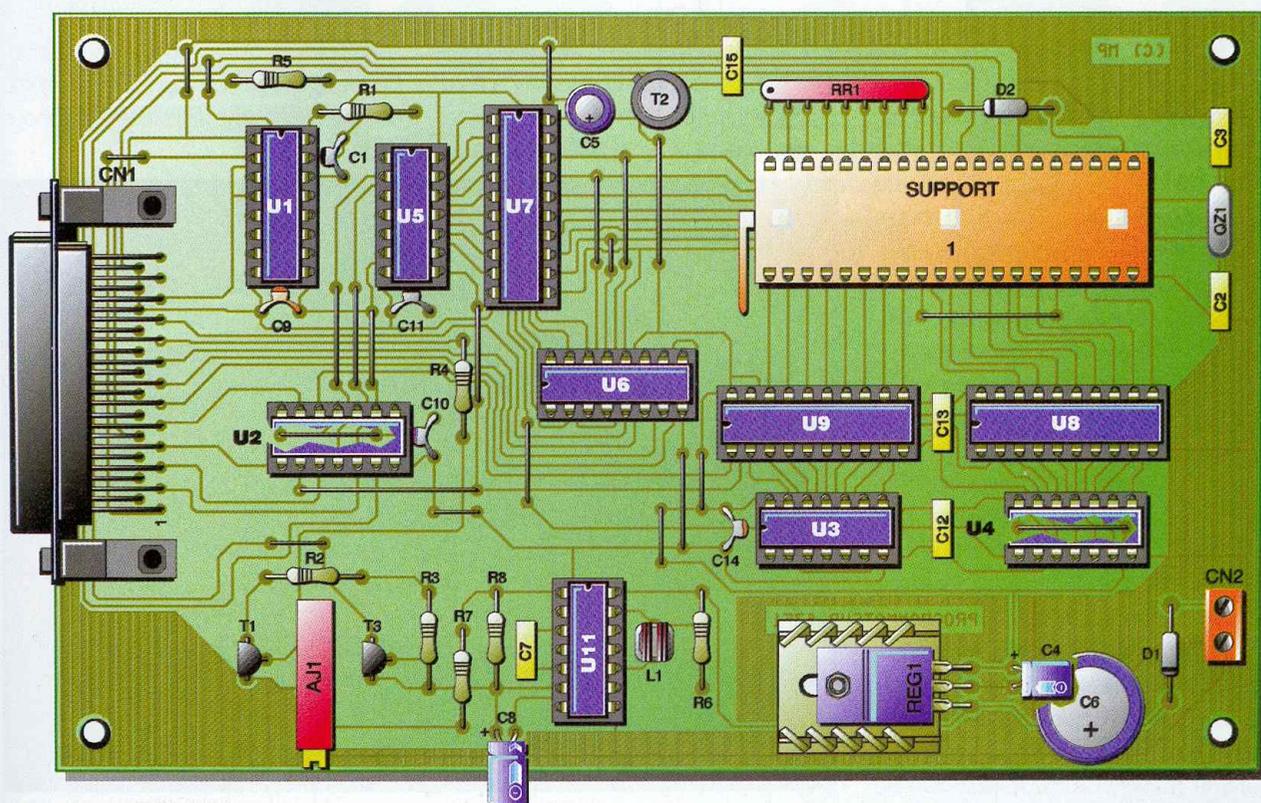
### 3 Tracé du circuit imprimé

qualité pour éviter de les emporter au moment où le foret débouche. En ce qui concerne le support, le régulateur, l'induc-

tance et les diodes, il faudra percer les pastilles avec un foret de 1 mm 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

### 4 Implantation des éléments

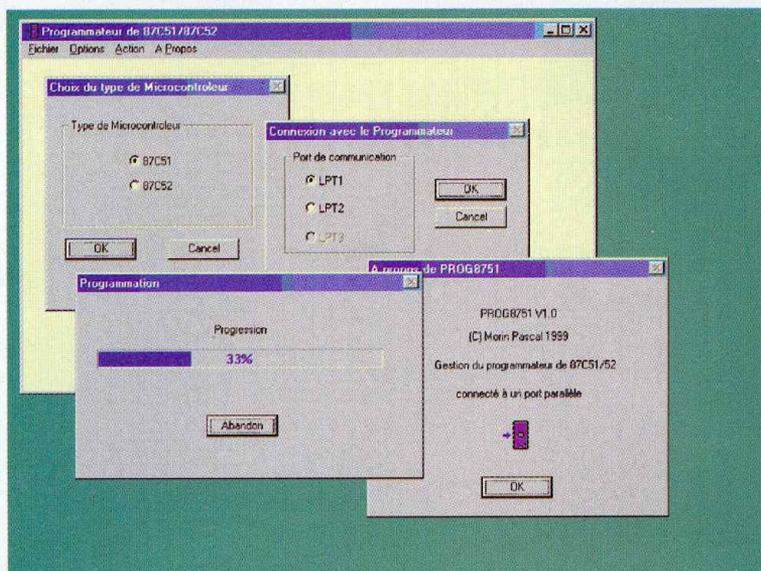


concerne particulièrement l'inductance et la résistance ajustable. Surtout ne prenez pas de liberté avec les paramètres de l'inductance, car c'est d'elle que dépend la génération de la tension VPP. Si vous n'arrivez pas à vous procurer l'inductance donnée en référence, choisissez une inductance de 330 mH capable d'absorber 0,5 A sans montrer de signe de saturation. Si l'on vous propose une inductance se présentant sous l'encombrement d'une résistance, refusez-la. Pour pouvoir «encaisser» 0,5 A, il faut que l'inductance soit réalisée sur un noyau magnétique de dimension suffisante.

Soyez vigilant au sens des composants, respectez bien la nomenclature et veillez bien au sens des composants. Par ailleurs, il est préférable de monter les circuits intégrés sur support et de vérifier la présence de tension d'alimentation sur les supports. Ensuite (après avoir éteint l'alimentation, bien sûr), vous pourrez insérer les circuits intégrés sur leur support. Vous noterez que les straps sont relativement nombreux puisqu'ils sont au nombre de 22. C'est le prix à payer pour garder un circuit imprimé simple face. Vous noterez que certains straps sont installés sous les circuits intégrés. Pour plus de commodité, il est indispensable de débiter l'implantation par les straps.

Le connecteur CN<sub>1</sub> sera sûrement soumis à de nombreuses manipulations. C'est pourquoi il sera immobilisé sur le circuit imprimé à l'aide de 2 petits boulons, dans les passages prévus à cet effet. Pour raccorder CN<sub>1</sub> au port parallèle du PC, vous devrez utiliser deux connecteurs 25 points mâles à sertir sur du câble plat. Étant donné le nombre élevé de manipulations que subira le câble, il sera indispensable d'ajouter des «clips anti-traction» aux connecteurs. Ceci vous évitera d'arracher le câble au bout de quelques manipulations.

Le régulateur sera impérativement monté sur un petit dissipateur présentant une résistance thermique inférieure à 18°C/W. Ceci est nécessaire pour limiter la température de jonction du régulateur à une valeur acceptable, car la consommation du montage est loin d'être négligeable. Si vous choisissez un modèle de dissipateur différent de celui préconisé dans la nomenclature, vérifiez qu'il ne



## 5 Écran

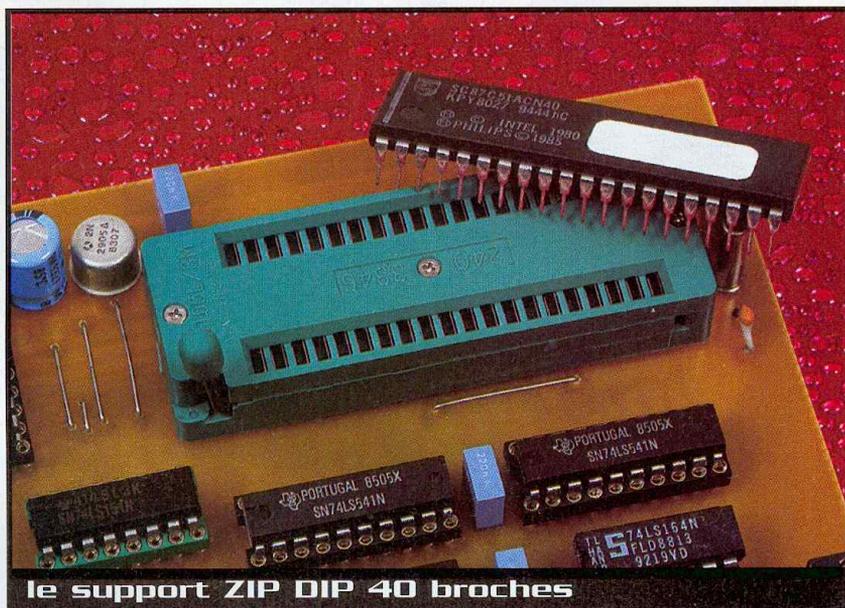
prendra pas trop de place sur le circuit imprimé.

### Utilisation du programmeur d'EPROM

Le programme WEPROM associé à ce montage est prévu pour fonctionner sous Windows 95 ou Windows 98. Il n'y a aucune procédure d'installation à prévoir puisque le programme se suffit à lui-même (pas de DLL ou de fichiers spécifiques). Pour mettre en route le programme, il vous suffit de l'appeler à partir du gestionnaire de fichiers, ou d'ajouter

son icône dans un groupe de programme (ou dans le menu DEMARRER sous WINDOWS 95).

Lors de la première mise en route du programme, ce dernier utilise les paramètres par défaut suivants : port imprimante LPT1 et type de microcontrôleur 87C51. Utilisez le menu OPTION pour modifier les paramètres de fonctionnements. Les options seront automatiquement enregistrées dans le fichier WIN.INI sous la section [PROG8751] pour être utilisées lors de la prochaine session. Pour le reste des opérations, le programme est tellement simple à utiliser que nous



le support ZIP DIP 40 broches

## Nomenclature

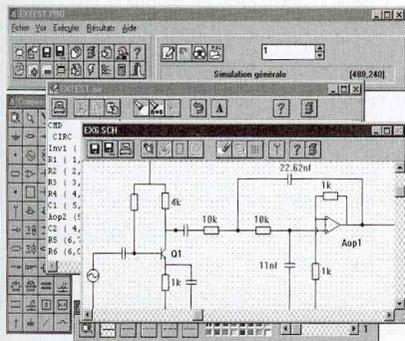
**AJ<sub>1</sub>** : ajustable multitours 1 k $\Omega$   
**CN<sub>1</sub>** : connecteur SubD 25 points femelle, sorties coudées, à souder sur circuit imprimé (par exemple référence HARTING 09 66 312 7601)  
**CN<sub>2</sub>** : bornier de connexion à vis 2 plots, au pas de 5,08 mm, à souder sur circuit imprimé, profil bas  
**C<sub>1</sub>** : 1 nF céramique au pas de 5,08 mm  
**C<sub>2</sub>, C<sub>3</sub>** : 33 pF céramique au pas de 5,08 mm  
**C<sub>4</sub>, C<sub>5</sub>** : 100  $\mu$ F/25V sorties radiales  
**C<sub>6</sub>** : 470  $\mu$ F/25V sorties radiales  
**C<sub>7</sub>** : 120 pF céramique au pas de 5,08 mm  
**C<sub>8</sub>** : 47  $\mu$ F/25V sorties radiales  
**C<sub>9</sub> à C<sub>15</sub>** : 220 nF céramique au pas de 5,08 mm  
**D<sub>1</sub>** : 1N4001 (diode de redressement 1A/100V)  
**D<sub>2</sub>** : diode Schottky BAT48A  
**L<sub>1</sub>** : inductance 330  $\mu$ H sur noyau de ferrite (I<sub>max</sub>=0,5A)  
**QZ** : quartz 4 MHz en boîtier HCU49  
**RR** : réseau résistif 8x10 k $\Omega$  en boîtier SIL  
**REG<sub>1</sub>** : régulateur LM7805 (5V) en boîtier

**T0220 +** dissipateur thermique 18°C/W maximum (par exemple SHAFFNER référence RAWA 400 9P)  
**R<sub>1</sub>** : 150 k $\Omega$  1/4W 5% (marron, vert, jaune)  
**R<sub>2</sub>** : 3,3 k $\Omega$  1/4W 5% (orange, orange, rouge)  
**R<sub>3</sub>, R<sub>8</sub>** : 10 k $\Omega$  1/4W 5% (marron, noir, orange)  
**R<sub>4</sub>** : 4,7 k $\Omega$  1/4W 5% (jaune, violet, rouge)  
**R<sub>5</sub>** : 2,2 k $\Omega$  1/4W 5% (rouge, rouge, rouge)  
**R<sub>6</sub>** : 1,8  $\Omega$  1/4W 5% (marron, gris, argent)  
**R<sub>7</sub>** : 820  $\Omega$  1/4W 5% (gris, rouge, marron)  
**SUPPORT<sub>1</sub>** : support ZIP DIP 40 broches  
**T<sub>1</sub>** : 2N2222A  
**T<sub>2</sub>** : 2N2905  
**T<sub>3</sub>** : 2N2907  
**U<sub>1</sub>** : 74HCT221  
**U<sub>2</sub>** : 74LS04  
**U<sub>3</sub> à U<sub>5</sub>** : 74HCT164  
**U<sub>6</sub>** : 74HCT151  
**U<sub>7</sub> à U<sub>9</sub>** : 74HCT541  
**U<sub>11</sub>** : TL497

n'avons pas besoin de vous en dire plus. Vous n'aurez qu'à suivre les messages qui seront affichés. Notez simplement que le programme accepte uniquement des fichiers au format binaire. Si vous devez utiliser des fichiers au format HEXA, vous devrez tout d'abord les convertir. On trouve sur Internet de nombreux programme de conversion pour cela.

P. MORIN

La Simulation pour tous !!!



N'hésitez plus à concevoir vos propres circuits !!

Turbo Analogic.....595 Francs  
 Port.....Gratuit

Total T.T.C.....595 Francs

Envoyer votre commande avec règlement à :

**PROSILOG**

\*\*\*\*\*

24 R.N 14  
 27380 Grainville

Tél: 0232491747 Fax: 0232481921

595 F

### Turbo Analogic

est un logiciel de simulation de circuits électroniques fonctionnant en régime linéaire, particulièrement adapté à la simulation de circuits BF, HF ou Hyperfréquences.

De nombreux outils facilitent la prise en main et permettent de développer rapidement différents circuits ou modèles.

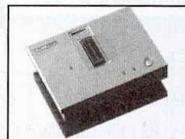
Les multiples résultats disponibles, Tensions, Courants, Impédances, Tos, paramètres [S] etc... peuvent être visualisés sous forme texte ou graphique.

**Logiciel et manuel en Français**

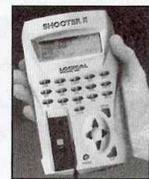
PC sous Windows 3.1 ou supérieur, lecteur de disquettes haute densité 3,5 pouces.

EP

## PROGRAMMATEURS : PLUS DE 50 MODÈLES



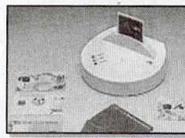
Nouveau ALL-07 C  
 Hi-Lo Systems



SHOOTER II



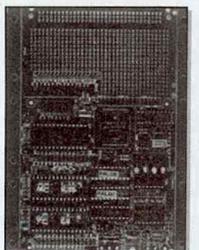
MEGAMAX  
 MEGAMAX-4C



**MEDIAL C/CHIPI**  
 Kit de développement de cartes à puce avec (ou sans) interpréteur Basic

**Dunfield**  
 Compileur C  
 Assembleur  
 Débogueur  
 Moniteur  
 Simulateur  
 CPU :

- 68HC08/6809
- 68HC11/12/16
- 8051/52
- 8080/85/86
- 8096



**Carte d'application avec CPU Intel et Motorola**

**Aussi disponibles :**

- Effaceur UV • Cartes I/O • Simulation logique-analogique mixte et routage
- Emulateur de ROM et de microcontrôleur • Analyseur logique

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

40, rue Saint-André  
 72000 LE MANS

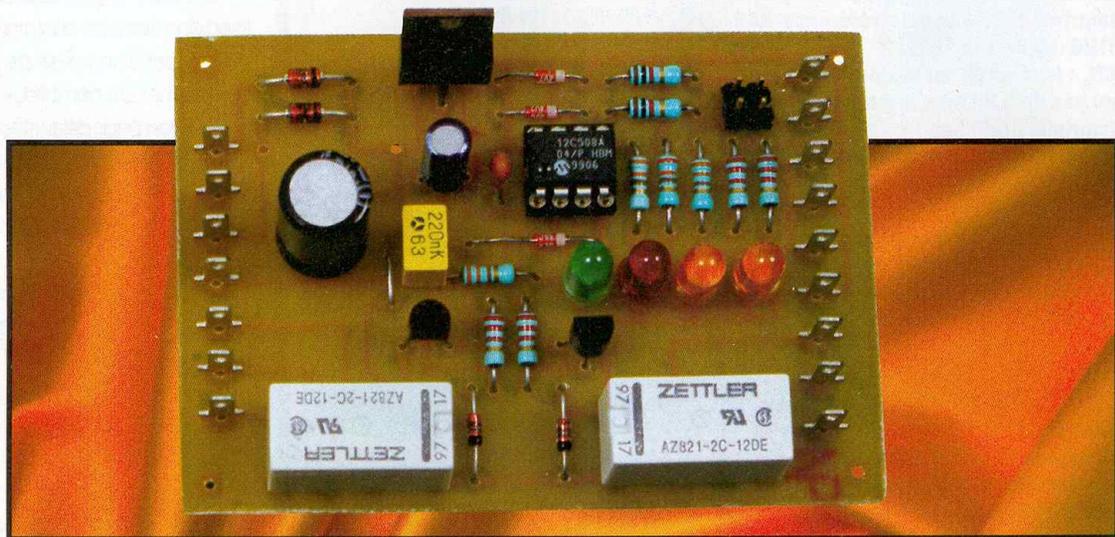
Tél. 02 43 28 15 04

Fax 02 43 28 59 61

E-mail : hitools@hitechtools.com

<http://www.hitechtools.com>

# Centrale d'alarme à microcontrôleur



Sauf dans quelques cas particuliers, les fonctions d'une centrale d'alarme domestique ne nécessitent pas le recours à un microcontrôleur puisqu'une poignée de circuits logiques peut suffire. Pourquoi dans ces conditions vous proposer cette réalisation ? Tout simplement parce qu'elle fait appel à une famille de microcontrôleurs assez récents de chez MICROCHIP qui permettent une réelle simplification du schéma. En effet, ces microcontrôleurs n'ont besoin d'aucun composant externe pour fonctionner (pas de circuit de reset ni d'horloge) et, de plus, ils sont disponibles en boîtiers DIL 8 pattes.

## Les microcontrôleurs 12C508 et 509 de MICROCHIP

Les circuits 12C508 et 12C509 (et leurs versions A) appartiennent à la famille de microcontrôleurs RISC de MICROCHIP, avec laquelle ils partagent les performances bien connues au rang desquelles se classe une très grande vitesse d'exécution. Ils ont cependant été optimisés en vue d'une mise en œuvre très simple afin de permettre leur intégration dans des applications où l'on n'a pas l'habitude de rencontrer un microcontrôleur ou bien, encore, où la mise en place d'un tel circuit serait normalement trop complexe. De ce fait, et même s'ils peuvent se comporter comme n'importe quel microcontrôleur normal avec circuit d'horloge et de reset, ils peuvent également fonctionner dans un mode que l'on peut qualifier d'autonome dans lequel toutes ces ressources sont internes au boîtier.

Toutes les pattes de ce dernier, hormis bien sûr l'alimentation positive et la masse, sont alors disponibles pour votre application. C'est heureux car, comme vous allez le découvrir dans un instant, le 12C508A que nous

avons retenu est disponible en boîtier DIL à 8 pattes seulement.

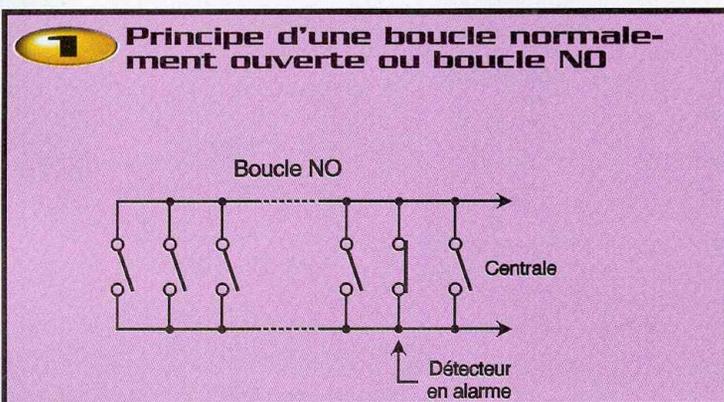
Nous en resterons là de la présentation de ce circuit ; une description un peu plus complète vous étant fournie par ailleurs, dans l'article consacré à la réalisation d'une minuterie à microcontrôleur.

## Centrales d'alarme et boucles

Avant de voir le schéma de notre centrale d'alarme, il nous semble important de préciser quelques notions relatives au mode de câblage de ces dernières et en particulier aux boucles appelées NC (ou NF) et NO. Pour relier les différents détecteurs, d'intrusion ou de défaut technique, à une centrale

d'alarme, on utilise ce que l'on appelle des boucles. C'est à dire encore que chaque détecteur n'est pas directement relié à la centrale mais que, au contraire, ces derniers sont câblés en série ou en parallèle constituant ainsi des boucles NF ou NO.

Pour bien comprendre ce principe et la différence qui existe entre ces deux méthodes de câblage, il faut retenir que la sortie d'un détecteur, quel qu'il soit, est toujours un contact qui peut être normalement ouvert ou NO, ou bien encore normalement fermé ou NF (NC pour Normally Closed dans les documentations d'origine anglo-saxonnes). L'état que l'on qualifie de normal dans ce cas est évidemment l'état hors alarme.

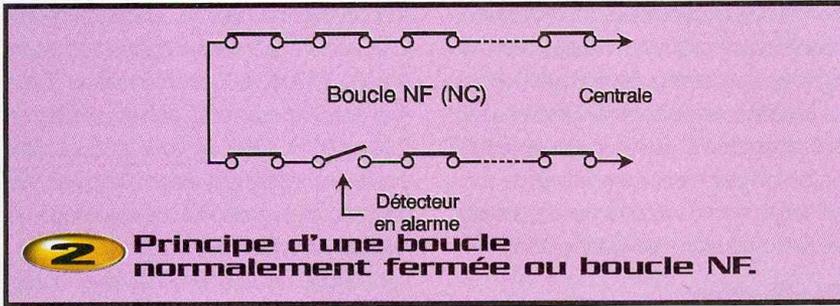


## Présentation de notre centrale

Notre centrale dispose de deux entrées pour boucles NF et d'une entrée pour boucle NO. Une des entrées NF est à action immédiate alors que l'autre dispose d'une temporisation programmée à 45 secondes. Par contre, l'entrée NO est utilisable en alarme 24 heures sur 24 pour de la surveillance technique ou bien encore comme entrée "panique" à action immédiate.

Deux sorties d'alarme sont disponibles : une pulsée et une continue permettant ainsi d'actionner les dispositifs de votre choix ; ces sorties ayant lieu via des relais disposant de contacts inverseurs. Ces sorties d'alarme sont temporisées et s'arrêtent seules après 5 mn de fonctionnement. Une sortie d'alarme complémentaire pilote une LED qui n'est pas temporisée, ce qui permet donc de bénéficier d'une mémoire d'intrusion si, par exemple, votre alarme a été déclenchée en votre absence et se trouve silencieuse à votre retour du fait de sa temporisation.

L'alimentation est confiée à un bloc secteur "prise de courant" très peu coûteux mais elle peut être sauvegardée par batterie ce que nous conseillons évidemment vive-



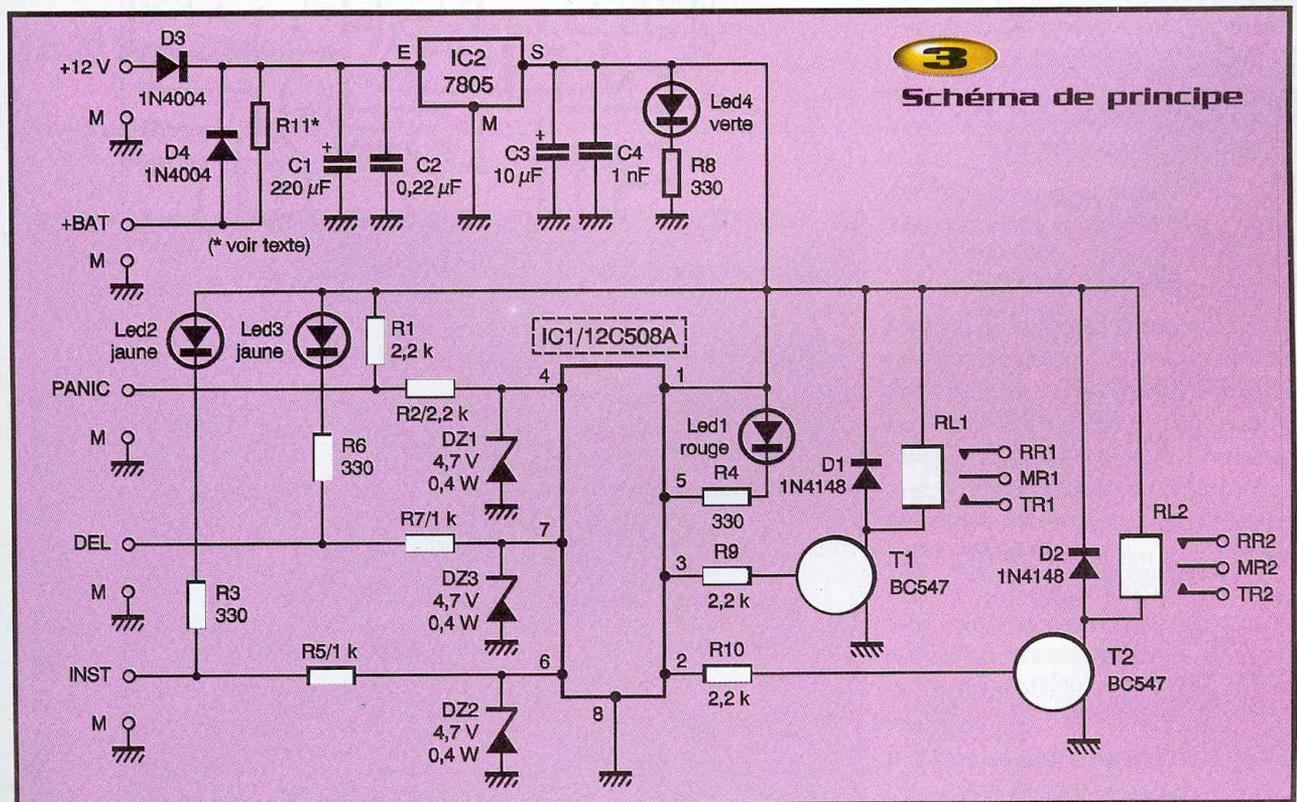
De ce fait, les détecteurs à sortie NO ne peuvent être reliés à la centrale d'alarme qu'en parallèle comme le montre la **figure 1**. On constitue alors ainsi ce que l'on appelle une boucle NO puisque, à l'état de repos, elle est ouverte. En cas d'alarme et quel que soit le détecteur qui sera sollicité, la boucle sera fermée du fait du câblage parallèle et la centrale le détectera

Par contre, les détecteurs à sortie NF ne peuvent être câblés qu'en série comme le montre la **figure 2**, constituant ainsi une boucle NF. En cas d'alarme et, ici aussi, quel que soit le détecteur sollicité, la boucle se trouvera ouverte du fait du câblage en série et la centrale le détectera.

A l'heure actuelle, la majorité des détecteurs dispose d'une sortie pour boucle NF car c'est la plus résistante à toute tentative de sabotage. En effet, pour neutrali-

ser une boucle NO, il suffit de couper un de ses fils pour rendre inutiles tous les détecteurs qui se trouvent au-delà de la coupure. Par contre, une boucle NF ne peut être neutralisée qu'en la court-circuitant, ce qui est beaucoup plus difficile à réaliser surtout si le câblage fait appel à du câble rond ou méplat style câble de téléphone. Il faut en effet dénuder ce dernier, ainsi que les "bons" fils internes, pour arriver à les court-circuiter avec des grappe-fils ou des pinces crocodiles miniatures. C'est nettement plus délicat que le brutal coup de pince coupante suffisant pour une boucle NO !

Signalons à ce propos que notre confrère "Les Dossiers du Haut-Parleur" consacre son numéro de juin/juillet à l'alarme et à la sécurité et que vous pourrez y trouver de nombreux conseils pertinents quant à l'installation d'un système d'alarme.



ment. Enfin, du fait de l'utilisation d'un microcontrôleur PIC 12C508A, le prix de revient de ce montage est dérisoire et son schéma est fort simple comme vous allez pouvoir le constater sans plus tarder.

## Analyse du schéma

Il vous est présenté dans son intégralité **figure 3** et peut difficilement être plus simple. En effet, toute la logique de la centrale est assurée par IC<sub>1</sub>, qui n'est autre que le microcontrôleur 12C508A. Nous allons analyser ce schéma en nous servant comme guide de ses pattes.

La patte 1 est l'alimentation positive du microcontrôleur et la patte 8 la patte de masse. Cette alimentation doit être réalisée sous une tension stabilisée de 5V qui est ici délivrée par le régulateur intégré IC<sub>2</sub>. Celui-ci reçoit en entrée une tension non stabilisée de 9 à 12V délivrée par un bloc secteur "prise de courant". La diode D<sub>3</sub> protège le montage des inversions de polarité accidentelles tandis que la diode D<sub>4</sub> et la résistance R<sub>11</sub> permettent l'utilisation d'une batterie de sauvegarde. Les pattes 2 et 3 sont programmées en sorties et commandent les deux relais d'alarme via les transistors amplificateurs T<sub>1</sub> et T<sub>2</sub> selon un schéma très classique.

La patte 4, quant à elle, est configurée en entrée et reçoit la boucle NO destinée à l'entrée "panique". Cette entrée est ramenée au niveau logique haut par R<sub>1</sub> et passe à la masse dès que la boucle correspondante se ferme. Notez la présence de R<sub>2</sub> et DZ<sub>1</sub>, qui protègent le microcontrôleur de toute tension transitoire violente pouvant provenir de l'extérieur.

La patte 5 est configurée en sortie et pilote la LED d'indication d'alarme ; LED qui reçoit par ailleurs une autre fonction que nous verrons lors de la description du mode d'emploi. Le 12C508 étant capable d'absorber près de 10 mA sur cette sortie, aucun transistor externe n'est nécessaire pour commander la LED. Les pattes 6 et 7 enfin sont câblées de la même façon et sont toutes deux destinées aux deux boucles NF et sont ramenées au niveau logique haut grâce aux résistances R<sub>3</sub> et R<sub>6</sub> et aux LED LED<sub>2</sub> et LED<sub>3</sub>. Lorsque les boucles correspondantes sont dans leur état normal, c'est à dire fermées, les LED sont allumées. Ceci

permet de s'assurer, lorsque l'on met la centrale en marche, que l'on n'a pas oublié une porte ou une fenêtre ouverte par exemple. Le fait que l'une des boucles (repérée DEL) soit temporisée et que l'autre (repérée INST) ne le soit pas n'a aucune influence sur le schéma comme vous pouvez le constater ; la distinction étant faite par logiciel.

## Réalisation

Compte tenu de la simplicité du schéma, le circuit imprimé que nous avons dessiné est très compact bien qu'il supporte tous les composants. Son tracé vous est proposé **figure 4**.

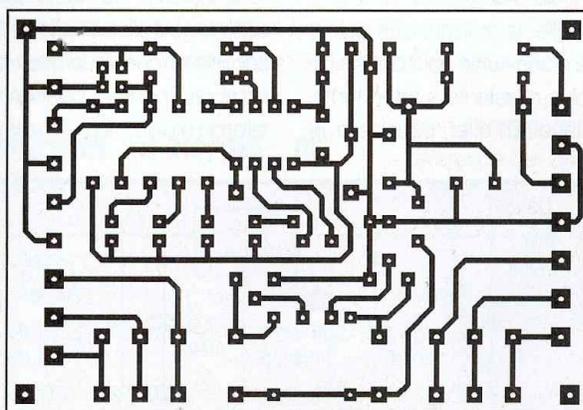
Les composants utilisés sont classiques et disponibles partout, de même que le PIC 12C508A car ce circuit a un usage bien moins noble que celui que nous lui avons trouvé ; c'est en effet lui le "Mod chip" qui sert, une fois correctement programmé bien sûr, à "pirater" les consoles de jeu Play Station de Sony...

Pour ce qui est du nôtre, il faudra bien évidemment le programmer avec le logiciel de

commande de l'alarme, logiciel disponible sur notre site Internet sous l'appellation ALARMPIC.HEX. Ce fichier est au format hexadécimal normalisé délivré par l'assembleur MICROCHIP et vous pouvez donc l'utiliser directement avec n'importe quel programmeur de PIC, tel celui décrit par ailleurs dans ce numéro par exemple.

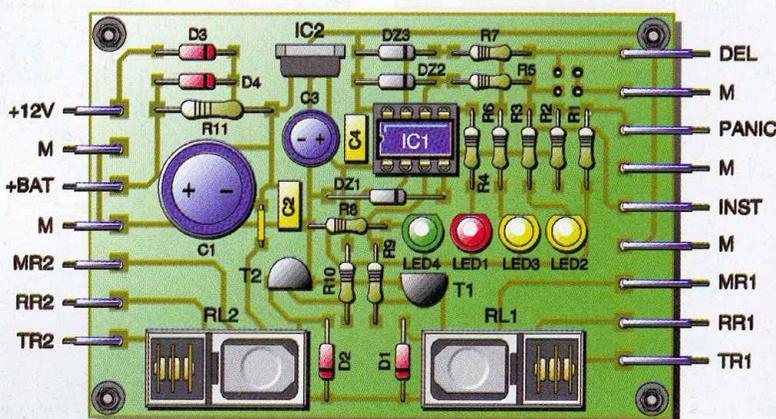
Le câblage de l'alarme est à faire en suivant le plan d'implantation de la **figure 5**, dans l'ordre habituel : support de CI, composants passifs, puis composants actifs. Compte tenu de la faible consommation du montage, le régulateur n'a pas besoin de radiateur. Les LED peuvent être directement soudées sur le circuit imprimé, côté cuivre (contrairement à la maquette où elles ont été soudées côté composants pour des raisons d'esthétique des photos !), ce qui permet de fixer le circuit imprimé derrière la face avant du boîtier recevant l'alarme en faisant dépasser ces LED par des trous judicieusement disposés.

L'interrupteur de mise en marche de l'alarme sera avantageusement constitué par une clé ronde disponible dans le com-



**4** Tracé du circuit imprimé

**5** Implantation des éléments



merce de détail courant des composants électroniques. L'alimentation sera réalisée avec n'importe quel bloc secteur "prise de courant" délivrant 12V sous 250 mA environ.

Si vous utilisez des batteries de sauvegarde, ce qui est vivement conseillé, choisissez des modèles au Cadmium Nickel de façon à disposer d'une tension de 9V environ (par exemple 8 éléments de 1,2V). La capacité sera fonction de la durée de fonctionnement de l'alarme en l'absence de secteur dont vous souhaitez bénéficier. La résistance  $R_{11}$  est à calculer de la façon suivante :

$$R_{11} = 10 \cdot (12 - V_{\text{BATT}}) / C \text{ où :}$$

-  $V_{\text{BATT}}$  est la tension nominale des batteries.

- C est la capacité des batteries exprimée en ampère/heure.

Si l'entrée pour la boucle NO est effectivement utilisée comme entrée "panique", vous pouvez câbler en parallèle dessus autant de poussoirs que vous le souhaitez, répartis en différents points stratégiques des locaux à protéger. Notez également que nous avons prévu la possibilité de mettre en place sur le circuit imprimé deux straps au pas de 2,54 mm permettant de court-circuiter les entrées INST et DEL au cas où l'une ou l'autre ne serait pas utilisée dans votre installation.

## Essais et utilisation

Les essais peuvent être réalisés sur table avec de simples fils volants ce qui est très

pratique. Commencez pour cela par court-circuiter les deux boucles NF (instantanée et retardée) et mettez le montage sous tension. La LED verte et les deux LED jaunes doivent s'allumer et la LED rouge d'alarme doit clignoter rapidement pendant 45 secondes, matérialisant ainsi le délai de la boucle temporisée. Passé ce délai, et si vos deux boucles NF sont toujours en court-circuit, la LED rouge s'éteint et rien ne se passe. L'alarme est en veille.

Si vous fermez la boucle NO (entrée "panique") un court instant (en fait plus de 0,5 seconde compte tenu de l'antiparasite par logiciel réalisé en interne) l'alarme se déclenche immédiatement. La LED rouge clignote lentement ainsi que le relais  $RL_2$  tandis que le relais  $RL_1$  est collé. Si vous attendez 5 mn, les deux relais vont décoller mais la LED rouge continuera de clignoter, mémorisant ainsi le fait qu'une alarme ait eu lieu. Ce clignotement ne cessera qu'à la prochaine séquence d'arrêt puis de mise sous tension du montage.

Si vous ouvrez la boucle NO temporisée, la LED rouge se met à clignoter rapidement mais aucune alarme ne se déclenche pendant les 45 secondes qui suivent et, si cette boucle est refermée à l'issue de ce délai (ou si vous avez arrêté l'alarme parce que vous rentriez tout simplement chez vous) rien ne se passe et la LED rouge s'éteint. Dans le cas contraire, c'est à dire boucle toujours ouverte, l'alarme se déclenche et fonctionne comme nous l'avons vu ci-dessus.

Si vous faites la même manipulation avec la boucle NO instantanée, le même délai de

45 secondes se produit mais, à l'issue de celui-ci, l'alarme se déclenche que la boucle ait été refermée ou non. Notez que si une boucle reste ouverte en permanence, la centrale s'arrête bien d'elle-même au bout des 5 mn de délai et ne se redéclenche pas du fait de cette ouverture permanente. Ceci a été prévu car nous ne connaissons pas beaucoup de voleurs qui referment les portes en quittant les lieux suite au déclenchement d'une alarme. Par contre, si la boucle qui a été ouverte et qui a déclenché l'alarme est refermée puis ouverte à nouveau au-delà des 5 mn de délai, l'alarme est à nouveau déclenchée.

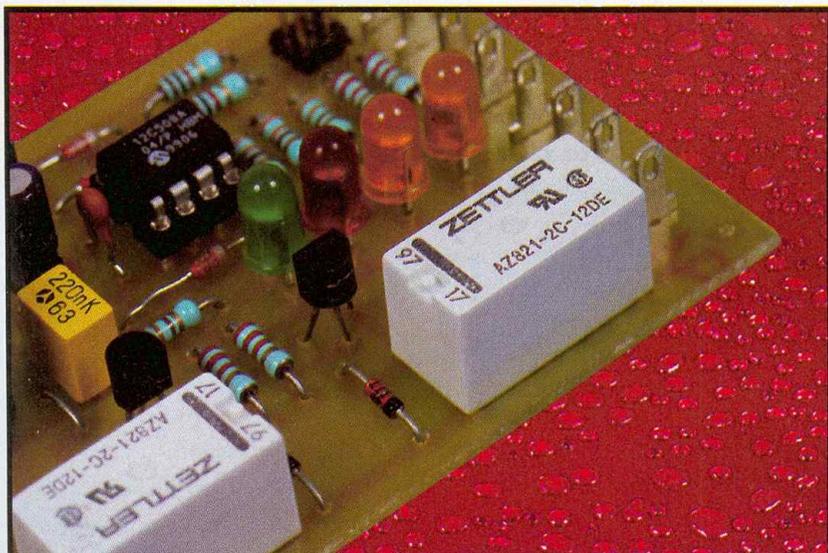
## Conclusion

Malgré sa simplicité, notre centrale d'alarme dispose de toutes les fonctions de modèles nettement plus complexes et encombrants, ceci grâce à l'utilisation judicieuse du 12C508A de MICROCHIP. Même si vous ne la réalisez pas, nous espérons qu'elle vous aura permis de découvrir cette nouvelle génération de microcontrôleurs qui tendent à remplacer de plus en plus souvent de simples circuits logiques, à moindre coût et avec une grande efficacité.

C. TAVERNIER

## Nomenclature

- C<sub>1</sub> : 12C508A programmé (voir texte)**
- IC<sub>2</sub> : 7805**
- T<sub>1</sub>, T<sub>2</sub> : BC547, 548**
- D<sub>1</sub>, D<sub>2</sub> : 1N914 ou 1N4148**
- D<sub>3</sub>, D<sub>4</sub> : 1N4004**
- DZ<sub>1</sub> à DZ<sub>3</sub> : zéner 4,7V/0,4W**
- LED<sub>1</sub> : LED rouge**
- LED<sub>2</sub>, LED<sub>3</sub> : LED jaunes**
- LED<sub>4</sub> : LED verte**
- R<sub>1</sub>, R<sub>2</sub>, R<sub>9</sub>, R<sub>10</sub> : 2,2 kΩ 1/4W 5% (rouge, rouge, rouge)**
- R<sub>3</sub>, R<sub>4</sub>, R<sub>6</sub>, R<sub>8</sub> : 330 Ω 1/4W 5% (orange, orange, marron)**
- R<sub>5</sub>, R<sub>7</sub> : 1 kΩ 1/4W 5% (marron, noir, rouge)**
- R<sub>11</sub> : voir texte**
- C<sub>1</sub> : 220 µF/25V chimique radial**
- C<sub>2</sub> : 0,22 µF mylar**
- C<sub>3</sub> : 10 µF/25V chimique radial**
- C<sub>4</sub> : 1 nF céramique**
- RL<sub>1</sub>, RL<sub>2</sub> : relais miniatures 12V/1RT (FUJITSU FBR 244, SIEMENS V23102 ou équivalents**
- 1 support de CI 8 pattes**



les deux relais miniatures 12V/1RT

# Horloge professionnelle à LED



Une horloge professionnelle à LED se rencontre dans les studios de radio, par exemple. Elle doit être jolie et offrir l'avantage d'afficher clairement les secondes par 60 LED en périphérie. Cette réalisation est, par définition, inaccessible à l'amateur mais nous relevons le défi en vous proposant un montage à la portée de tous, sans réglages et activé par un microcontrôleur 68HC11F1 ; un des moins chers du marché et néanmoins très performant.

## Constitution

Elle est constituée de deux platines. L'une pour l'affichage ne comporte que les LED et une multitude de straps imposée par un circuit simple face ; l'autre, également en simple face, supporte toute l'électronique de commande, l'alimentation et les boutons poussoirs de mise à l'heure. En cas de coupure de courant, une batterie optionnelle, en charge tampon, conserve l'heure en mémoire sans l'afficher, par économie.

## Schéma de principe

Le schéma de la **figure 1** fait apparaître une relative simplicité, compte tenu de la réalisation (pas de mémoire externe ni de circuit d'interface spécialisé).

### L'alimentation

La tension issue du transformateur est redressée par les diodes  $D_1$  et  $D_2$ , filtrée par le condensateur  $C_8$  et découplée par  $C_7$ . Cette tension d'environ 12V est acheminée au régulateur  $CI_4$  à travers la diode anti-

retour  $D_3$  pour donner 5V. Elle subit un dernier filtrage par le condensateur  $C_6$ . La batterie tampon est chargée en permanence, à faible intensité, via la diode  $D_4$  et la résistance  $R_5$ . En cas de manque de la tension du secteur, c'est elle qui alimente le régulateur  $CI_4$  à travers la diode anti-retour  $D_5$ . Le transistor  $T_1$ , polarisé négativement par la résistance  $R_7$ , reçoit sur sa base la tension positive redressée, limitée par la résistance  $R_6$ . Tant que la tension du secteur est présente, il force négativement l'entrée PC7 du  $\mu C$ .

### Les circuits de service du $\mu C$

Le MC68HC11F1 est bien sûr le cœur de notre horloge. Il est cadencé par un quartz de 4,9152 MHz ; cette valeur permet après deux divisions logicielles (par 32768 et par 150) d'obtenir la seconde précise. Le quartz est accompagné des condensateurs  $C_1$ ,  $C_2$  et de la résistance  $R_1$ . La résistance  $R_2$  maintient l'entrée "MODE B" à +VCC en mode "MONOCHIP" ; le cavalier  $J_2$  établit en force le mode "BOOTSTRAP"

pour une éventuelle programmation du  $\mu C$  à même le circuit. La résistance  $R_3$ , le condensateur  $C_4$  et le bouton poussoir RST constituent le circuit d'initialisation. La résistance  $R_4$  et le condensateur  $C_3$  fixent la tension de référence du convertisseur analogique/numérique inutilisé pour cette application.

### Les entrées de contrôle

Les quatre dernières broches du port C remplissent cette tâche. La touche  $S_1$ , en forçant PC4 à la masse, provoque l'avance rapide de l'heure.  $S_2$  sur PC5 agit sur l'avance lente ;  $S_3$  sur PC6 déclenche la mise à zéro des secondes et le fonctionnement normal de l'horloge. Le rôle de PC7, étudié précédemment, surveille la présence du secteur. Au repos, les broches PC0 à PC7 sont positionnées au +VCC par le réseau de résistances RES<sub>1</sub>.

### Les secondes

Le multiplexage est la seule solution permettant la gestion de 60 LED avec si peu de lignes. Le port F se charge de la valeur à afficher sur une ligne en

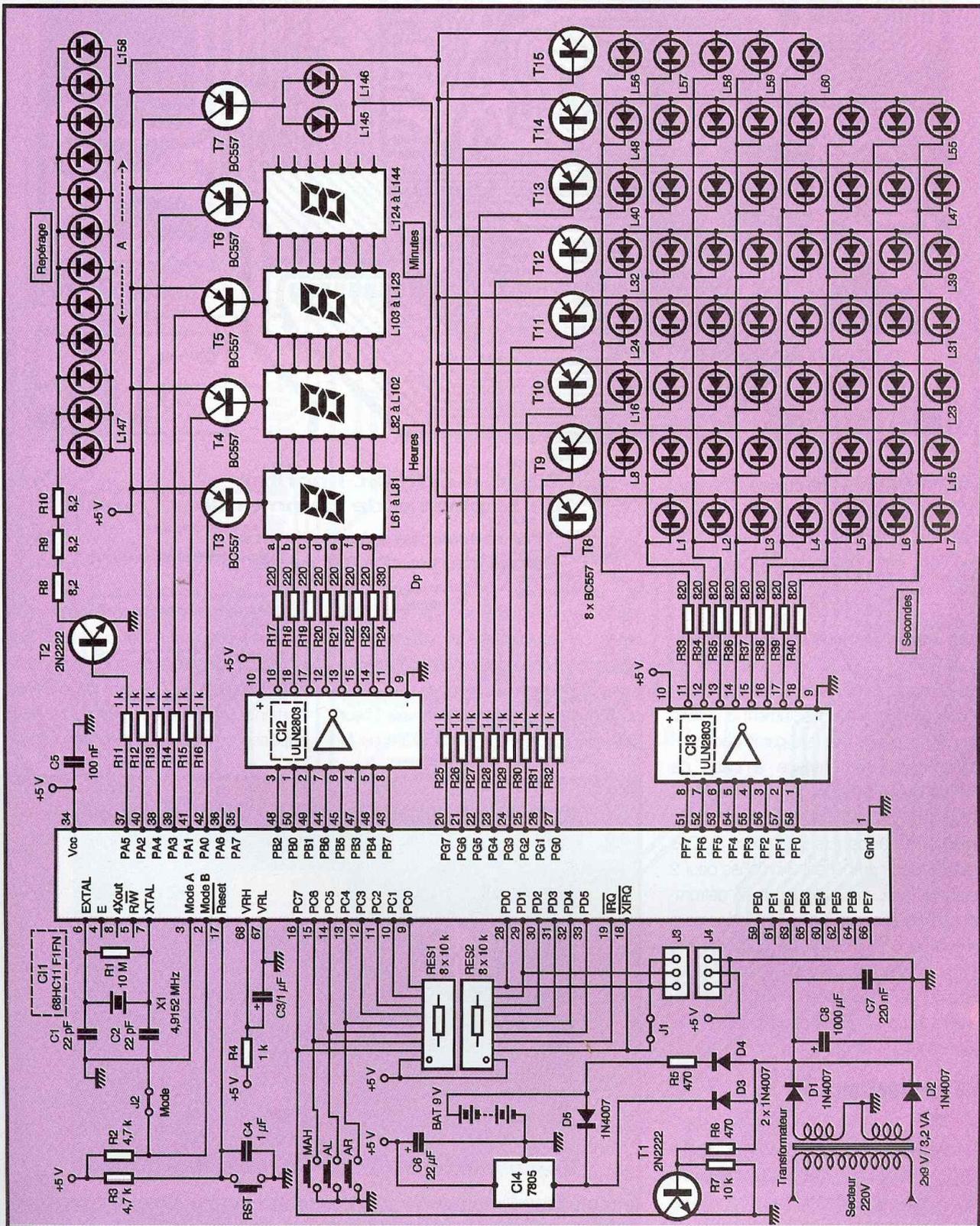
fonction de la colonne balayée par le port G. Le circuit C<sub>13</sub>, constitué de 6 inverseurs de puissance, alimente les cathodes des LED à travers les résistances de limitation R<sub>33</sub> à R<sub>40</sub>. Les anodes sont alimentées par les transistors T<sub>8</sub> à T<sub>15</sub> accompagnés de leurs résistances de base R<sub>25</sub> à R<sub>32</sub>.

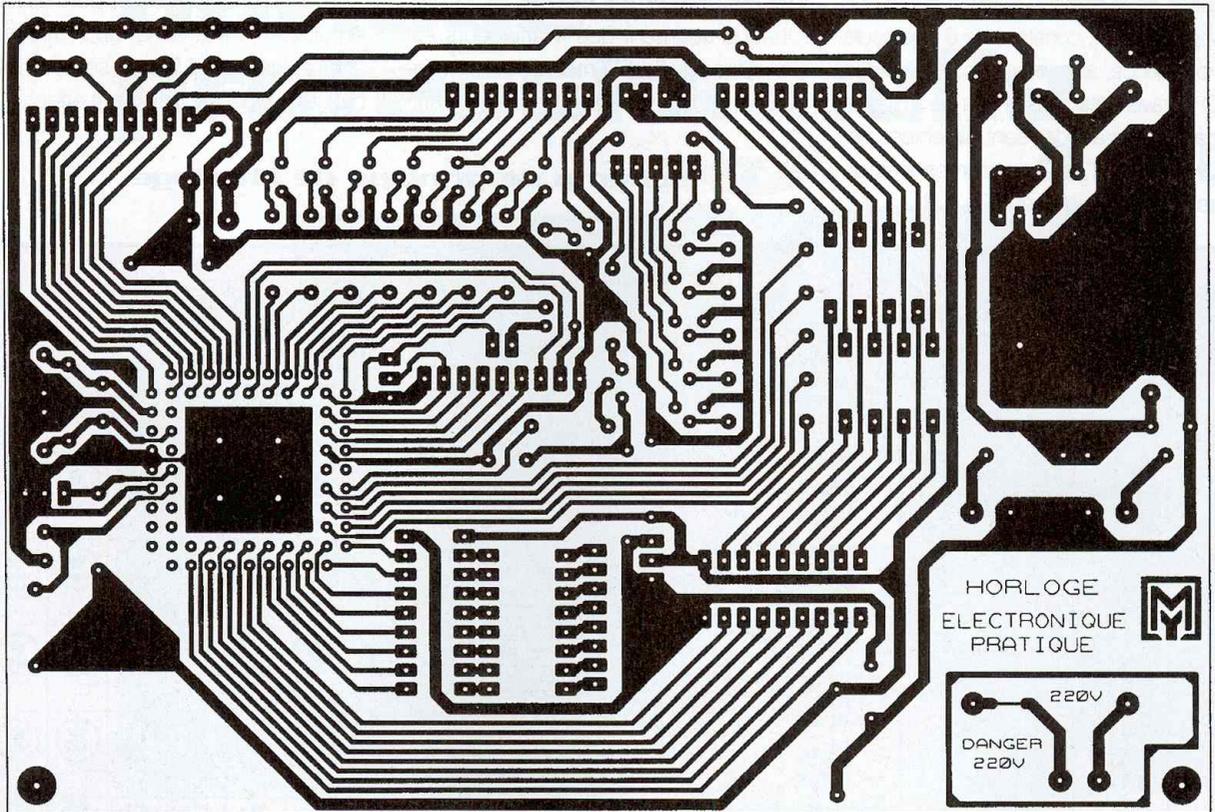
**Les minutes et les heures**

Chaque segment des 4 afficheurs est constitué de 3 LED montées en parallèle. Comme pour les secondes, l'affi-

chage est multiplexé. Le port B donne simultanément la valeur du chiffre à afficher et les deux LED de séparation. Le balayage des afficheurs est effectué par

**1 Schéma de principe de l'horloge**





le port A, ce dernier se charge aussi des 12 LED de repérage montées en parallèle. Le principe d'amplification des signaux est identique à celui des secondes.

### Les autres ports du microcontrôleur

Les ports E et D ne contribuent pas au fonctionnement de l'horloge. Ceux qui souhaitent programmer directement le  $\mu C$  sur la platine, peuvent le faire par le port D en raccordant une interface à base de MAX232 par exemple, sur la prise  $J_3$  pour PD0 et PD1. L'alimentation, +VCC et masse, est disponible sur la prise  $J_4$ . Ces deux prises comportent 3 broches pour 2 signaux ; ceci nous dispense de détrompeur si les prises mâles sont câblées de la même façon. Le cavalier  $J_1$  contribue à la programmation en reliant PD0 et XIRQ. Le réseau de résistances  $RES_2$  porte au +VCC le port D et les deux lignes d'interruption IRQ et XIRQ.

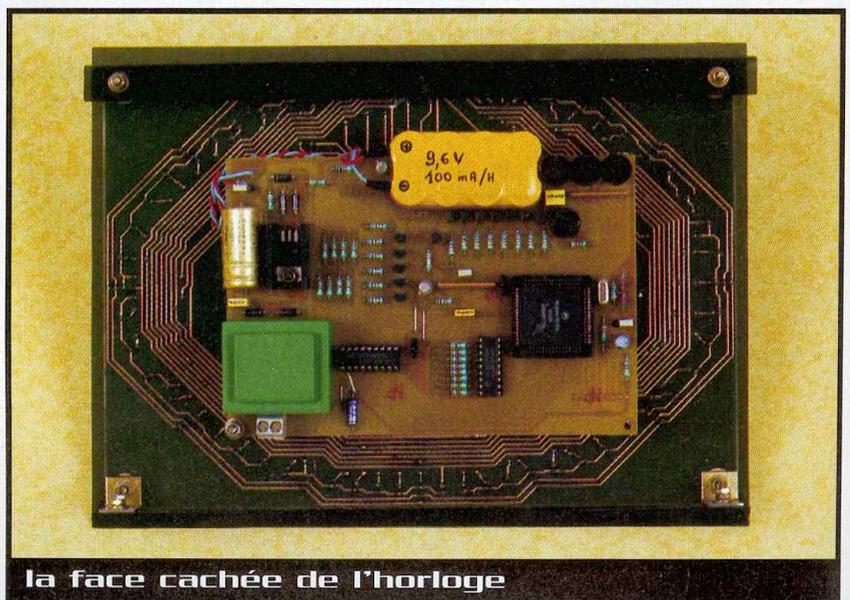
### La réalisation

La **figure 2** donne le dessin du circuit imprimé de la carte d'affichage, la **figure 3** celui de la platine de commande. La méthode de gravure par photo, dans une

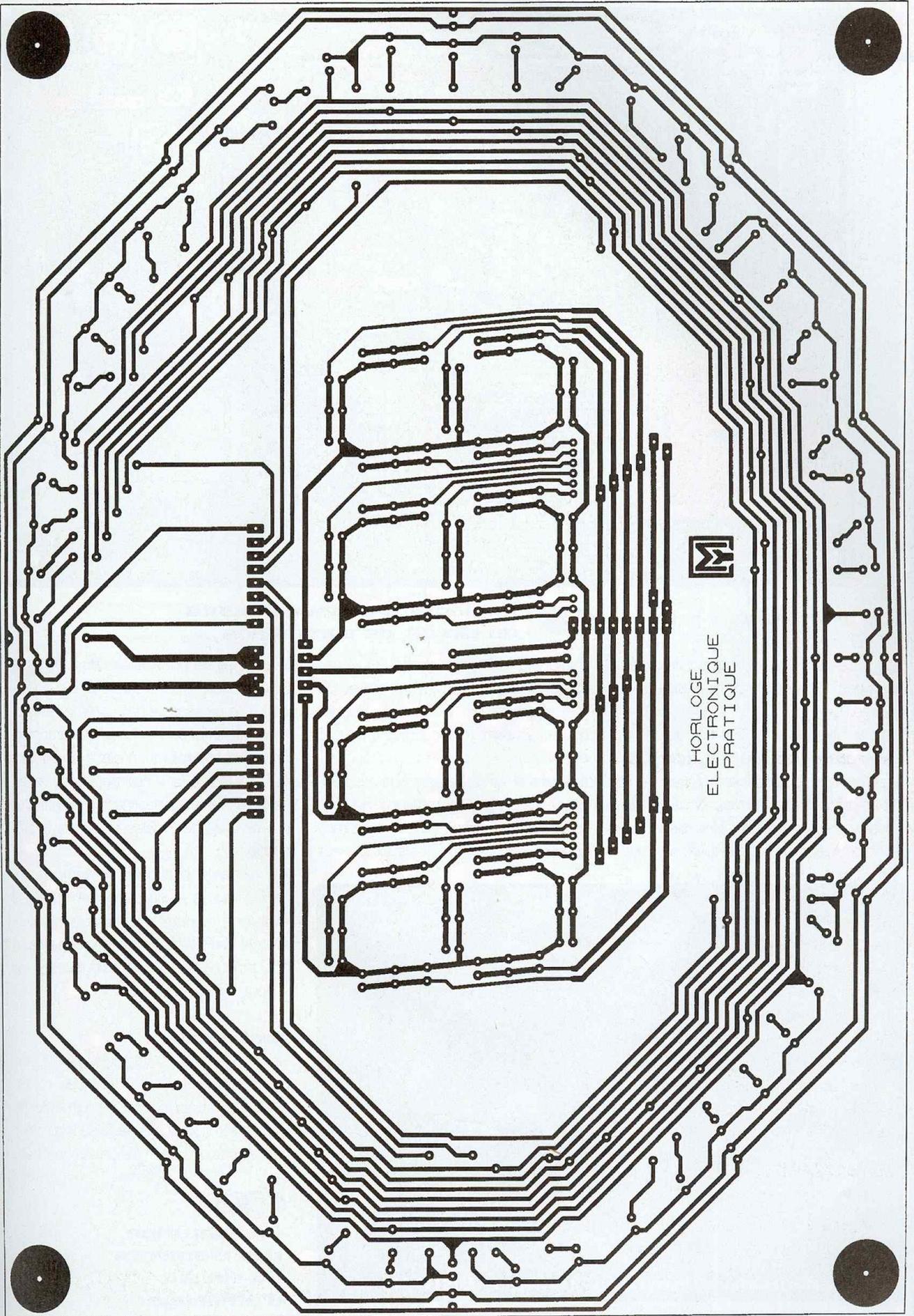
- 2** Tracé du circuit imprimé de la platine de commande
- 3** Tracé du circuit imprimé de la carte d'affichage

insoleuse, s'impose pour de tels montages car les pistes sont vraiment très proches les unes des autres. L'étape suivante consiste à plonger le circuit quelques secondes dans le révélateur, puis dans le bain de perchlorure de fer

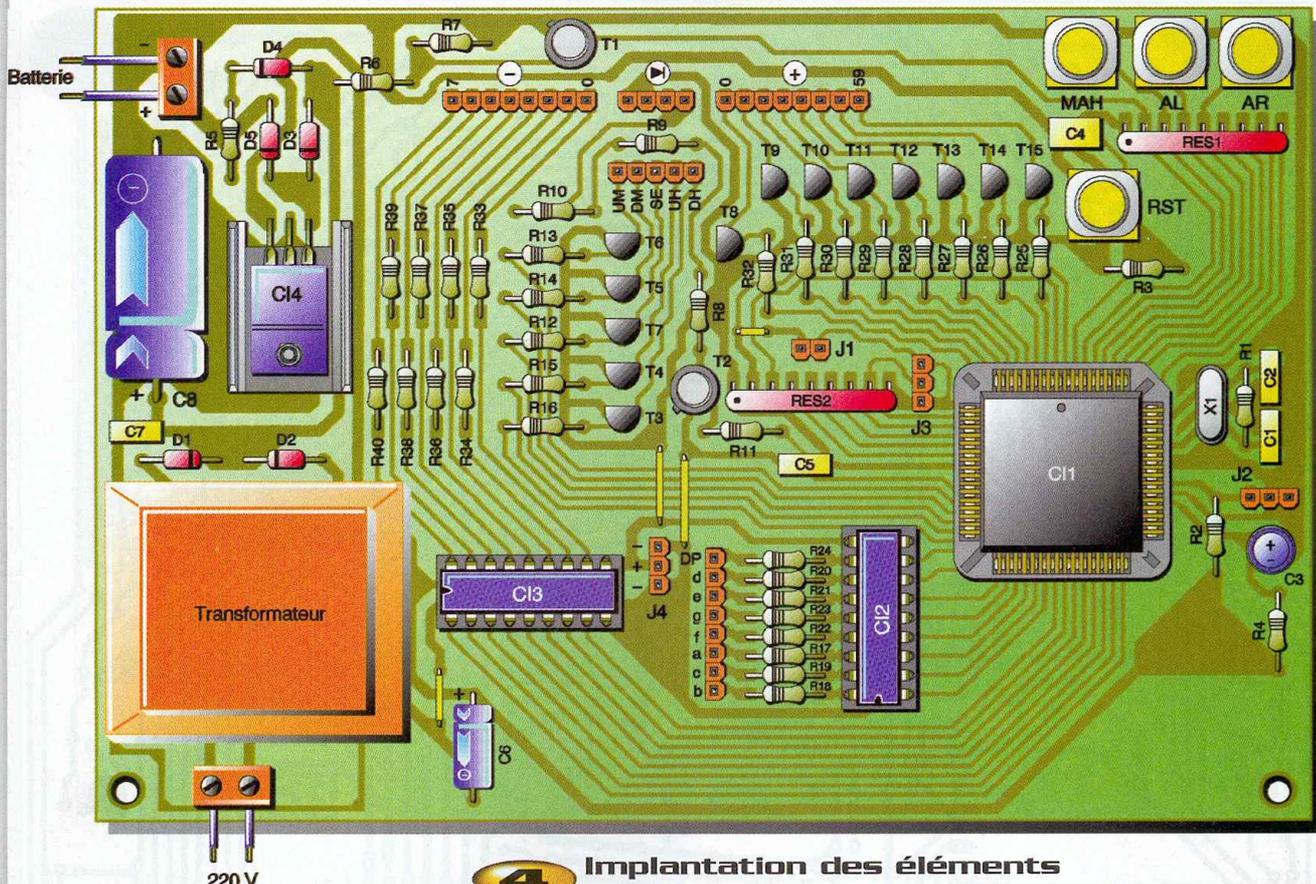
pour l'attaque du cuivre. Il convient de bien rincer la plaque d'époxy à l'eau tiède entre chaque bain. Les trous sont percés avec un foret de 0,8 mm de diamètre, certains sont alésés à des diamètres supérieurs selon les pattes des compo-



la face cachée de l'horloge



HORLOGE  
ELECTRONIQUE  
PRATIQUE



#### 4 Implantation des éléments du circuit de commande

sants que vous utilisez. Ne percez pas les trous des connecteurs sur la plaque d'affichage. L'implantation des composants du circuit d'affichage est donnée à la **figure 4**. Commencez par souder les 79 straps. Afin de donner un aspect professionnel à votre horloge, il est préférable, à ce stade, de peindre à la bombe la face composants en

noir mat avant d'implanter les 12 LED vertes et les 146 LED rouges (la soixantième LED des secondes est facultative). Soudez ensuite, les barrettes sécables mâles du côté cuivré. La **figure 5** donne l'implantation des composants du circuit de commande. Soudez, en premier, les 4 straps et poursuivez traditionnellement par les résistances, les

diodes, les supports de circuits intégrés et les composants par ordre de taille et de fragilité. Les grosses pièces comme le régulateur vissé sur son radiateur et le transformateur seront mises en place en dernier. Soudez, sur la face cuivrée, les barrettes sécables femelles destinées à s'emboîter sur les barrettes mâles du circuit d'affichage. Prenez garde, beaucoup de composants sont polarisés et ne supportent pas les inversions, reportez-vous aux plans et aux photos. Un Plexiglas teinté foncé en façade donnera un aspect professionnel à votre horloge.

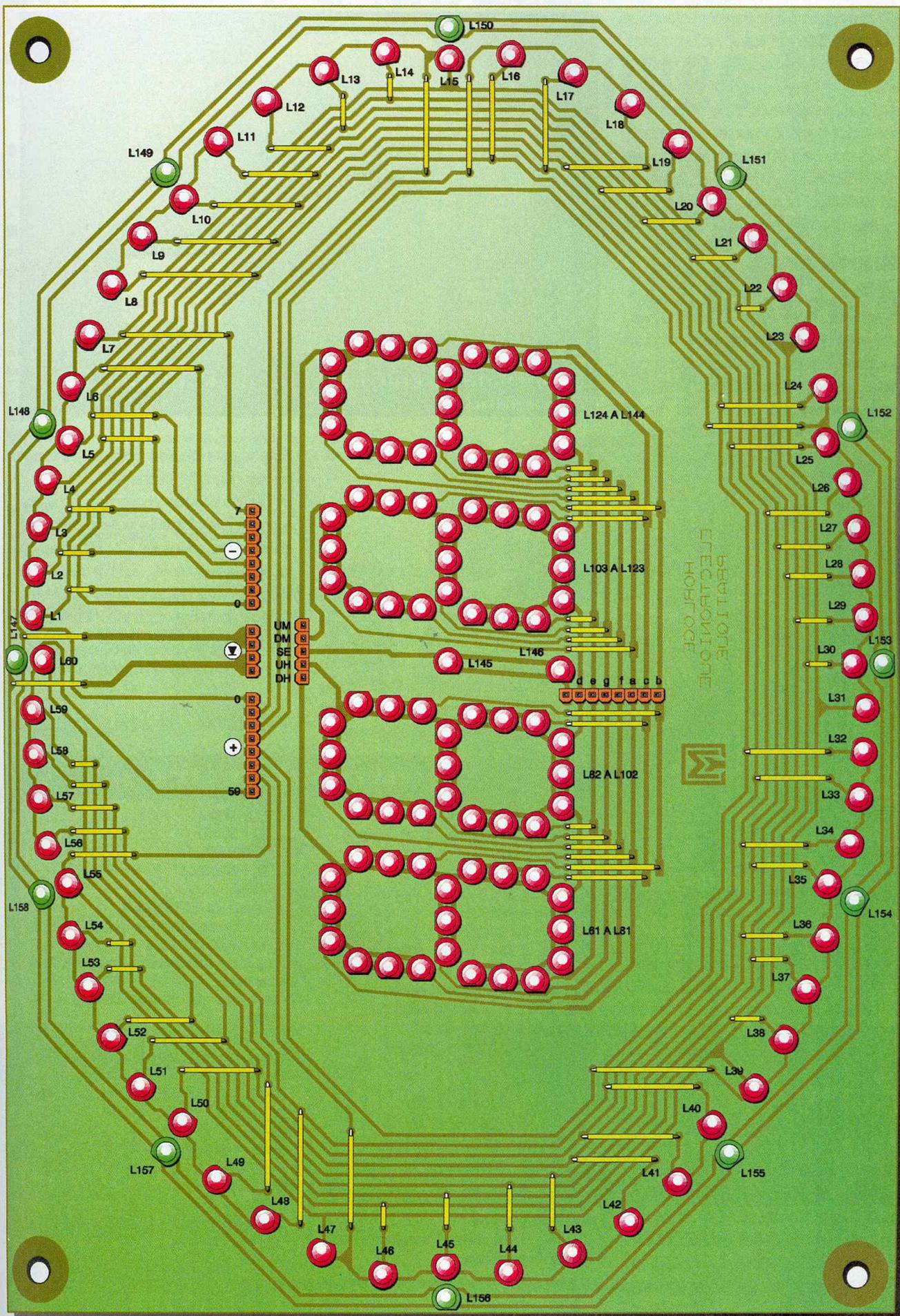
#### Avertissements

Ne négligez pas le radiateur de C<sub>4</sub>, il doit dissiper quand même quelques watts, surtout si vous décidez d'augmenter la luminosité des LED (en réduisant la valeur des résistances de limitation), auquel cas,



utilisation du désormais célèbre 68HC11

#### 5 Implantation des éléments du circuit d'affichage



il faudra opter pour un transformateur plus puissant. Sachez aussi qu'un quartz, comme tout composant, est fabriqué avec une précision relative, ce qui explique une dérive possible de quelques secondes par jour. Pour remédier à cela, il convient, peut être, de régler avec précision l'oscillateur à quartz avec un condensateur ajustable à la place de  $C_1$ , ou  $C_2$ , mais l'opération est délicate.

## Mise en service

Avant toute chose, observez bien à la loupe vos deux circuits afin d'éliminer les risques de courts-circuits entre deux pistes cuivrées. Raccordez ensuite au secteur la carte principale dépourvue de ses circuits intégrés pour contrôler la présence de +VCC par rapport à la masse en divers points et, notamment, sur les supports de circuits intégrés et sur le connecteur  $J_4$ . Hors tension et après une minute pour la décharge de  $C_{B1}$ , embrochez les circuits  $Cl_2$  et  $Cl_3$ . Pour  $Cl_1$ , voyez le paragraphe suivant.

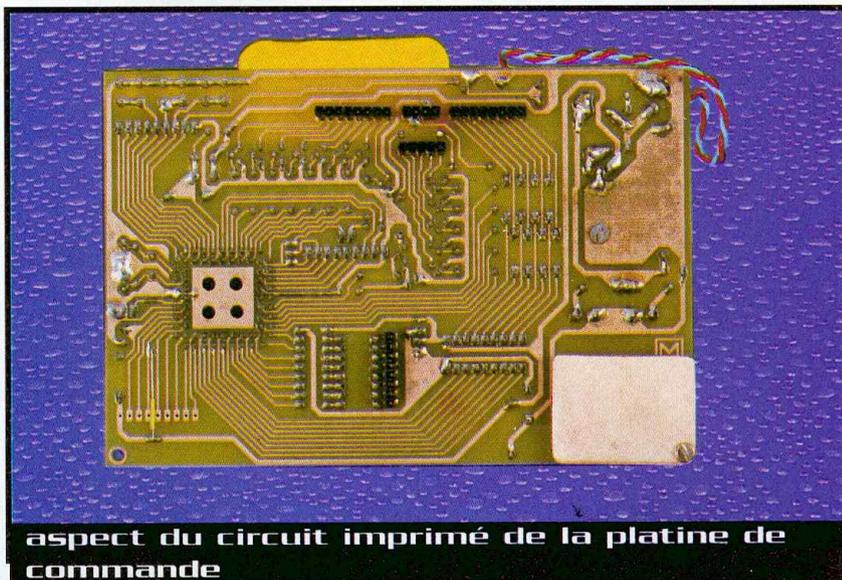
## Programmation

Le 68HC11F1 contient 1K d'octets de mémoire RAM et 512 octets de mémoire EEPROM à effacement électrique qui nous intéressent tout particulièrement. Ensuite, le  $\mu C$  tournera en mode "MONO-CHIP" bien sûr ! Le cavalier  $J_1$  reste en place puisque nous programmons de la EEPROM. Vous pouvez laisser le 68HC11F1 à sa place pour cette opération via une interface spécialisée (voir paragraphe sur le schéma), sinon programmez-le au moyen d'un des montages proposés récemment dans Électronique Pratique.

Le programme est disponible sur notre site Internet "www.eprat.com" ou sur simple demande à la rédaction contre une disquette formatée sous enveloppe affranchie, auto-adressée.

## Le programme

Il est développé au moyen de l'assembleur "AS11" faisant partie, avec le débogueur, de l'ensemble "BASIC11"; mais tout autre assembleur pour 68HC11 peut faire l'affaire. Le programme utilise la totalité de la



aspect du circuit imprimé de la platine de commande

mémoire EEPROM et tolère peu de modifications.

## Utilisation

A la mise sous tension après une action sur la touche RST, vous devez mettre l'horloge à l'heure. Une impulsion sur la touche AR provoque le défilement rapide de l'heure, une action sur la touche AL ralentit nettement l'avance et, pour finir, la touche MAH stoppe le défilement pour un cadencement normal avec remise à zéro des secondes.

Si vous êtes allé trop loin, refaites le processus depuis la touche AR.

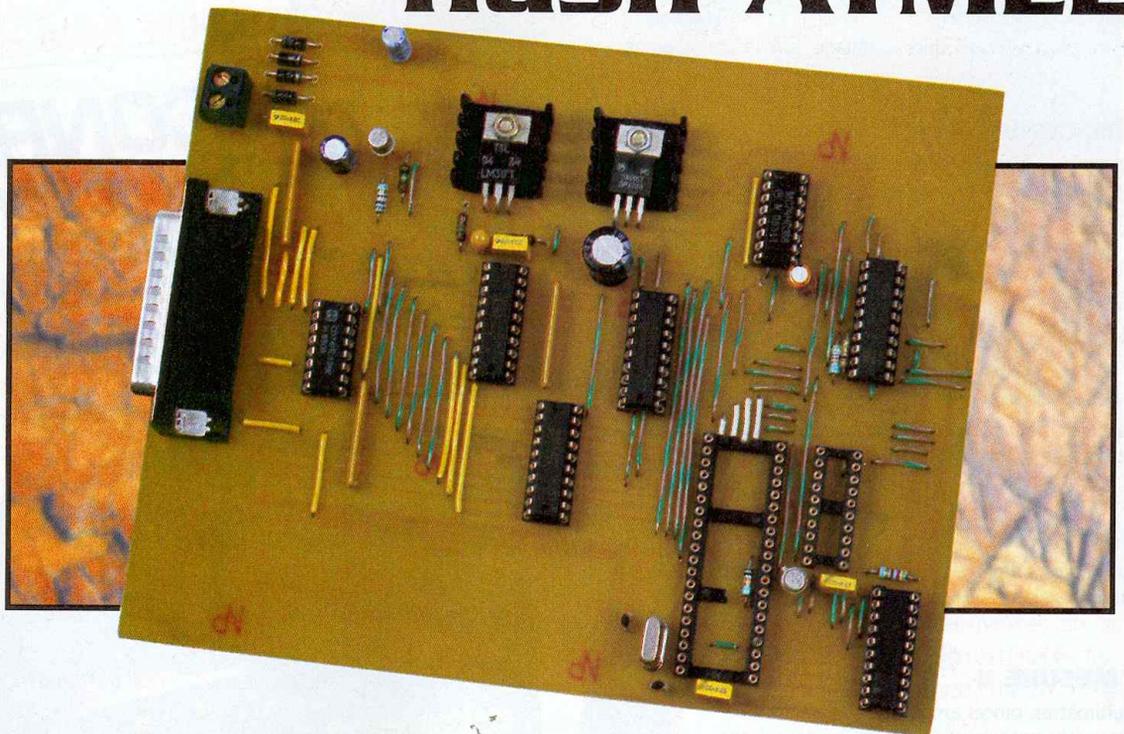
En cas de coupure du secteur, la batterie prend le relais instantanément, l'affichage est inhibé et seules les 12 LED vertes de repérage clignotent au rythme de la seconde. Quand la tension est rétablie, l'affichage normal réapparaît, l'heure a été maintenue malgré la coupure.

Y. MERGY

## Nomenclature

$R_1$ : 10 M $\Omega$ (marron, noir, bleu)	$T_1, T_2$ : 2N2222
$R_2, R_3$ : 4,7 k $\Omega$ (jaune, violet, rouge)	$T_3$ à $T_{15}$ : BC557
$R_4, R_5, R_{11}$ à $R_{16}, R_{25}$ à $R_{32}$ : 1 k $\Omega$ (marron, noir, rouge)	$Cl_1$ : M68HC11FN
$R_6$ : 470 $\Omega$ (jaune, violet, marron)	$Cl_2, Cl_3$ : ULN2803
$R_7$ : 10 k $\Omega$ (marron, noir, orange)	$Cl_4$ : 7805
$R_8$ à $R_{10}$ : 8,2 $\Omega$ (gris, rouge, or)	$X_1$ : quartz de 4,9152 MHz
$R_{17}$ à $R_{23}$ : 220 $\Omega$ (rouge, rouge, marron)	1 transformateur moulé pour circuit imprimé 2x9V/3,2VA
$R_{24}$ : 330 $\Omega$ (orange, orange, marron)	1 support de circuit intégré PLCC à 68 broches
$R_{33}$ à $R_{40}$ : 820 $\Omega$ (gris, rouge, marron)	2 supports de circuit intégré à 18 broches
RES <sub>1</sub> , RES <sub>2</sub> : 8x10 k $\Omega$ (9 broches)	1 bornier à 2 bornes
$C_1, C_2$ : 22 pF (céramiques)	1 batterie 9V (format pile) et son connecteur
$C_3$ : 1 $\mu F$ /25V (électrochimique à sorties radiales)	1 petit dissipateur thermique horizontal
$C_4$ : 1 $\mu F$ (mylar jaune)	4 touches à 1 contact travail
$C_5$ : 100 nF (mylar jaune)	Barrettes sécables femelles (38 broches)
$C_7$ : 220 nF (mylar jaune)	Barrettes sécables mâles (39 broches)
$C_8$ : 1000 $\mu F$ /16V (électrochimique à sorties axiales)	2 cavaliers de configuration pour barrette mâle
$D_1$ à $D_5$ : 1N4007	
$L_1$ à $L_{146}$ : LED rouges 5 mm haute luminosité de préférence	
$L_{147}$ à $L_{158}$ : LED vertes 5 mm haute luminosité de préférence	

# Programmateur de microcontrôleurs flash ATMEL



Nous avons eu l'occasion de publier plusieurs montages à base de microcontrôleurs ATMEL ce qui nous a valu de recevoir de nombreuses questions sur la façon de les programmer. Cet article répond à ces questions en vous donnant ce moyen, à condition bien sûr de posséder un compatible PC et de brancher sur son port parallèle le montage que nous vous proposons.

La conception de ce programmeur n'a pas exigé beaucoup de réflexion, en effet sur le site [www.atmel.com](http://www.atmel.com) tout est disponible y compris schéma et logiciel. Pour ceux qui voudraient en savoir d'avantage, la note d'application concernant cette application est très bien documentée si toutefois vous maîtrisez l'Anglais. Il est capable de programmer les composants à 40 et 20 broches de la série des AT89C51/C52/LV51/LV52/C1051/C2051.

L'intérêt pour ce programmeur est d'autant plus important qu'il dispose d'un logiciel puissant permettant la lecture/écriture, sauvegarde, effacement du composant, verrouillage du bit de sécurité et lecture de la signature fabricant.

Le seul regret sera sûrement pour les adeptes de l'interface graphique de Windows, en effet le logiciel tourne sous DOS avec l'obscur écran à fond noir, mais peu importe, efficacité avant tout.

Le tableau suivant vous donne un

résumé du menu que vous aurez à l'écran :

Chip Erase	(D)
Program from File	(F)
Verify against File	(G)
Save to File	(H)
Blank Check	(I)
Read Signature	(J)
Write Lock Bit 1	(L)
Write Lock Bit 2	(N)
Exit	(X)

Voici un résumé de ces commandes

- Chip Erase : il efface entièrement le contenu de la mémoire FLASH du microcontrôleur, tous les bits sont positionnés à 1, l'exécution de cette commande n'est pas vérifiée automatiquement par le programme.

- Program from File : il écrit le contenu du fichier spécifié dans la mémoire du microcontrôleur. Vous serez invités à saisir le nom du fichier avec l'extension et chemin du répertoire.

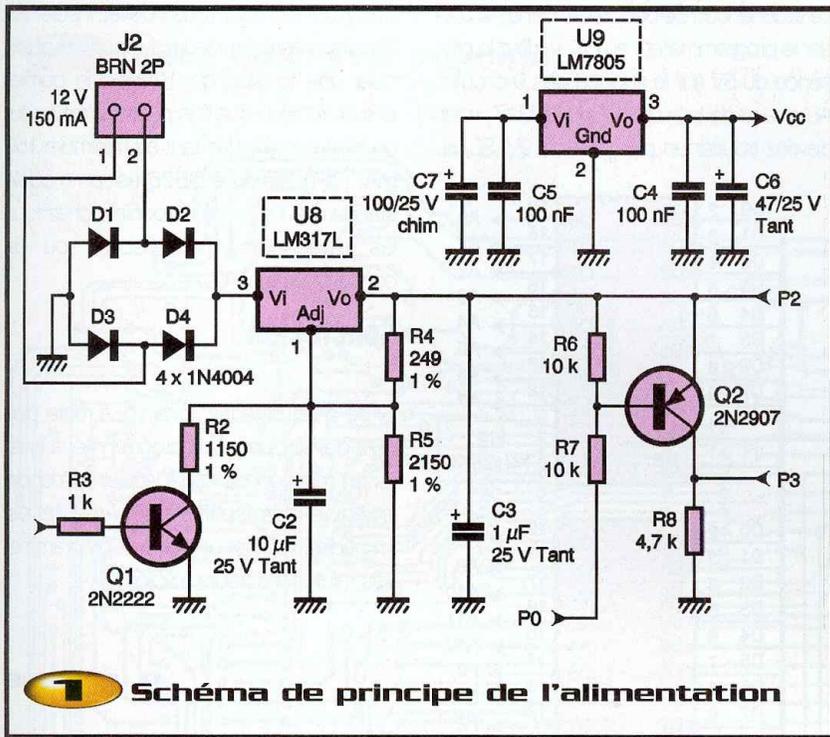
Le fichier INTEL HEXA n'étant pas accepté, il devra être uniquement un fichier binaire. Après la programmation le contenu de la mémoire du microcontrôleur n'est pas vérifié avec celui du fichier. Il n'y a pas non plus d'indication visuelle de la programmation en cours, mis à part que le retour au menu général.

- Verify against File : compare le contenu mémoire avec le fichier spécifié tout comme la programmation, le nom et le chemin seront à saisir.

Le premier octet du fichier binaire est comparé avec le premier octet de la mémoire et ainsi de suite. Si une erreur de comparaison est trouvée, l'adresse de cet octet ainsi que son contenu seront affichés à l'écran. Dans le cas contraire, si la comparaison est bonne, le programme retourne au menu principal.

- Save to File : comme son nom l'indique, le nom que vous donnerez au fichier sera enregistré dans le répertoire désigné.

- Blank Check : vérifie que les bits de



**1 Schéma de principe de l'alimentation**

chaque octet mémoire soient à un. Seuls deux messages seront retournés : vérification bonne ou mauvaise. L'adresse de l'octet défectueux n'est pas non plus retournée.

- Read Signature : cette commande retourne et affiche la signature interne propre à chaque type de microcontrôleurs. Le tableau suivant donne la signature de quelques microcontrôleurs ATMEL :

	AT89C51	AT89C52	AT89C1051	AT89C2051
Número fabricant (ATMEL)	0x1E	0x1E	0x1E	0x1E
signature	0x51	0x52	0x11	0x21

- Write Lock Bit 1 :

- Write Lock Bit 2 : ce menu permet de positionner le bit de sécurité au degré de protection désiré contre les relectures.

Ce menu général est identique pour chaque type de microcontrôleurs, l'appel du programme se fait en ligne de commande sous DOS en spécifiant le port LPT1 ou LPT2.

Par exemple, si vous voulez accéder au menu de l'AT89C1051, il vous faudra taper "PC1051 LPT1" ou "PC52-3 LPT1" pour un AT89C52.

Si le port parallèle n'est pas spécifié, le programme vous retourne un message d'erreur.

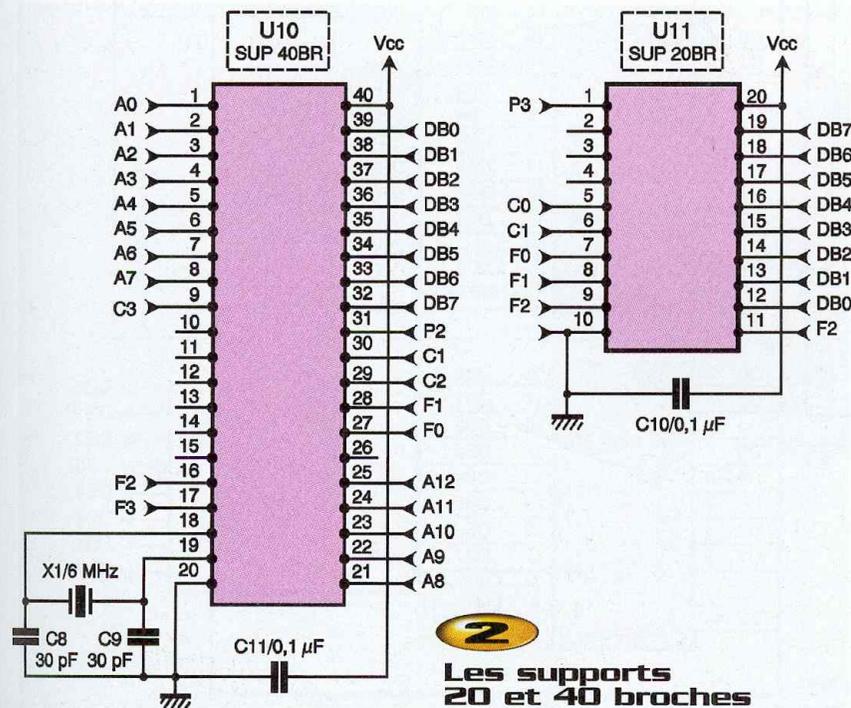
Pour ceux qui sont intéressés, sachez que les sources en langage C de ces programmes vous sont aussi fournis sur le site ATMEL, rien ne vous empêche d'inclure les différents menus dans un seul et même programme. Des compilateurs comme VISUAL C++ ou C++BUILDER de Borland permettent aisément la chose avec des menus beaucoup plus attrayants sous Windows 98. Vous pourrez également vous procurer les exécutables et les sources de ce programmeur sur notre site eprat.com. Il se résume à un seul fichier compressé nommé "apcpgm.exe", une fois lancé, il se défragmente en deux fichiers appelés PCPGM.EXE et READ1.ME. Il vous suffira alors de relancer pcpgm.exe pour qu'il se défragmente à son tour en une multitude de fichiers exécutables et sources.

## Réalisation

Le circuit imprimé (figure 4) étant en simple face des straps sont nécessaires. Un soin particulier est demandé lors de la soudure des supports de C.I qui pourront être des tulipes, voire des supports standards.

Les diodes, régulateur et borniers devront être percés à un diamètre de 1mm, quant au reste des pastilles, elles seront percées à 0,8mm.

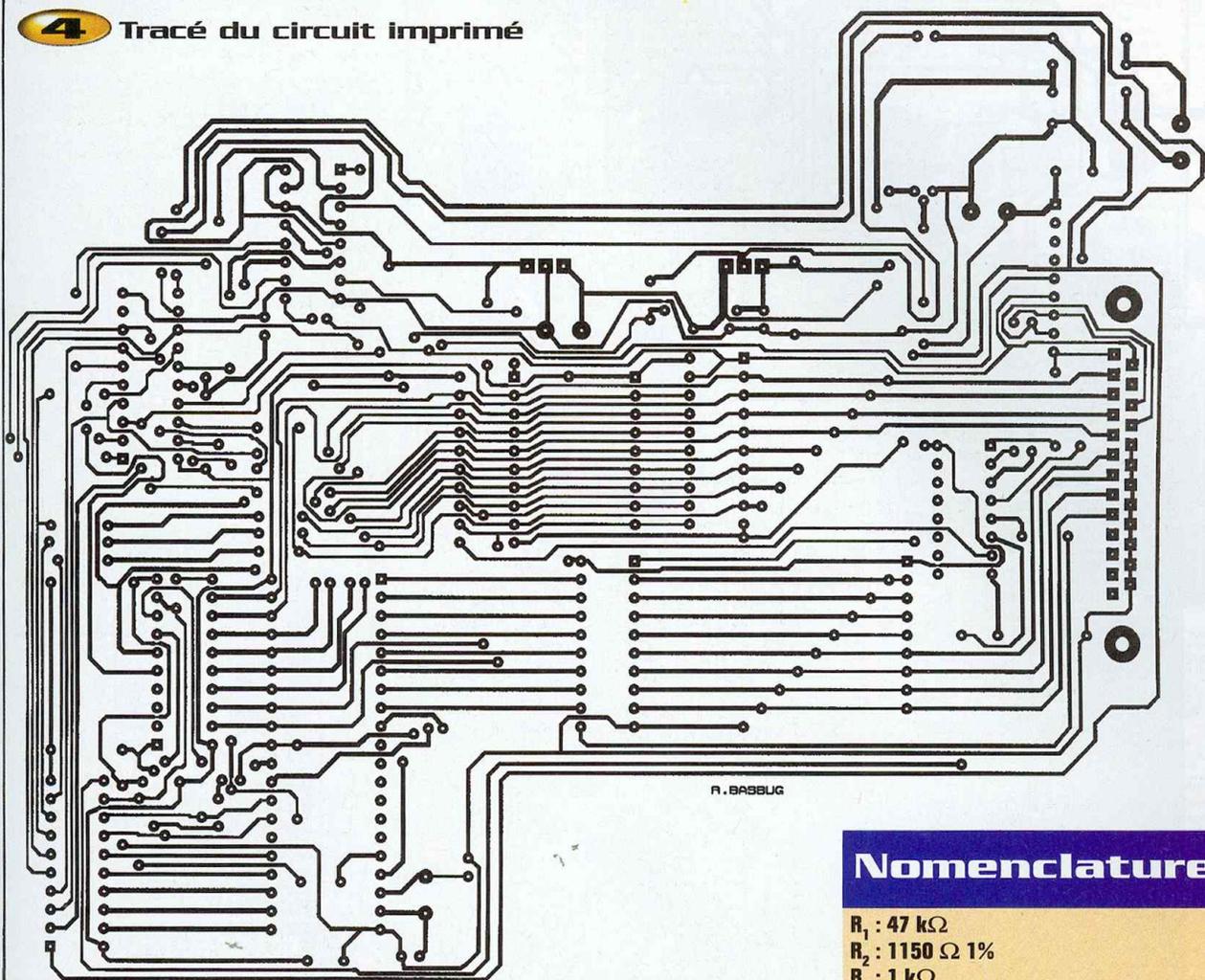
Après la mise en place des straps, on plantera les résistances, les diodes et les supports de circuits intégrés (figure 5). Des supports à force d'insertion nulle sont quasiment obligatoires pour IC<sub>10</sub> et IC<sub>11</sub>.



**2 Les supports 20 et 40 broches**



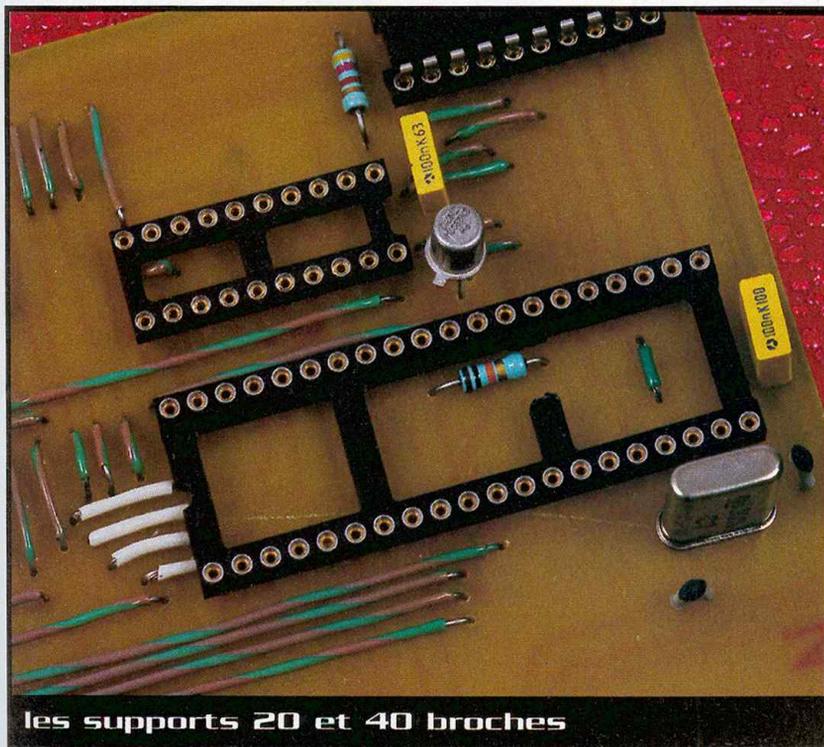
**4** Tracé du circuit imprimé



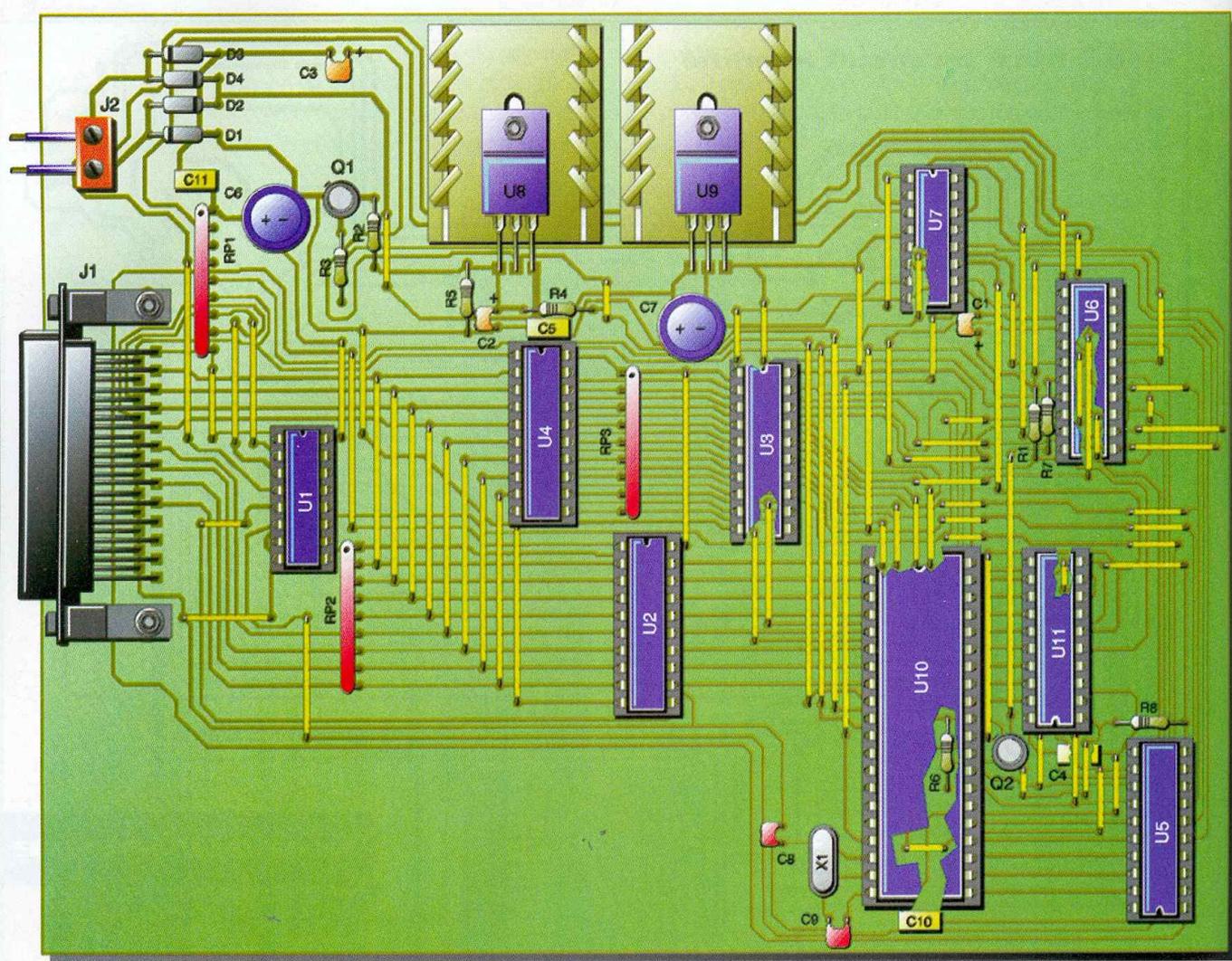
R. BASBUG

**Nomenclature**

- R<sub>1</sub> : 47 kΩ
- R<sub>2</sub> : 1150 Ω 1%
- R<sub>3</sub> : 1 kΩ
- R<sub>4</sub> : 249 Ω 1%
- R<sub>5</sub> : 2150 Ω 1%
- R<sub>6</sub>, R<sub>7</sub> : 10 kΩ
- R<sub>8</sub> : 4,7 kΩ
- RP<sub>1</sub>, RP<sub>2</sub> : 4,7 kΩ
- RP<sub>3</sub> : 10 kΩ
- C<sub>1</sub> : 1 μF
- C<sub>2</sub> : 10 μF/25V tantale
- C<sub>3</sub> : 1 μF/25V tantale
- C<sub>4</sub>, C<sub>5</sub> : 100 nF
- C<sub>6</sub> : 47/25 V tantale
- C<sub>7</sub> : 100/25V chimique
- C<sub>8</sub>, C<sub>9</sub> : 30 pF
- C<sub>10</sub>, C<sub>11</sub> : 0,1 μF
- U<sub>1</sub> : 74HC139
- U<sub>2</sub>, U<sub>3</sub>, U<sub>5</sub> : 74HC574
- U<sub>4</sub> : 74HC541
- U<sub>6</sub> : 74HC273
- U<sub>7</sub> : 74HC368
- U<sub>8</sub> : LM317L
- U<sub>9</sub> : LM7805
- U<sub>10</sub> : SUP\_40BR
- U<sub>11</sub> : SUP\_20BR
- Q<sub>1</sub> : 2N2222
- Q<sub>2</sub> : 2N2907
- D<sub>1</sub> à D<sub>4</sub> : diodes 1N4004
- J<sub>1</sub> : connecteur DB25
- J<sub>2</sub> : bornier 2 points
- X<sub>1</sub> : 6 MHz

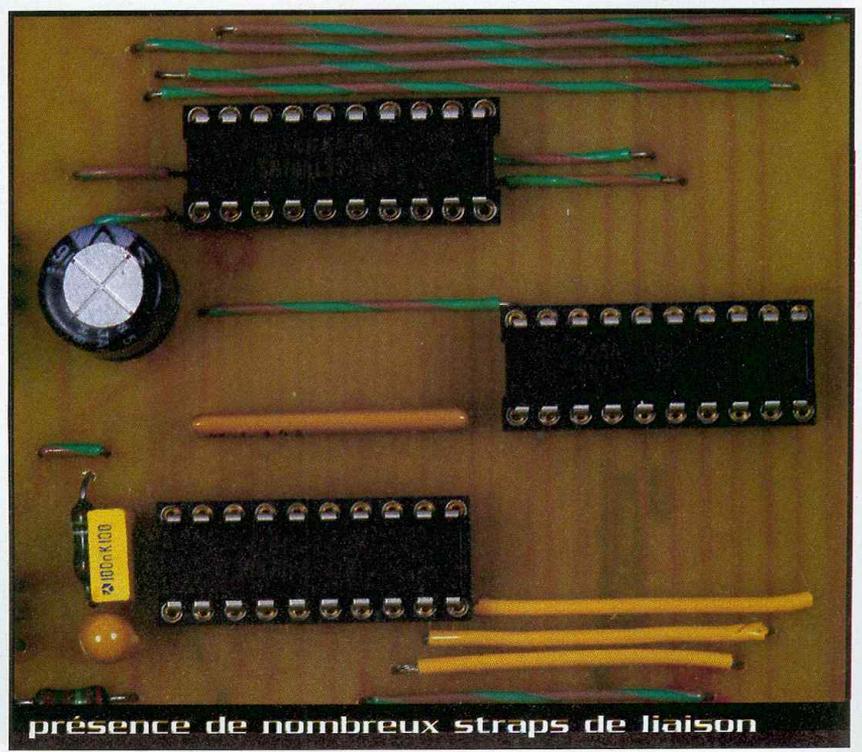


les supports 20 et 40 broches

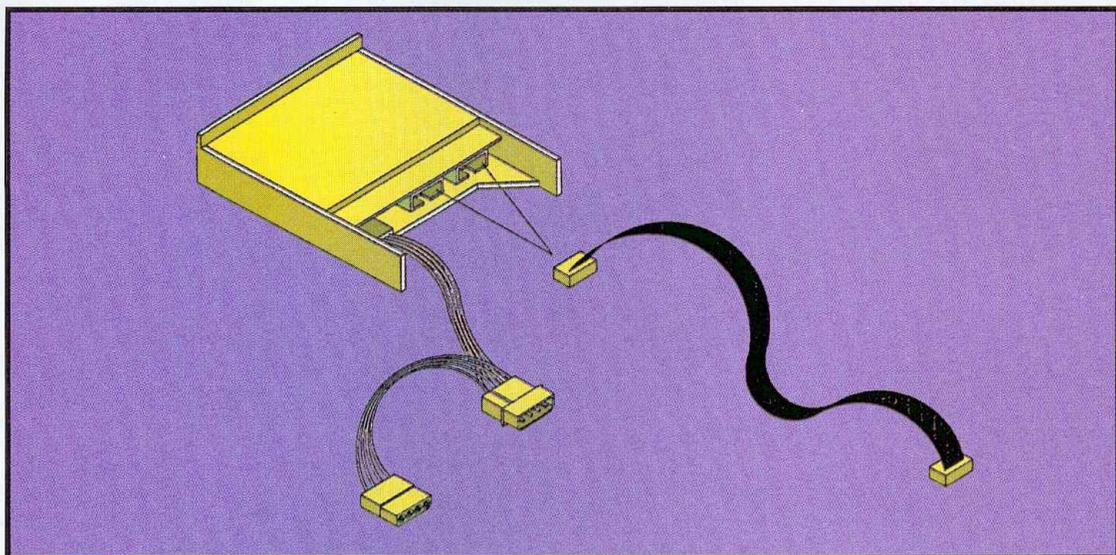


**5**

**Implantation des éléments**



# Le «CHIFI», unité d'encodage et de lecture des cartes à puces (Basic-Card)



L'utilisation des cartes à puces a pris, ces dernières années, une très grande ampleur. Cette ampleur s'explique par le fait que cette utilisation rend d'innombrables services et simplifie de très nombreuses opérations très diverses qui concernent aussi bien le domaine des communications, des banques que celui de la santé, avec la carte récemment diffusée par l'organisme de la sécurité sociale.

Ceux que les cartes à puce intéressent savent que programmer une carte à microprocesseur était jusqu'alors une tâche fastidieuse qui nécessitait de multiples opérations. Cela nécessitait également un matériel de développement très onéreux. La situation a maintenant changé. Les cartes sont facilement disponibles. Ce type de cartes utilise un interpréteur appelé P-CODE. Il suffit d'écrire un programme sous JAVA ou BASIC qui est ensuite compilé sous P-CODE et donne un programme ressemblant au langage machine. Il ne reste plus qu'à le télécharger dans la carte où il est exécuté par l'interpréteur. Autre avantage : si le programme comporte des erreurs, il suffit de rectifier le programme et de le charger à nouveau dans la puce.

C'est ce que nous propose la société HI TECH TOOLS qui commercialise deux lecteurs/programmeurs de cartes à puces : les modèles CHIFI interne et externe à

## 1 Connexion à la carte mère

l'ordinateur. L'avantage que présente le modèle interne est qu'il occupe un espace restreint puisqu'il se monte dans un emplacement réservé à un lecteur de disquette 3,5 pouces. Sa connexion à la carte mère de l'ordinateur s'effectue très simplement par un câble en nappe tandis que son alimentation est assurée par le même type de connecteur que ceux que l'on rencontre habituellement (figure 1). Ses caractéristiques, données ci-dessous, sont très intéressantes, les types de cartes pouvant être manipulés étant nombreux :

Programmation en BASIC (logiciel inclus)  
PC/PS compatible avec interface API de Microsoft

Cartes à mémoire synchrone :

- protocole deux fils (SLE 4432/4442)
- protocole trois fils (SLE 4418/4428)

- bus I2C
- extension bus I2C jusqu'à 32 kBytes
- Carte asynchrone à microprocesseur :
- T0 (carte bancaire)
- T1 (Eurochip)
- T1.4 (carte GSM avec interface GSM intégrée)

Adaptation automatique au baud rate de la carte (1200 bauds - 38400 bauds)

Interface RS232-C (V24)

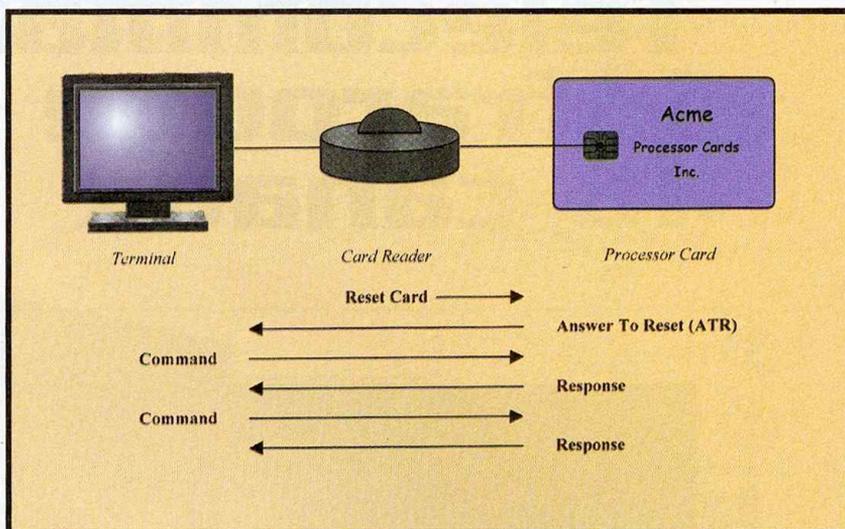
Alimentation par piles pour CHIFI EXTERN (autonome)

Pas d'alimentation externe pour CHIFI  
Afficheur LED d'indication de statut  
Lecteur carte avec zone de contact dorée

La communication entre l'ordinateur et la carte à microprocesseur (bien entendu via le lecteur/programmeur) s'effectue très simplement au moyen d'un protocole

de commande/réponse dont le schéma synoptique est donné en **figure 2**. La carte est le partenaire passif dans cet échange. Après avoir envoyé le signal ANSWER TO RESET (mise à zéro), elle ne fait plus rien jusqu'à ce qu'elle reçoive une commande. Après avoir répondu à cet ordre, elle se remet en état d'attente.

Le langage BASIC utilisé (ZC-BASIC), n'a rien à envier aux langages de ce type puisqu'il contient toutes les instructions nécessaires à l'élaboration d'un programme digne de ce nom. Sans trop entrer dans le détail, nous dirons que pratiquement toutes les instructions existent, auxquelles ont été ajoutées des instructions spécifiques aux cartes à puce.



## 2 Synoptique du montage

Abs	Access	And	Append	ApplicationID
As	Asc	At	ATR	AutoEncrypt
Base	BasicCard	Binary	ByRef	Byte
ByVal	Call	CardReader	CardInReader	Case
Chr\$	Close	Command	Const	Declare
DefByte	DefInt	DefLng	DefSng	DefString
Dim	Disable	Do	Dynamic	Eeprom
EepromLog	Else	Elseif	Enable	End
EOF	Erase	Exit	Explicit	For
FreeFile	Function	Get	GoSub	GoTo
Hex\$	If	InKey\$	Input	Input\$
Integer	Is	Key	KeyFile	LBound
LCase\$	Left\$	Len	Let	Line
Lock	Long	Loop	LTrim\$	Mid\$
Mod	Next	Not	On	Open
Option	Or	Output	OverflowCheck	Peek
Poke	Polynomials	Print	Private	Public
Put	Random	Read	ReDim	Rem
ResetCard	Return	Right\$	Rnd	RTrim\$
Seek	Select	Shared	Single	Space\$
Spc	Sqrt	Static	Step	Str\$
String	String\$	Sub	Tab	Then
To	Trim\$	Type	UBound	UCase\$
Until	Val!	Val&	ValH	Wend
While	Write	Wtx	Xor	

-	tiret bas (continuer une ligne)	'	Apostrophe
(	parenthèse gauche	)	Parenthèse droite
+	plus	-	Moins
*	multiplier	/	Diviser
,	virgule	:	deux points
=	égalité	<>	Différent de
<	plus petit que	>	plus grand que
<=	plus petit ou égal à	>=	plus grand ou égal à
.	point	#	ordre au pré-processeur ou numéro de fichier
:	point virgule		



### Liste des instructions nécessaires au programme

Le **tableau 1** ci-contre représente la liste de ces instructions, le **tableau 2** donne les signes mathématiques et de ponctuation.

N'oublions pas de signaler, chose pratique, que le langage est disponible sur un site Internet sous l'adresse :

[www.zeitcontrol.de/basic.zip](http://www.zeitcontrol.de/basic.zip).

Ce logiciel est libre et peut être utilisé comme bon vous semble.

Ce produit est attrayant par les possibilités qu'il offre et par le prix auquel il est vendu, tout au moins pour le modèle interne.

Le CHIPI interne est vendu au prix de 1590,00 F. HT, tandis le modèle externe est commercialisé au prix de 2490,00 F. HT.

### HI TECH TOOLS

40 rue Saint-André

72000 LE MANS

Tél. : 02.43.28.15.04

Fax : 02.43.28.59.61

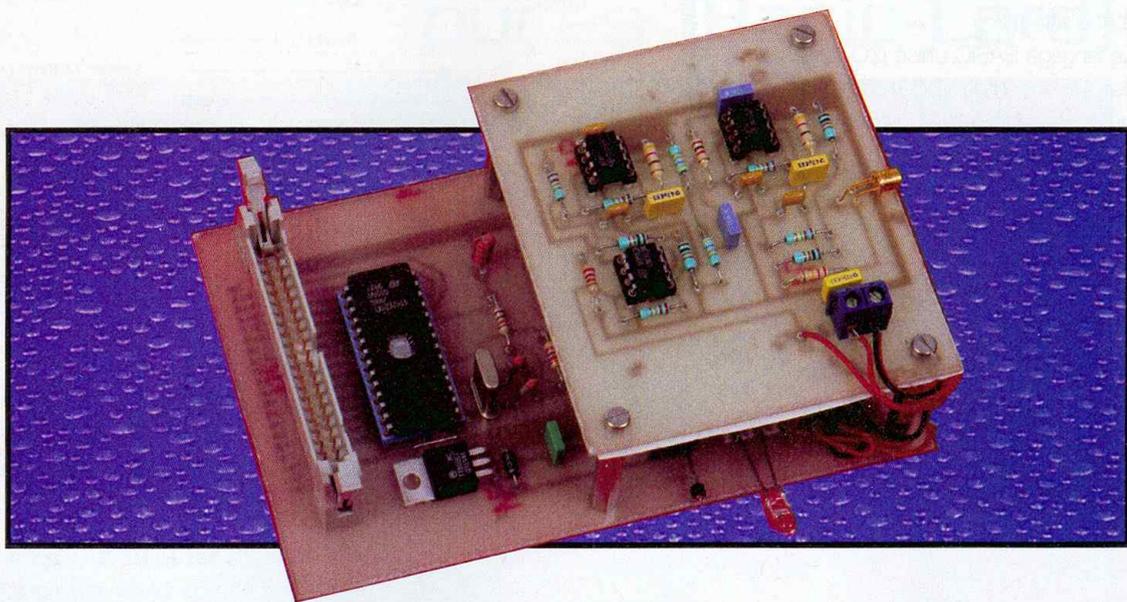
P. OGUIC



Signes mathématiques et ponctuation

# Télécommande infrarouge 16 canaux

pour PC



**Le montage dont nous vous proposons la réalisation dans ces pages est une télécommande infrarouge destinée à être utilisée à partir d'un ordinateur. Avec un tel montage vous pourrez commander, depuis le clavier de votre ordinateur, jusqu'à 16 appareils électriques différents en toute sécurité et avec une grande fiabilité.**

## Caractéristiques

La télécommande infrarouge décrite dans ces pages a été conçue pour pouvoir commander avec un maximum de souplesse et une fiabilité maximale 16 appareils électriques différents et de façon indépendante. L'émetteur de la télécommande se connecte sur un des ports de communication série asynchrone de l'ordinateur, la transmission des commandes s'effectue à une vitesse de 300 bauds (300 bits par secondes). L'émission des bits est effectuée par modulation FSK de l'énergie fournie par deux diodes électroluminescentes émettant dans l'infrarouge. Le récepteur reçoit et décode les commandes reçues grâce à l'utilisation d'un microcontrôleur ST6225 développé par la société SGS-THOMSON. La distance entre l'émetteur et le récepteur peut atteindre 6 m. Chaque commande de mise à l'état 1 ou 0 de chacune des 16 sorties logiques disponibles correspond à une suite de codes binaires selon un protocole spécialement développé et

décrit dans ces pages. Puisqu'il s'agit d'une transmission série asynchrone, les codes sont donc nécessairement envoyés au port de communication série de l'ordinateur auquel est connecté l'émetteur, par l'intermédiaire d'un câble. Le fait que les commandes soient transmises par l'intermédiaire d'un ordinateur permet d'envisager d'automatiser la commande électrique de différents appareils, ce qui ouvre donc la voie vers la domotique.

On peut envisager, par exemple, de pouvoir réaliser un programme permettant de commander à heure fixe l'arrosage de plantes ou la mise en route de la cafetière, et cela sans liaisons électriques puisque la communication entre l'émetteur et le récepteur est réalisée de façon optique. Les caractéristiques principales du montage sont les suivantes :

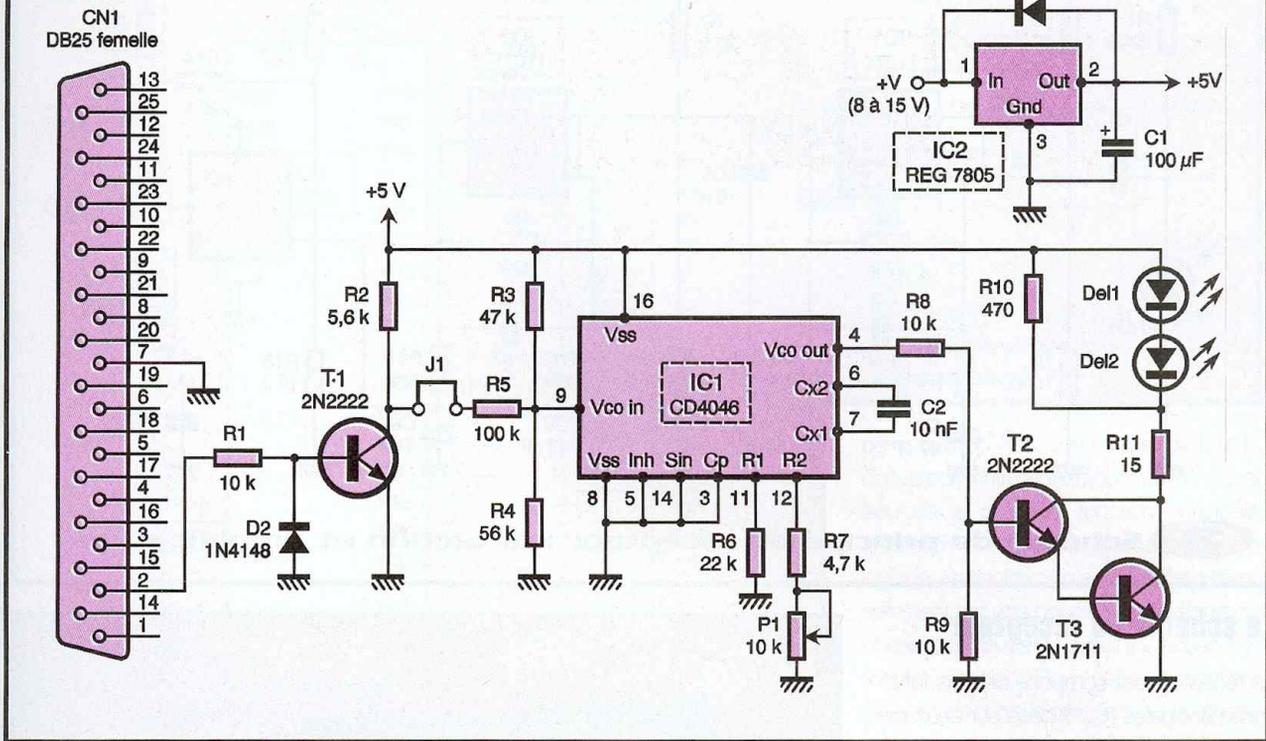
- Vitesse de transmission : 300 bauds (8 bits sans parité, 1 stop),
- Transmission des commandes par modulation FSK assurant une grande fiabilité,
- Portée maximale : 6m,

- 16 sorties logiques indépendantes,
- Témoin de communication entre l'émetteur et le récepteur (facilitant l'alignement),
- Protocole de communication assurant une fiabilité de la transmission des commandes,
- Coût relativement modique lié à l'utilisation de composants courants.

## Le schéma de l'émetteur

Le schéma de l'émetteur est présenté en **figure 1**. Celui-ci fait apparaître un composant bien connu des lecteurs d'électronique pratique puisqu'il s'agit du circuit CD4046, référencé IC<sub>1</sub>. Ce composant intègre tous les éléments d'une boucle à verrouillage de phase (PLL). En effet, le principe adopté pour transmettre l'état d'un bit est celui de la modulation FSK qui n'est autre que la modulation en tout au rien de la fréquence de la porteuse. C'est à dire que la fréquence du courant traversant les diodes émettrices va être modulée ainsi : pour un état logique 0 à transmettre va correspondre une fréquence FL du

## 1 Schéma de principe de l'émetteur

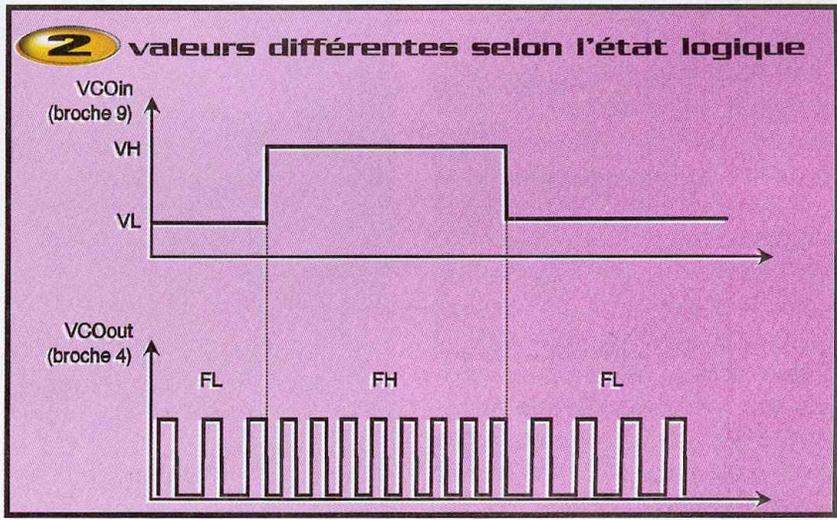


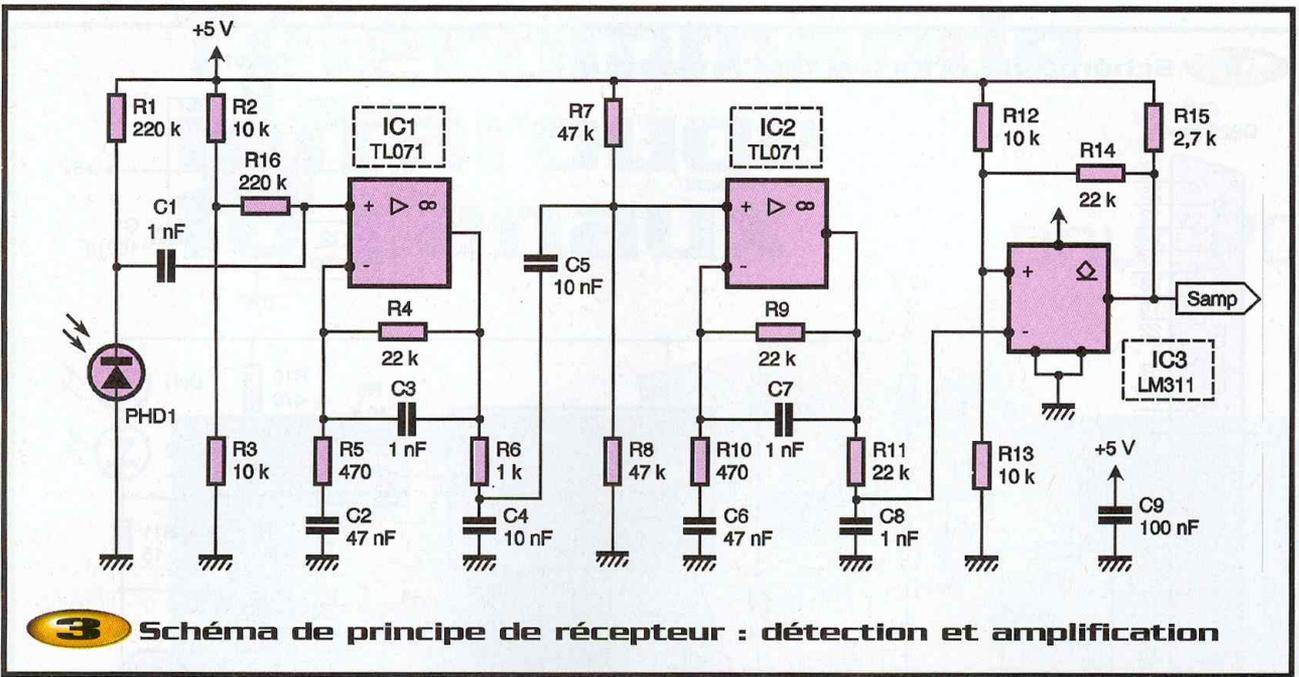
courant et pour un état logique 1 à transmettre va correspondre une fréquence FH différente de la fréquence FL. Au niveau du récepteur, pour pouvoir récupérer l'information logique 0 ou 1 transmise, il faut nécessairement faire la conversion inverse, c'est à dire que pour une fréquence FL ou FH reçue le récepteur doit interpréter cela comme étant un état logique 0 ou 1. En ce qui concerne notre récepteur, cette fonction est réalisée à l'aide d'une boucle à verrouillage de phase de type CD4046. C'est pour rester compatible avec le récepteur que l'on a choisi le même composant CD4046 pour réaliser l'émetteur, à la différence que seul le VCO est utilisé.

Techniquement, la modulation FSK est obtenue par modulation de la tension d'entrée du VCO de IC<sub>1</sub> par l'intermédiaire du transistor T<sub>1</sub> et des résistances associées. La tension présente sur la broche 2 du connecteur CN<sub>1</sub> (relié au port de communication série de l'ordinateur) correspond à un état logique du bit à transmettre et vaut -12V ou +12V (valeurs pouvant différer de quelques volts). Selon sa valeur, cette tension rend le transistor bloqué (pour une valeur de -12V) ou saturé (pour une valeur de +12V) ce qui a pour conséquence de faire varier, selon deux valeurs différentes, la tension aux bornes de la résistance R<sub>4</sub> et

donc le potentiel de la broche 9 de IC<sub>1</sub> (entrée du VCO). De ce fait, la fréquence issue de la broche 4 de IC<sub>1</sub> (sortie du VCO) peut avoir deux valeurs différentes selon que l'on transmette un état logique 0 ou 1, comme cela est illustré par la **figure 2**. Cette tension périodique modulée commande un étage de puissance constitué des transistors T<sub>2</sub> et T<sub>3</sub> destiné à fournir un courant suffisamment élevé dans les diodes émettrices infrarouge DEL<sub>1</sub> et DEL<sub>2</sub>. Ces deux transistors constituent un étage Darlington, R<sub>11</sub> limite le courant dans les diodes DEL<sub>1</sub> et DEL<sub>2</sub> alors que R<sub>10</sub> améliore

le temps de réponse de l'étage de puissance et des diodes par élimination des charges accumulées. La diode D<sub>2</sub> permet de protéger la jonction base-emetteur du transistor T<sub>1</sub>, lorsque le potentiel de la broche 2 du connecteur CN<sub>1</sub> vaut -12V. Le potentiomètre P<sub>1</sub> permet de faire varier l'intervalle de fréquences afin que l'émetteur et le récepteur soient "accordés" l'un avec l'autre. Le strap J<sub>1</sub> permet d'effectuer une mise au point de la télécommande, comme cela est décrit dans les pages suivantes, et doit resté connecté pour un mode de fonctionnement normal.



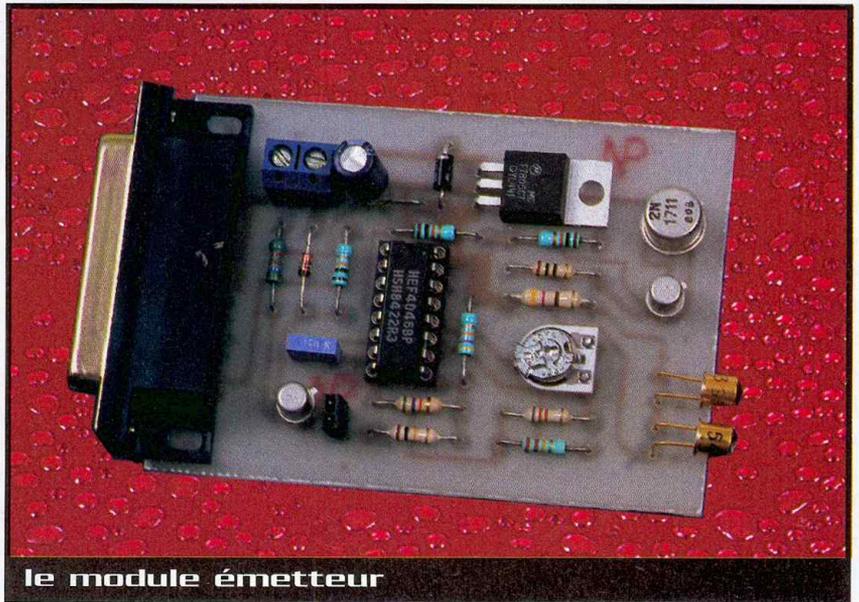


**3** Schéma de principe de récepteur : détection et amplification

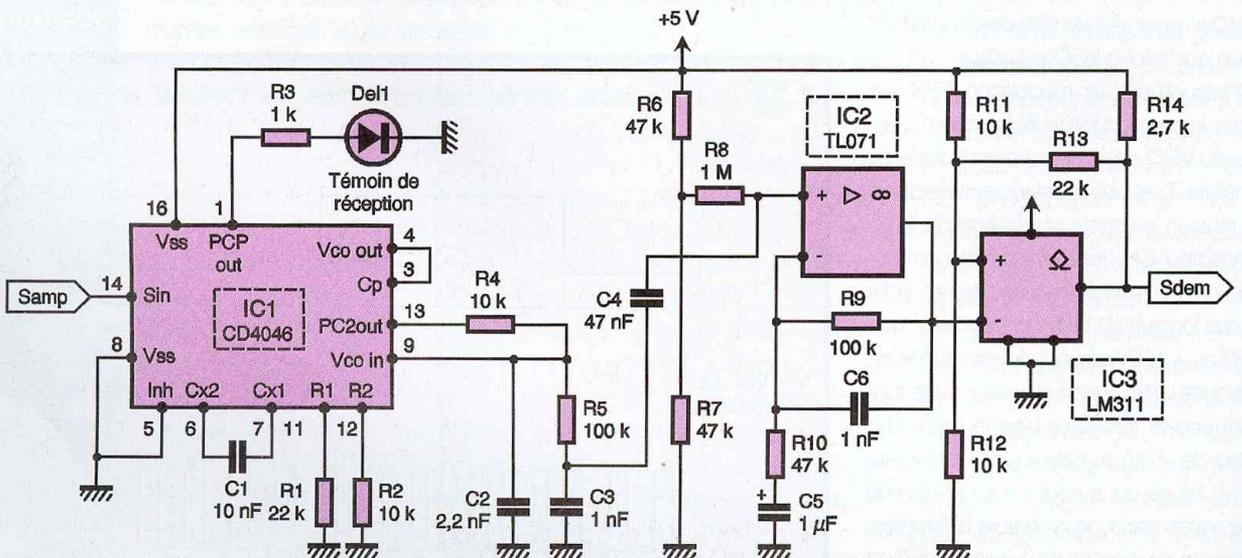
### Le schéma du récepteur

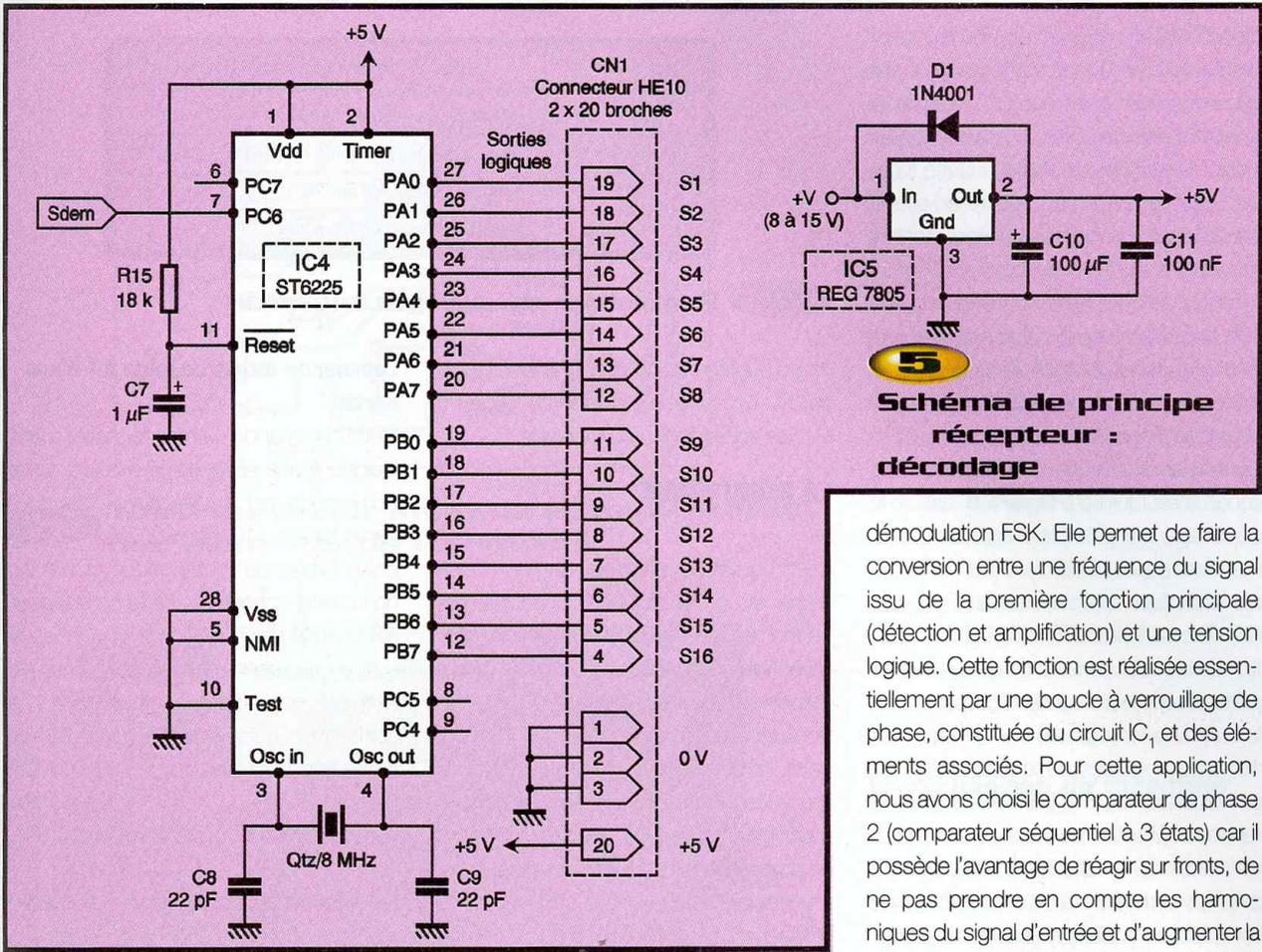
Le récepteur est composé de trois fonctions principales regroupées sur deux circuits imprimés. La première fonction principale, dont le schéma est fourni à la **figure 3**, assure la détection, l'amplification et la mise en forme du signal reçu. La photodiode PHD<sub>1</sub> permet de capter le flux lumineux reçu et de le convertir en une tension électrique modulée en fréquence. Cette tension est amplifiée par une succession d'amplificateurs sélectifs afin que la tension

**4**  
**Schéma de principe de récepteur : démodulation FSK**



le module émetteur



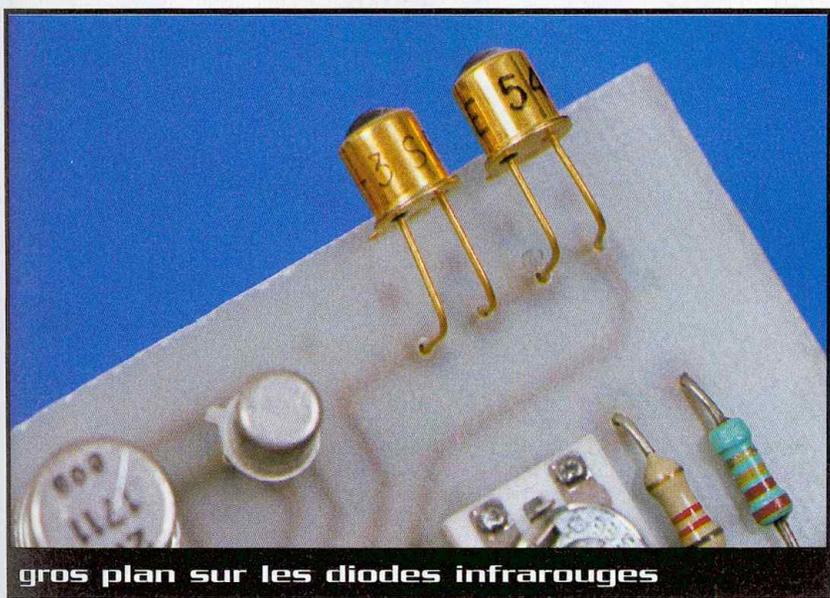


**5**  
Schéma de principe récepteur : décodage

produite soit suffisamment élevée et exempte de parasites. Ainsi, on obtient aux bornes du condensateur  $C_3$  une tension pratiquement sinusoïdale dont l'amplitude est voisine de 2V pour une portée de 5 m. Cette tension est mise en forme rectangulaire par l'intermédiaire d'un comparateur à seuils (comparateur à hystérésis) constitué

par le circuit  $IC_3$  et les résistances  $R_{12}$  à  $R_{15}$ . La tension issue de la sortie de ce comparateur est donc de type logique (0V/+5V) et conserve les mêmes propriétés de modulation de fréquence que le signal capté par la photodiode PHD<sub>1</sub>. La deuxième fonction principale, dont le schéma est fourni à la **figure 4**, assure la

démodulation FSK. Elle permet de faire la conversion entre une fréquence du signal issu de la première fonction principale (détection et amplification) et une tension logique. Cette fonction est réalisée essentiellement par une boucle à verrouillage de phase, constituée du circuit  $IC_1$  et des éléments associés. Pour cette application, nous avons choisi le comparateur de phase 2 (comparateur séquentiel à 3 états) car il possède l'avantage de réagir sur fronts, de ne pas prendre en compte les harmoniques du signal d'entrée et d'augmenter la plage de verrouillage. Son inconvénient majeur étant sa sensibilité aux bruits, c'est pour cela que l'accent a été mis sur le filtrage des parasites lors de l'amplification du signal reçu. Le filtre passe-bas de la boucle est réalisé par les composants  $C_2$ ,  $C_3$ ,  $R_4$  et  $R_5$ . La tension démodulée, présente aux bornes du condensateur  $C_3$ , est amplifiée par l'intermédiaire du circuit  $IC_2$  (TL71) et des composants associés. Il s'agit là encore d'un étage de type passe-bande destiné à amplifier le signal démodulé tout en supprimant les éventuels parasites non significatifs d'un état logique. La tension issue de cet étage passe-bande est transformée en une tension logique (0V/+5V) par l'intermédiaire d'un comparateur à seuils (constitué de  $IC_3$  et  $R_{12}$  à  $R_{14}$ ) identique à celui utilisé dans la première fonction principale décrite dans les lignes précédentes. Ainsi, la tension logique issue de cette deuxième fonction principale est à l'image de l'état logique du bit transmis. La diode électroluminescente DEL<sub>1</sub> permet de renseigner l'utilisateur de la télécommande que le récepteur reçoit correctement le signal modulé fourni par l'émetteur. Cette diode DEL<sub>1</sub> est connectée à la broche 1 (PCP) du



**gros plan sur les diodes infrarouges**

circuit CD4046 délivrant un état logique 1 lorsque la boucle à verrouillage de phase est verrouillée, autrement dit lorsque le récepteur détecte et réagit normalement au signal transmis par l'émetteur. Cette fonction vous sera utile pour mettre en œuvre votre télécommande lors de l'alignement du récepteur avec l'émetteur.

A cette étape de la réception, la succession des bits transmis par le port de communication série de l'ordinateur est reconstituée en sortie de la deuxième fonction principale (figure 3).

La troisième fonction principale, dont le schéma est fourni à la **figure 5**, assure le décodage et l'interprétation des commandes reçues. Cette fonction est assurée essentiellement par le circuit IC<sub>4</sub> qui est un microcontrôleur ST6225. Outre la gestion de la liaison série, le programme interprète chaque commande reçue et affecte la ou les sorties logiques concernées. Les 16 sorties logiques de la télécommande correspondent aux 16 bits des ports A et B répartis de la sorte : les 8 bits du port A pour les sorties S1 à S8 et les 8 bits du port B pour les sorties S9 à S16. Ces broches sont définies en tant que sorties logiques symétriques, c'est à dire qu'un état logique 0 correspond à une tension voisine de 0V et un état logique 1 est équivalent à une tension voisine de +5V. Les 16 sorties sont disponibles par l'intermédiaire du connecteur CN<sub>1</sub> de type HE10 2x20 broches.

La configuration des entrées/sorties du ST6225 est la suivante :

Port A : Sorties logiques symétriques,  
Port B : Sorties logiques symétriques,  
Port C : PC4 à PC7 : En entrée logique avec résistances de rappel sans interruption.

Seul la broche PC6 est utilisée comme entrée série de communication.

Comme cela a été précisé précédemment, les trois fonctions principales sont réparties sur deux circuits imprimés :

- Un circuit pour la première fonction principale (détection et amplification),
- Un circuit pour les deux autres fonctions principales (démodulation, décodage).

L'alimentation est réalisée par un régulateur +5V de type 7805 alimentant les fonctions logiques et analogiques, la diode D<sub>1</sub> permet de protéger l'étage ballast du régulateur. Le quartz, associé aux condensateurs C<sub>8</sub> et C<sub>9</sub>, constitue la pièce maîtresse de

Code transmis	Action associée à l'émission du code
\$Ax	Activation de la sortie numéro x (x = 0 à 15)
\$Dx	Désactivation de la sortie numéro x (x = 0 à 15)
\$FA	Activation de toutes sorties
\$FD	Désactivation de toutes sorties

## 6 Protocole de communication

l'horloge intégrée au microcontrôleur. Le circuit R<sub>15</sub>/C<sub>7</sub> assure l'impulsion de RESET à la mise sous tension du montage.

### Le programme

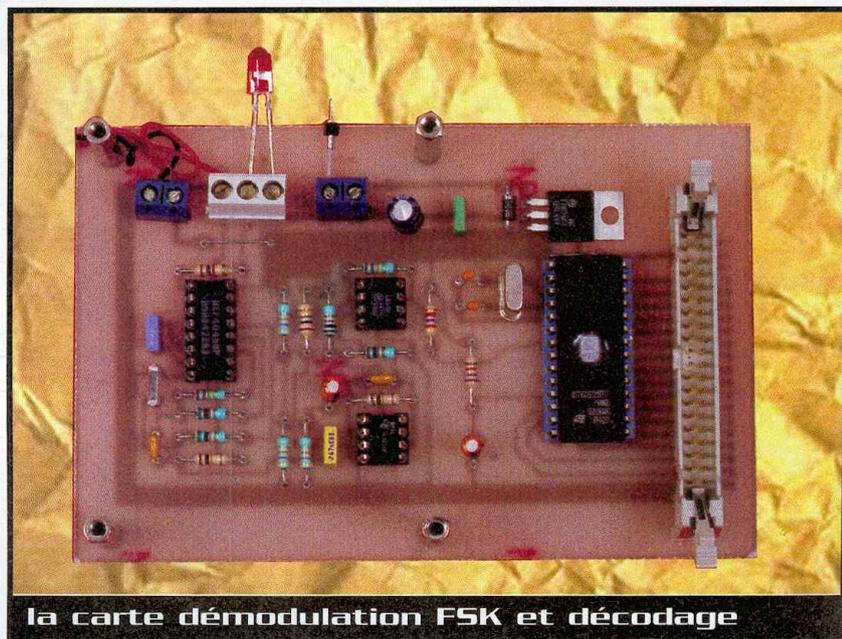
Le programme principal est composé d'une structure qui boucle en permanence en attendant la réception d'un caractère puis en interprétant la commande reçue. Celle-ci peut ne pas exister (cas d'une mauvaise liaison optique entre l'émetteur et le récepteur) auquel cas le caractère est purement et simplement ignoré et le programme revient dans sa phase d'attente. Le protocole de communication est présenté sous la forme d'un tableau à la **figure 6**. Il a été défini de telle sorte que chaque commande puisse être codée sur un seul octet. Ces commandes sont regroupées en 2 familles principales : les commandes locales qui n'agissent que sur une seule sortie à la fois, et les commandes globales qui agissent sur l'ensemble des sorties simultanément.

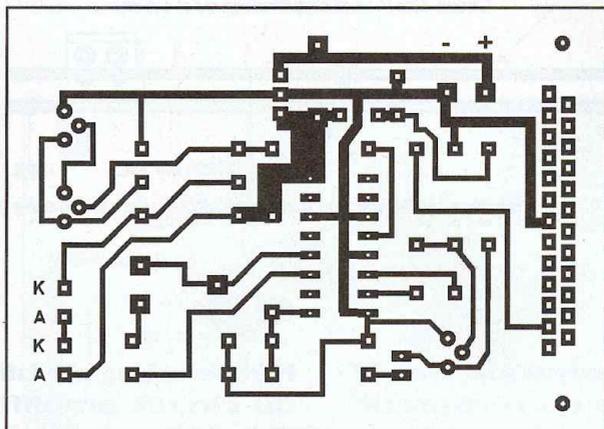
### Commande locale de mise à 1 d'une sortie

Cette commande permet de mettre à l'état logique 1 une seule sortie à la fois. Cette commande est codée sur 8 bits sous forme binaire : 1010xxxx<sup>(2)</sup>. Les 4 bits de poids faibles de l'octet codent le numéro de la sortie concernée, cette combinaison est donnée par la relation xxx = (n° sortie - 1) en représentation binaire. Ainsi, par exemple, pour mettre à l'état logique 1 la sortie n°5, l'octet correspondant à cette commande est le suivant : 10100100<sup>(2)</sup> puisque 5 - 1 = 4 = 0100<sup>(2)</sup>. Sous forme hexadécimale le même code vaut \$A4.

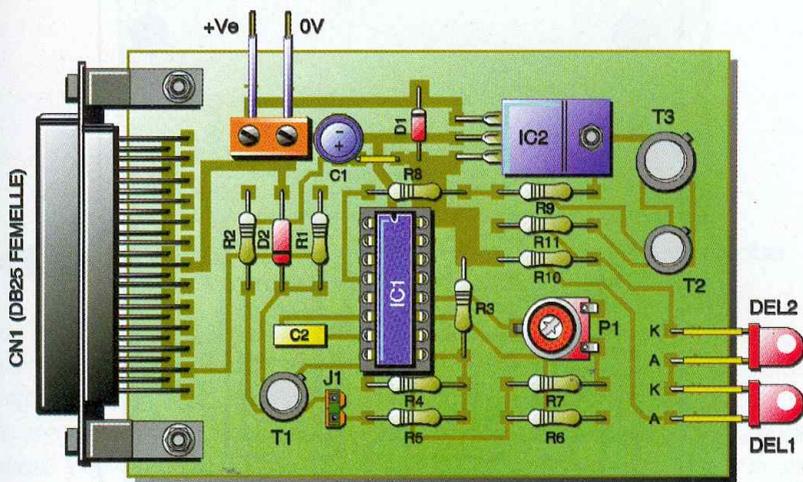
### Commande locale de mise à 0 d'une sortie

Cette commande permet de mettre à l'état logique 0 une seule sortie à la fois. Cette commande est codée sur 8 bits sous forme binaire : 1101xxxx<sup>(2)</sup>. Là encore, les 4 bits de poids faibles de l'octet codent le numéro de la sortie concernée, cette combinaison est donnée par la relation xxx = (n° sortie - 1) en représentation binaire. Ainsi, par





**718** Tracé du circuit imprimé et implantation des éléments de l'émetteur



exemple, pour mettre à l'état logique 0 la sortie n°12, l'octet correspondant à cette commande est le suivant : 11011011(²) puisque 12 - 1 = 11 = 1011(²). Sous forme hexadécimale le même code vaut \$DB.

**Commande globale de mise à 1 de toutes les sorties**

Cette commande permet de mettre à l'état logique 1 toutes les sorties en même temps. Cette commande est codée sur 8 bits sous forme binaire : 11111010(²). Sous forme hexadécimale le même code vaut \$FA.

**Commande globale de mise à 0 de toutes les sorties**

Cette commande permet de mettre à l'état logique 0 toutes les sorties en même temps. Cette commande est codée sur 8 bits sous forme binaire : 11111101(²). Sous forme hexadécimale le même code vaut \$FD.

Vous pourrez vous procurer le programme rédigé en assembleur "Tele16s.asm" correspondant à l'application ainsi que le fichier "6225\_Reg.asm" par les voies habituelles (Internet...). C'est deux fichiers sont nécessaires pour générer le code exécutable (le fichier 6225\_reg.asm contient les définitions des adresses des différents registres du microcontrôleur). Une fois ce programme assemblé, vous disposerez d'un fichier binaire que vous devrez programmer dans votre microcontrôleur ST6225 à l'aide de l'outil de développement qui lui est dédié. Si vous ne disposez pas de l'assembleur vous pourrez vous procurer, par les mêmes sources, le fichier assemblé au format Intel-Hex "Tele16s.Hex".

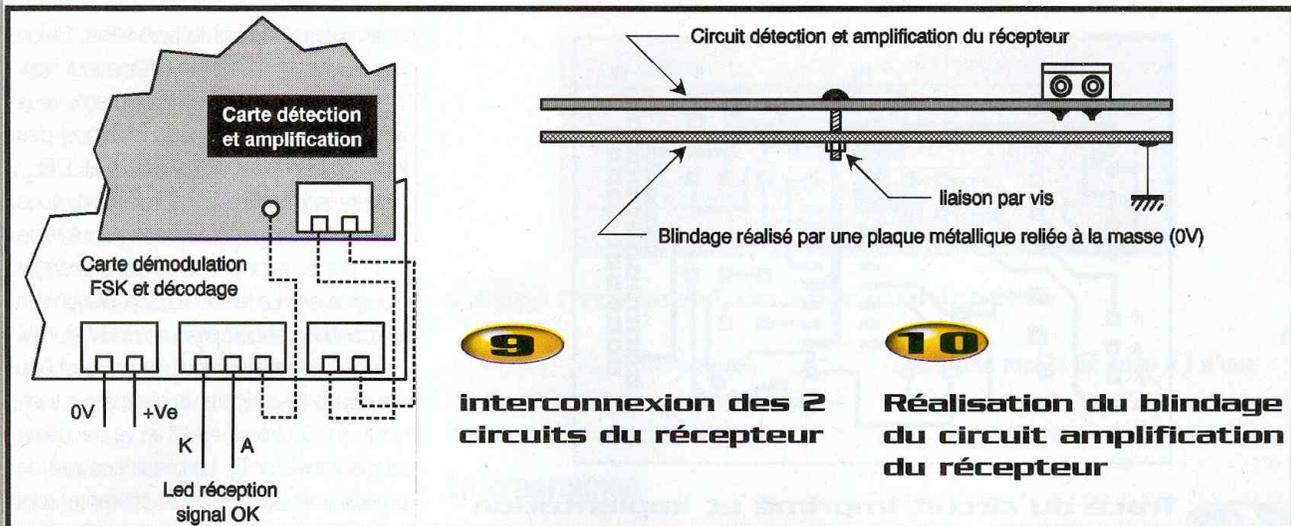
**La réalisation de l'émetteur**

La réalisation de l'émetteur ne pose aucune difficulté particulière compte tenu du peu de composants utilisés. Comme il se doit, vous veillerez à implanter tous les compo-

sants polarisés dans le bon sens. Selon que vous souhaitez placer l'émetteur horizontalement ou bien verticalement, vous devrez couder ou non les broches des deux diodes émettrices DEL<sub>1</sub> et DEL<sub>2</sub>. Compte tenu du courant consommé, nous déconseillons d'alimenter l'émetteur à l'aide d'une pile sous peine de devoir la changer fréquemment. La tension d'alimentation doit être continue et comprise entre 8V et 12V. Pour cette dernière valeur, il sera peut être nécessaire d'adjoindre un dissipateur thermique au régulateur +5V. Il en va de même pour le transistor T<sub>3</sub>. Le circuit imprimé de l'émetteur et l'implantation afférente sont fournis aux figures 7 et 8. Pour réaliser le câble de liaison entre la carte et votre ordinateur, nous vous conseillons d'utiliser du câble en nappe et des connecteurs SUBD à sertir. Il vous suffit de respecter la connexion borne à borne des deux connecteurs et le tour est joué.

**La réalisation du récepteur**

La réalisation du récepteur est un peu délicate compte tenu de la relative complexité des deux circuits imprimés. Il faudra s'assurer de la continuité des pistes (avec un testeur de continuité) avant de souder les composants, ce contrôle est particulièrement nécessaire pour le circuit intégrant le microcontrôleur ST6225. Vous veillerez à implanter tous les composants polarisés dans le bon sens. De plus, apportez un soin particulier au soudage du quartz et des deux condensateurs de 22 pF qui sont les pièces maîtresses de l'horloge intégrée au ST6225. Il est préférable que le circuit de détection et d'amplification (comportant la diode de réception infrarouge) soit fixé le plus près possible du circuit de démodulation et de décodage afin de minimiser les perturbations liées au câblage des deux cartes. Il a été prévu des repères, matérialisés par des pastilles de grand diamètre, afin de pouvoir fixer le circuit d'amplification au-dessus du circuit de démodulation et de décodage, comme cela est illustré par la figure 9. De plus, pour éliminer toutes perturbations liées à la grande impédance d'entrée et à l'amplification assez élevée, le circuit de détection et d'amplification devra être blindé. Ce blindage peut être réalisé facilement en fixant à l'aide de vis une plaque métallique



9

interconnexion des 2 circuits du récepteur

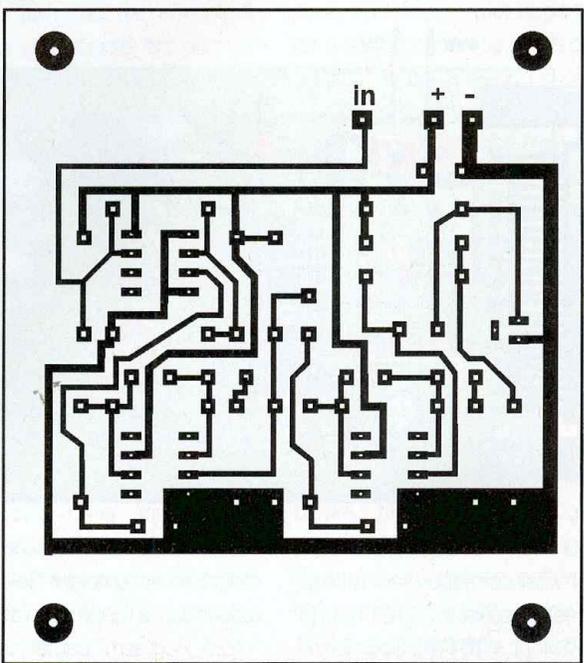
10

Réalisation du blindage du circuit amplification du récepteur

le plus près des pistes possible, comme cela est illustré par la **figure 10**. Ce blindage doit impérativement être relié à la masse du montage (0V) si vous voulez qu'il soit efficace. La tension de 5V étant délivrée par un régulateur intégré, l'alimentation extérieure peut varier de 8V à environ 15V. Le circuit imprimé de la carte de détection et d'amplification et l'implantation afférente sont fournis aux **figures 11** et **12**. Le circuit imprimé de la carte de démodulation et décodage et l'implantation afférente sont fournis aux **figures 13** et **14**. Le témoin de réception, constitué par la diode DEL<sub>1</sub>, doit être visible par l'utilisateur de la télécommande.

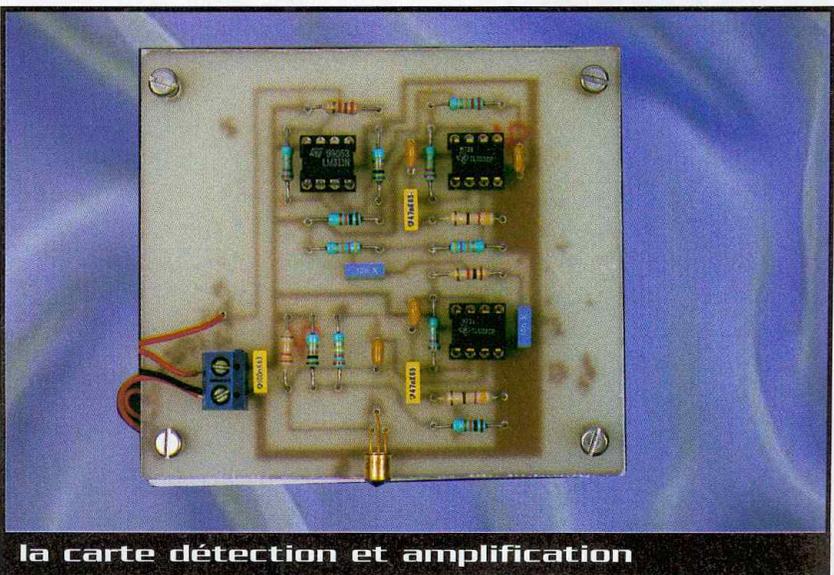
**Mise en œuvre**

Après avoir réaliser les différents circuits et, à la condition que vous ayez pris soin de suivre attentivement les conseils qui vous sont fournis lors de leur réalisation, l'ensemble doit fonctionner dès la mise sous tension. Pour les essais, vous pouvez alimenter l'émetteur et le récepteur par une pile de 9V. Avant toute utilisation avec votre ordinateur, il vous faut "accorder" l'émetteur et le récepteur. Pour cela, placez l'émetteur en face du récepteur, à environ 50 cm et ôtez le strap J<sub>1</sub> du circuit émetteur. En agissant sur le potentiomètre P<sub>1</sub> de l'émetteur et à l'aide d'un multimètre numérique, réglez la tension aux bornes de C<sub>3</sub> à environ 3,2V (cela doit correspondre au potentiomètre à mi-course environ). Cette valeur peut varier entre 3V et 3,5V sans trop modifier la qualité de la réception. Vous devez constater l'allumage sans scintillements



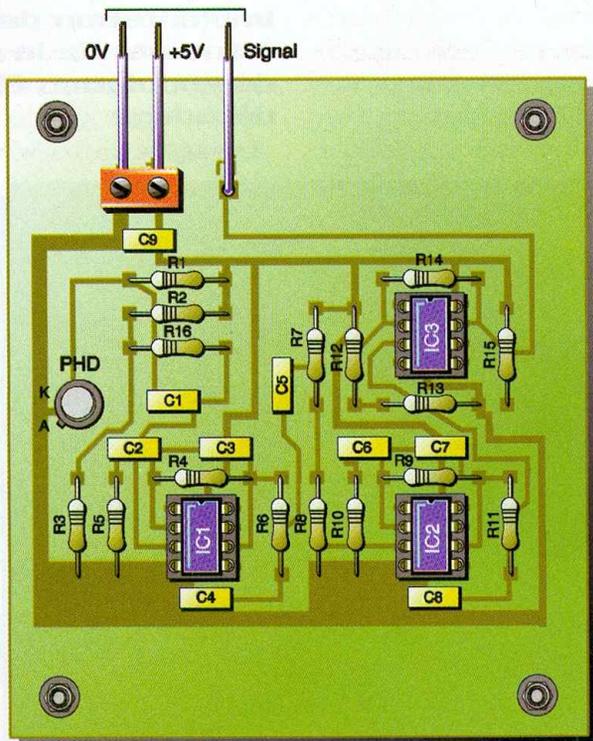
11

Tracé du circuit imprimé de la carte de détection et amplification



la carte de détection et amplification

Connexion avec la carte démodulation

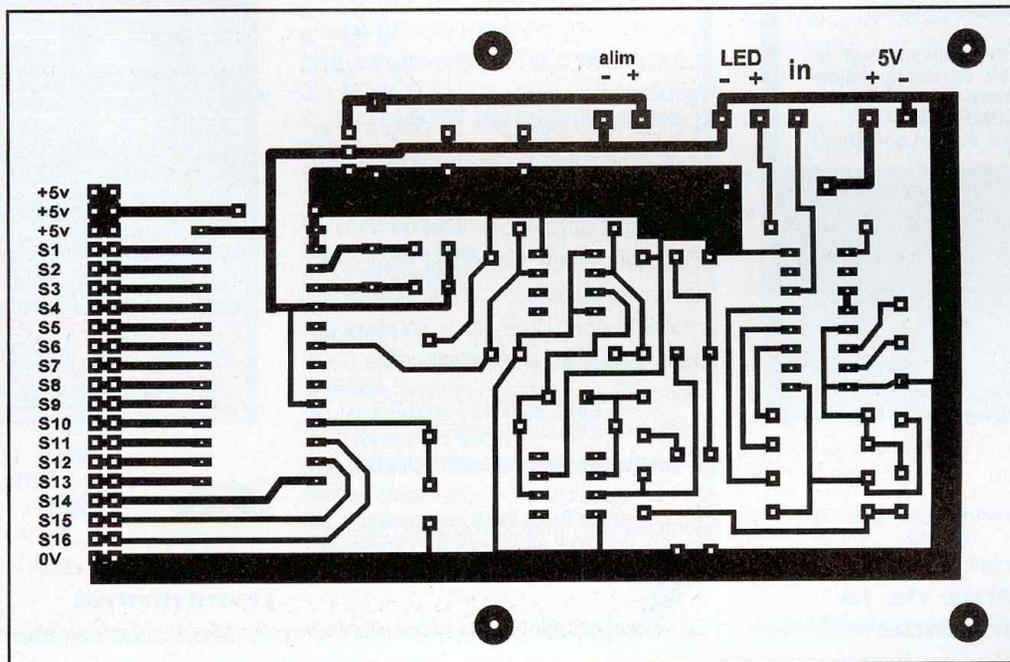


## 12 Implantation des éléments de la carte détection et amplification

de la diode électroluminescente DEL<sub>1</sub> (témoin de bonne réception). Une fois ce réglage effectué, remplacez le strap J<sub>1</sub> sur le circuit de l'émetteur. A présent, vous devez pouvoir constater que lorsque vous ne dirigez plus l'émetteur en direction du récepteur, la diode DEL<sub>1</sub> doit

s'éteindre franchement. De même, celle-ci doit s'allumer franchement lorsque vous pointez vers le récepteur. Si la diode scintille, faites varier finement la position du potentiomètre P<sub>1</sub> de l'émetteur (essayez dans un sens puis dans l'autre) de telle sorte à éliminer ce scintillement.

## 13 Tracé du circuit imprimé de la carte démodulation FSK et décodage



Voilà, votre télécommande est prête à fonctionner.

Pour mettre en œuvre votre télécommande dans vos applications vous avez plusieurs possibilités. La première consiste à rédiger un programme répondant à vos besoins et capable de transmettre vos commandes au port de communication série choisi. Un tel travail n'est pas excessivement difficile à réaliser, le plus dur étant de configurer les paramètres de communication.

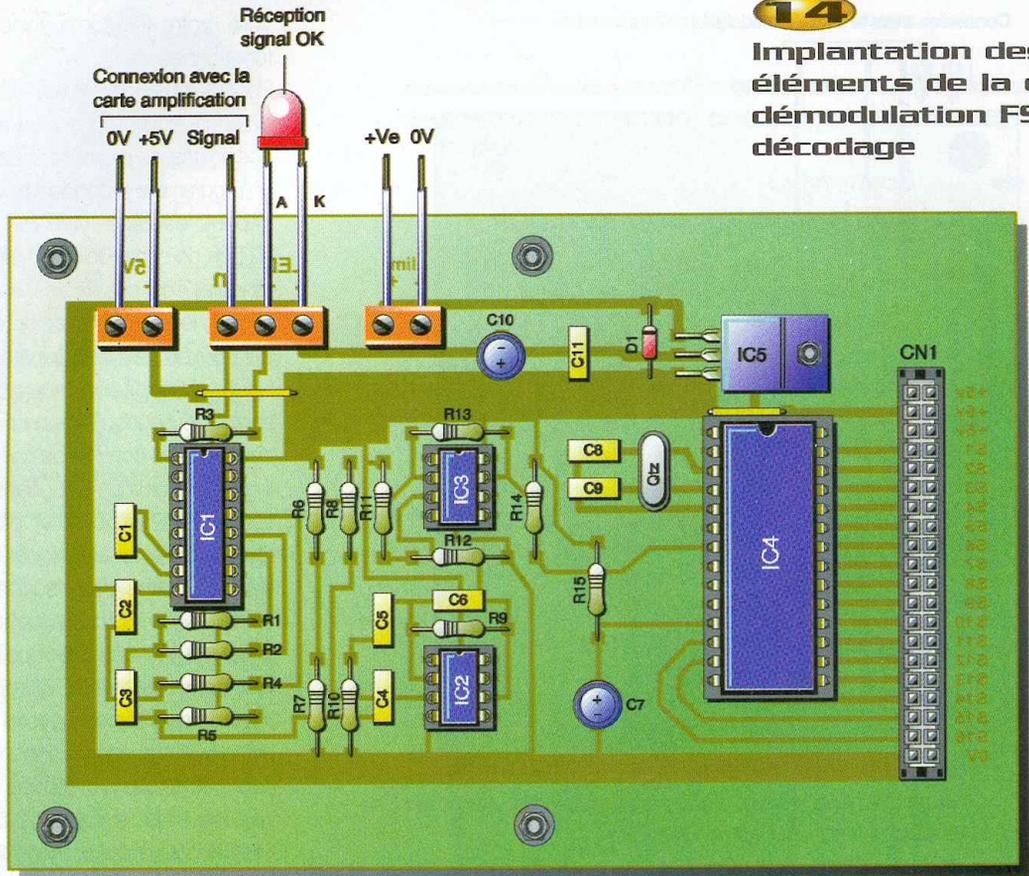
Nous vous rappelons que ces paramètres sont : 300 bauds, 8 bits, 1 bit de stop, sans parité et sans contrôle de flux (ni matériel, ni logiciel). Pour vous aider dans votre travail, vous pourrez exploiter les programmes et renseignements techniques déjà publiés dans nos pages. La deuxième possibilité est l'utilisation d'un programme spécialement rédigé pour cette application, fonctionnant sous WINDOWS 95, et faisant apparaître sur l'écran de votre ordinateur un boîtier de télécommande directement exploitable. Ce programme a été développé en PASCAL OBJET (à l'aide de DEL-PHI3) fonctionnant sous WINDOWS 95 et dont la présentation est proposée dans les lignes suivantes.

## Utilisation du programme sous WINDOWS 95

Afin que vous puissiez utiliser votre télécommande immédiatement après sa réalisation, nous vous proposons un programme fonctionnant sous WINDOWS 95

14

Implantation des éléments de la carte démodulation FSK et décodage



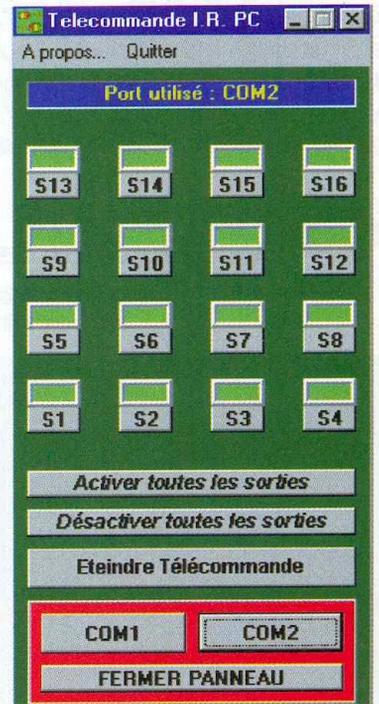
15

Configuration du programme de la télécommande

que vous pourrez vous procurer par les voies habituelles (Internet, ...). Il est préférable de copier le programme "Tele\_ir.exe" dans un répertoire spécifique à cette application, par exemple C:\TELECOM. Vous pouvez bien entendu choisir tout autre nom de répertoire. Cette raison s'explique simplement par le fait que lors de la première utilisation le programme enregistre un fichier de configuration du port série utilisé, ceci évite de devoir systématiquement spécifier quel port série est utilisé par l'application.

A la première exécution du programme, vous devez voir apparaître une fenêtre telle que celle présentée à la figure 15. Il vous faut alors choisir le port de communication série sur lequel vous allez connecter l'interface. Veillez à ne pas choisir le port utilisé par la souris sous peine d'avoir de grosses difficultés pour reprendre le contrôle de WINDOWS à l'aide du clavier. Si vous avez commis une telle erreur, ce n'est pas dramatique puisqu'il est possible de modifier le fichier de configuration "PortCom.cfg" à l'aide du bloc-notes se trouvant dans les acces-

soires. Ce fichier contient deux lignes dont la



16

Utilisation du programme

deuxième est soit COM1 soit COM2, il est donc facile de changer le dernier caractère. Une fois le port de communication choisi, à chaque utilisation du programme vous devez voir apparaître une fenêtre telle que celle présentée à la **figure 16**. Le programme gère automatiquement la configuration du port de communication série choisi ainsi que la transmission des commandes chaque fois que vous cliquez sur un bouton correspondant à une sortie. Son utilisation demeure fort simple et intuitive.

## Conclusion

Le montage proposé a été utilisé et testé avec satisfaction conjointement avec le

programme "tele\_ir.exe" fonctionnant sous WINDOWS 95. L'émetteur et le récepteur peuvent être fixés sur deux murs opposés. Avec un peu de programmation il est possible, par exemple, de développer un programme permettant de commander jusqu'à 16 appareils différents à différentes heures de la journée. Il est même possible, par modification du programme contenu dans le microcontrôleur ST6225 et du protocole, d'étendre le nombre de sorties à un nombre bien supérieur à 16 (par exemple 256).

E. QUAGLIOZZI

## Nomenclature

### Émetteur

$R_1, R_8, R_9$  : 10 k $\Omega$  - 1/4W  
 $R_2$  : 5,6 k $\Omega$  - 1/4W  
 $R_3$  : 47 k $\Omega$  - 1/4W  
 $R_4$  : 56 k $\Omega$  - 1/4W  
 $R_5$  : 100 k $\Omega$  - 1/4W  
 $R_6$  : 22 k $\Omega$  - 1/4W  
 $R_7$  : 4,7 k $\Omega$  - 1/4W  
 $R_{10}$  : 470  $\Omega$  - 1/4W  
 $R_{11}$  : 15  $\Omega$  - 1/2W  
 $P_1$  : potentiomètre horizontal 10 k $\Omega$   
 $D_1$  : 1N4001 à 1N4007  
 $D_2$  : 1N4148 ou équivalent (diode de commutation)  
 $C_1$  : 100  $\mu$ F/16V chimique, sorties radiales  
 $C_2$  : 10 nF  
 $T_1, T_2$  : 2N2222  
 $T_3$  : 2N1711  
 $IC_1$  : CD4046 (PLL)  
 $IC_2$  : régulateur +5V type 7805  
 $DEL_1, DEL_2$  : diodes infrarouges SE 5470 (HONEYWELL)  
 $CN_1$  : connecteur SUBD 25 broches femelle  
 $J_1$  : strap au pas de 2,54 mm  
 1 bornier 2 fils au pas de 5,08 mm

### Carte détection et amplification du récepteur

$R_1, R_{16}$  : 220 k $\Omega$  - 1/4W  
 $R_2, R_3, R_{12}, R_{13}$  : 10 k $\Omega$  - 1/4W  
 $R_4, R_9, R_{11}, R_{14}$  : 22 k $\Omega$  - 1/4W  
 $R_5, R_{10}$  : 470  $\Omega$  - 1/4W  
 $R_6$  : 1 k $\Omega$  - 1/4W  
 $R_7, R_8$  : 47 k $\Omega$  - 1/4W  
 $R_{15}$  : 2,7 k $\Omega$  - 1/4W  
 $C_1, C_3, C_7, C_8$  : 1 nF  
 $C_2, C_6$  : 47 nF

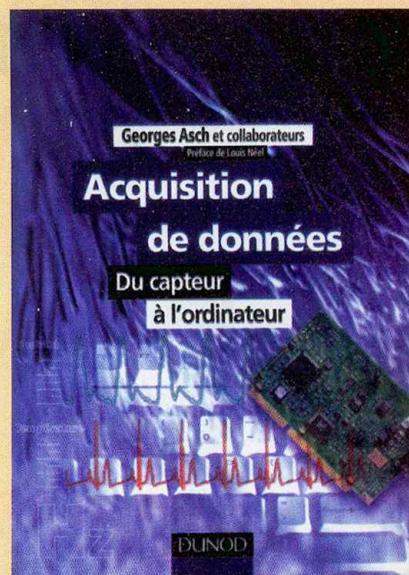
$C_4, C_5$  : 10 nF  
 $C_9$  : 100 nF  
 $IC_1, IC_2$  : TL071  
 $IC_3$  : MAX232  
 $IC_4$  : LM311  
 $PHD_1$  : photodiode SD5421-2 (HONEYWELL)  
 2 borniers 2 fils au pas de 5,08 mm

### Carte démodulation FSK et décodage

$R_1, R_{13}$  : 22 k $\Omega$  - 1/4W  
 $R_2, R_4, R_{11}, R_{12}$  : 10 k $\Omega$  - 1/4W  
 $R_3$  : 1 k $\Omega$  - 1/4W  
 $R_5, R_9$  : 100 k $\Omega$  - 1/4W  
 $R_6, R_7, R_{10}$  : 47 k $\Omega$  - 1/4W  
 $R_8$  : 1 M $\Omega$  - 1/4W  
 $R_{14}$  : 2,7 k $\Omega$  - 1/4W  
 $R_{15}$  : 15 k $\Omega$  - 1/4W  
 $C_1$  : 10 nF  
 $C_2$  : 2,2 nF  
 $C_3, C_6$  : 1 nF  
 $C_4$  : 47 nF  
 $C_5, C_7$  : 1  $\mu$ F/16V électrochimique (tantalé)  
 $C_8, C_9$  : 22 pF céramique  
 $C_{10}$  : 100  $\mu$ F/16V chimique  
 $C_{11}$  : 100 nF/MKT  
 $Qtz$  : Quartz 8 MHz  
 $IC_1$  : CD4046 (PLL)  
 $IC_2$  : TL071  
 $IC_3$  : LM311  
 $IC_4$  : Microcontrôleur ST62E25 ou ST62T25  
 $IC_5$  : régulateur +5V type 7805  
 $D_1$  : 1N4001 à 1N4007  
 $DEL_1$  : diode électroluminescente (couleur au choix)  
 $CN_1$  : connecteur HE10 2x20 contacts  
 2 borniers 2 fils au pas de 5,08 mm  
 1 bornier 3 fils au pas de 5,08 mm

## Acquisition de données

### Du capteur à l'ordinateur



Qu'il s'agisse des laboratoires de recherche scientifique, des laboratoires à vocation industrielle, des fabricants de bancs de tests et d'essais ou, encore, des unités de production aux procédés industriels si diversifiés, tous ont ce point commun d'être confrontés au quotidien à la nécessité d'acquieser un nombre toujours croissant de grandeurs physiques. Et toujours, l'acquisition de ces données génère de nombreuses interrogations face aux difficultés inhérentes à «l'art» de l'instrumentation. C'est pour répondre à ces interrogations et pour aider tous ceux qui, de près ou de loin, sont concernés par la chaîne de mesures à faire des choix «raisonnables» que «Acquisition de données», complémentaire du maintenant classique «Les capteurs en instrumentation industrielle», a été pensé et conçu. A la fois pédagogique et en adéquation avec les besoins des professionnels de la mesure, cet ouvrage s'adresse à tous ceux qui, au laboratoire ou dans l'industrie, conçoivent, choisissent ou utilisent des chaînes d'acquisition, mais également aux étudiants scientifiques des IUT, des universités et des écoles d'ingénieurs.

G. ASCH et collaborateurs  
 DUNOD

488 pages - 330 F

# Communication série asynchrone avec un microcontrôleur ST6230



Dans cet article nous allons expliquer comment fonctionne le port de communication série asynchrone (U.A.R.T.) du microcontrôleur ST6230 de chez ST-Microelectronics et donner ensuite un circuit accompagné d'un programme pour gérer ce protocole.

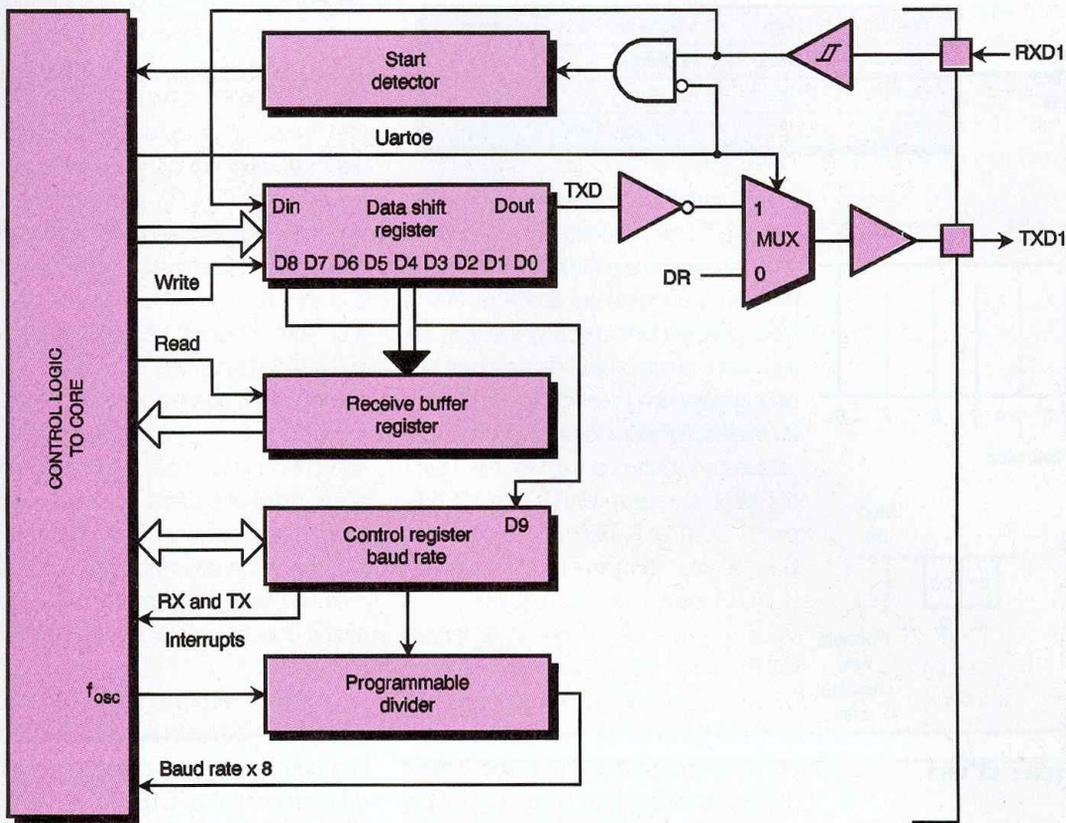
## Description de l'émetteur/récepteur série asynchrone universel (U.A.R.T.) du ST6230

L'émetteur/récepteur asynchrone universel (U.A.R.T. ou «Universal Asynchronous Receiver/Transmitter») est un des périphériques intégrés du microcontrôleur ST6230 dont le schéma bloc interne est représenté à la **figure 1**.

Il fournit une structure matérielle de base pour les communications séries asynchrones qui, associée à une routine logicielle appropriée, se comporte comme une interface série procurant des communications avec différentes vitesses de transmission (jusqu'à 38400 bauds avec une fréquence de l'oscillateur externe de 8 MHz) et avec des formats de caractères très flexibles. Opérant en mode «Half Duplex» seulement, l'U.A.R.T. utilise 11 bits de caractères comprenant un bit de start, 9 bits de données et 1 bit de stop. La parité est prise en charge par le logiciel seulement pour la transmission et pour tester le bit de parité

en réception (bit 9). Les données transmises sont envoyées directement, tandis que les données reçues sont mises dans un registre tampon. Celui-ci permet d'autres réceptions de caractères tandis que la donnée est en train d'être lue à la sortie du registre tampon. Les données en transmission sont prioritaires sur les données en réception. L'U.A.R.T. est alimenté avec une horloge interne au microprocesseur qui est aussi disponible en mode WAIT. Les lignes de réception RXD et de transmission TXD partagent les mêmes broches externes que deux lignes d'entrée/sortie. Par conséquent, la configuration de l'U.A.R.T. demande de positionner ces deux lignes d'entrée/sortie à travers les registres de port correspondants de la manière suivante : la ligne commune avec RDX/PD4 doit être définie en mode entrée (avec ou sans pull-up), tandis que la ligne commune avec TXD/PD5 doit être définie en mode sortie (push-pull ou drain ouvert). La donnée transmise est inversée et peut, par conséquent, utiliser un simple transistor

pour isoler l'étage. Définie comme une entrée, la ligne TXD peut être lue à tout moment comme une ligne d'entrée durant les opérations en mode U.A.R.T. La broche TXD suit la valeur des registres de port d'entrée/sortie quand le bit UARTOE du registre de contrôle UARTCR est à 0, ce qui signifie qu'aucune transmission série n'est en cours. Par conséquent, un niveau haut permanent doit être écrit dans le registre de donnée du port d'entrée/sortie de manière à remplir une condition de stop correcte sur la ligne TXD lorsqu'aucune transmission n'est active. L'U.A.R.T. contient un diviseur de fréquence interne au microcontrôleur ST6230 pour les différentes vitesses de transmission actuellement utilisées, comme le détaille la **figure 2**. D'autres valeurs de vitesses de transmission peuvent être calculées à partir d'une fréquence d'oscillation choisie divisée par la valeur du diviseur montrée. La transmission est fixée avec un format de 1 bit de start, de 9 bits de données et de 1 bit de stop. Les bits de start et de stop sont automatique-



## 1 Synoptique

ment générés par l'U.A.R.T. Les 9 bits de données sont sous le contrôle de l'utilisateur et sont flexibles quant à leur emploi. Les bits 0 à 7 sont typiquement utilisés comme bits de données tandis que le bit 8 est généralement utilisé comme bit de parité, mais peut aussi être un neuvième bit de donnée ou un second bit de stop. Comme la parité n'est pas générée par l'U.A.R.T., elle doit être calculée par logiciel et insérée dans la position appropriée de la donnée (c'est-à-dire comme bit 7 pour 7 bits de données, avec le bit 9 positionné à

1 donnant deux bits effectifs de stop ou comme un bit 9 indépendant). Les différents formats de caractères disponibles sont résumés à la **figure 3** et la **figure 4** représente le format type d'un caractère. Le bit 9 reste dans l'état programmé pour des transmissions consécutives jusqu'à un changement de la part de l'utilisateur ou jusqu'à ce qu'un caractère soit reçu quand l'état de ce bit est changé en celui du bit 9 entrant. La procédure recommandée est, par conséquent, de positionner la valeur de ce bit avant que la transmission ne débute. Une transmission démarre en écrivant dans le registre de donnée UARTDR (la vitesse

de transmission ainsi que la valeur du bit 9 doivent être positionnées avant cette action dans le registre de contrôle UARTCR). Le signal interne au circuit UARTOE commute la sortie du multiplexeur, comme le montre la **figure 5**, vers la sortie de l'UART et un bit de start est généré (un 0 pendant un temps bit) suivi par les 8 bits de données (avec le bit de poids faible en premier) et par la valeur du bit 9. La sortie est alors mise à 1 pendant un temps bit afin de générer un bit de stop et, ensuite, le signal UARTOE fait commuter la sortie vers la ligne d'entrée/sortie. La fin de la transmission est signalée en mettant le bit TXMT du registre UARTCR à 1 et une interruption est générée si elle est activée. Le drapeau TXMT est initialisé en écrivant un 0 à sa position dans le registre UARTCR, mais il l'est aussi lorsqu'un nouveau caractère est écrit dans le registre de données UARTDR. TXMT peut être positionné à 1 par logiciel pour générer une interruption logicielle, aussi faut-il faire attention lors de la manipulation du registre UARTCR. L'U.A.R.T. scrute continuellement un front descendant sur la broche d'entrée pour voir si une transmission est en cours. Une fois qu'un front a été détecté, il attend un temps bit pour rece-

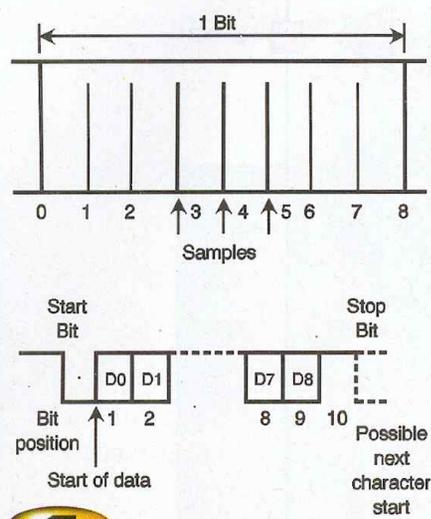
## 2 Diviseur de fréquences

BR2	BR1	BR0	Fint Division	Baud Rate	
				Fint = 8 MHz	Fint = 4 MHz
0	0	0	6,656	1200	600
0	0	1	3,328	2400	1200
0	1	0	1,664	4800	2400
0	1	1	832	9600	4800
1	0	0	416	19200	9600
1	0	1	256	31200	15600
1	1	0	208	38400	19200
1	1	1	Reserved		

**3**

**Différents formats de caractères**

Start Bit	8 Data	1 Software Parity	1 Stop
Start Bit	9 Data	No Parity	1 Stop
Start Bit	8 Data	No Parity	2 Stop
Start Bit	7 Data	1 Software Parity	2 Stop



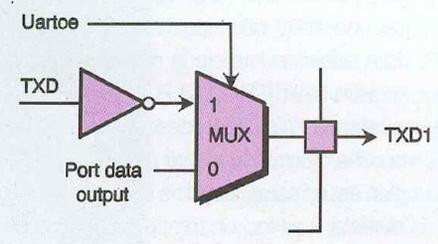
**4**

**Format type d'un caractère**

voir le bit de start, puis il assemble le flot de bits en série qui suit et le transfert dans le registre UARTDR. La donnée du bit 9 est copiée dans DAT9 du registre UARTCR et remplace ainsi la valeur précédente mise pour la transmission. Une fois que les 9 bits ont été reçus, le récepteur attend un temps bit pour le bit de stop et transfert ensuite la donnée reçue dans le tampon de réception permettant ainsi la réception d'un nouveau caractère. Le drapeau d'interruption RXRDY est alors positionné à 1 lorsque la donnée est transférée dans le tampon et, si l'interruption est validée, une interruption sera générée. Si une transmission démarre durant le déroulement d'une réception, la transmission prend la priorité et la réception est stoppée pour laisser libre les ressources

**5**

**Signal interne de commutation**



pour la transmission ; ceci implique qu'un système de handshaking soit implémenté, en instaurant une scrutation permanente de l'U.A.R.T. pour détecter que la réception n'est pas disponible. Le processus de réception, comme celui de transmission, peut générer une interruption vers le cœur du microcontrôleur. Ces interruptions sont validées en positionnant à 1 les bits TXIEN et RXIEN du registre UARTCR et les drapeaux TXMT et RXRDY sont à 1 suivant leur source d'interruption respective. L'U.A.R.T. comporte deux registres :

- Le registre de données s'appelle UARTDR, il se trouve à l'adresse mémoire D6h du microcontrôleur et il peut être lu et écrit. Les bits 0 à 7 de ce registre sont les bits de données de l'U.A.R.T. Une écriture dans ce registre charge la donnée dans le registre à décalage de transmission. De plus, ceci initialise le drapeau d'interruption de la transmission TXMT. Une lecture dans ce registre retourne la donnée qui se trouve dans le tampon de réception. Il est à remarquer qu'aucune instruction de lecture/écriture peut être utilisée avec ce registre puisque la transmission et la réception partageant la même adresse.
- Le second registre est le registre de contrôle UARTCR, il se trouve à l'adresse mémoire D7h du microcontrôleur et il peut être lu et écrit. Le bit 7 RXRDY (prêt à recevoir) est un drapeau qui devient actif aussitôt qu'un octet complet a été reçu et copié dans le tampon de réception. Il peut être remis à zéro en écrivant un 0 dans ce bit. Si le bit d'interruption RXIEN est validé, une interruption logicielle sera générée. Le bit 6 TXMT (transmission vide) est un drapeau qui devient actif aussitôt qu'un octet complet a été envoyé. Il peut être remis à zéro en écrivant un 0 dans ce bit. Il est automatiquement remis à zéro par une action d'écriture d'une valeur de donnée dans le registre de donnée de l'UART. Le bit 5 RXIEN (interruption en réception validée) est mis à 1 lorsqu'une interruption en réception est validée. L'écriture dans RXIEN n'affecte pas l'état du drapeau d'interruption RXRDY. Le bit 4 TXIEN (interruption en transmission

validée) est mis à 1 lorsqu'une interruption en transmission est validée. L'écriture dans TXIEN n'affecte pas l'état du drapeau d'interruption TXRDY. Les bits 3 à 1, BR2 à BR1 (sélection de la vitesse de transmission) sélectionnent la vitesse de transmission active de l'U.A.R.T. et dépendent de la fréquence de l'oscillateur interne. Il faut faire attention à ne pas changer ces bits durant la communication car l'écriture de ces bits a un effet immédiat. Le bit 0 DAT9 (parité ou bit 9 de la donnée) représente le neuvième bit de la donnée qui est transmise ou reçue. Une écriture dans ce bit établit le niveau pour le bit 9 qui doit être transmis ; aussi, il doit être mis à un niveau correct avant la transmission. S'il est utilisé comme bit de parité, la valeur doit être calculée en premier par logiciel. La lecture de ce bit retourne le neuvième bit de la donnée reçue. Nous allons mettre en garde, dans ce paragraphe, l'utilisateur afin qu'il évite des interruptions parasites durant l'initialisation du microcontrôleur ST6230. Si durant la séquence de RESET, la ligne de réception RXD/PD4 est au niveau zéro, une transition de l'état haut vers l'état bas sera verrouillée à l'intérieur de la cellule de l'U.A.R.T. due à la structure du RESET, même si aucun front descendant réel ne s'est produit sur la ligne RXD/PD4. Ce front actif descendant sera reconnu par la cellule de l'U.A.R.T. comme un bit valide start de réception. 88 périodes d'horloge de l'U.A.R.T. (8 périodes par bit) seront alors nécessaires pour recevoir un train de 11 bits (1 bit de start + 8 bits de données + 1 bit de parité + 1 bit de stop). Le drapeau d'interruption en réception RXRDY est positionné au niveau 1 après la réception de ces 11 bits et si elle est validée, une interruption en réception est ensuite générée. Afin d'éviter cette interruption de l'U.A.R.T. précoce, il faut valider l'interruption en réception de l'U.A.R.T. uniquement après une séquence logicielle générant un retard qui sera inséré juste après l'instruction de jump du RESET. Le drapeau RXRDY doit aussi être mis à 0 avant ou au moment de la validation de l'interruption en réception de l'U.A.R.T. pour la première fois. La longueur du temps de retard dépend de la vitesse de transmission sélectionnée avec les bits BR2 à BRO du registre UARTCR.

Voici un sous-programme qui réalise cette fonction :

La **figure 6** représente le temps nécessaire par rapport à la vitesse de transmission sélectionnée. Il faut aussi noter que la configuration pour la vitesse

```

Reset :      ldi          UARTCR,0ch      ; sélectionne la vitesse de transmission la plus rapide
ldi          count,01h                    ; chargement de la valeur de temporisation dans une variable mémoire
call        tempo                          ; dans cet exemple, la temporisation est de 100 µs à 8 MHz
Start :      ldi          UARTCR,3Ch      ; mise à zéro du drapeau RXRDY (bit 7). les interruptions de l'UART
                                                    peuvent maintenant être validées
...
                                                    ; programme principal
Tempo : ld    a,count                      ; temporisation totale = (62 cycles horloge) x (count) + (16 cycles horloge)
jz          endtemp
loop1 :      ldi          x,08h
loop2 :      dec         x                  ; par exemple, il faut 62 cycles horloge pour 100 µs à 8 MHz
jnz        loop2
dec        count
nop
nop
jnz        loop1
endtemp
.org        0FFEH
jp         Reset
    
```

de transmission la plus rapide doit être sélectionnée (BR2 = 1, BR1 = 1, BRO = 0) après la séquence de RESET de manière à utiliser un temps de retard le plus court possible. Cette interruption en réception précoce de l'U.A.R.T. peut aussi être utilisée pour détecter, après la séquence de RESET, si la ligne RXD/PD4 est au niveau zéro ou non.

## Description du circuit et du programme réalisant la communication série asynchrone

La **figure 7** représente le schéma de notre montage d'application réalisant une communication série asynchrone pour le transfert de données en mode sans handshake (c'est-à-dire utilisant seulement les lignes RXD et TXD) par le protocole RS232 avec une vitesse de transmission de 9600 bauds, 1 bit de start + 8 bits de donnée + 2 bits de stop. Vous trouverez ci-contre un petit programme permettant de lire, puis d'écrire une valeur à travers cette liaison série asynchrone.

## Réalisation pratique

Le câblage de notre circuit ne pose aucune difficulté particulière. Il est bien sûr recommandé de mettre le ST6230 sur

### Définition des registres

a	def	0ffh,0ffh,0ffh,m	; accumulateur a
DWDR	.def	0d8h,0ffh,0ffh	; registre du watchdog
IOR	.def	0c8h,00h,0ffh	; registre option des interruptions
DRD	.def	0c3h,0ffh,0ffh	; registre données du port D
DDRD	.def	0c7h,0ffh,0ffh	; registre de direction du port D
ORD	.def	0cfh,0ffh,0ffh	; registre option du port D
UARTDR	.def	0d6h,0ffh,0ffh	; registre données de l'UART
UARTCR	.def	0d7h,0ffh,0ffh	; registre de contrôle de l'UART

### Initialisation de la liaison RS232

```

ldi          DWDR,0feh      ; watchdog dévalidé
clr          IOR            ; toutes les interruptions sont dévalidées
ldi          DDRD,2fh       ; PD4 configuré en entrée avec 'pull-up'
ldi          ORD,25h        ; et PD5 configuré en sortie avec 'push-pull'
ldi          DRD,21h
ldi          UARTCR,07h     ; interruptions UART dévalidées, 9600 Bauds
                                                    -1 bit de start + 8 bits de données + 2 bits de stop
    
```

### Sous-programme de réception

```

txd : jrr    6,UARTCR,txd   ; attente que la transmission précédente soit terminée
ldi          a,55h          ; chargement de la valeur 55h dans a
ld          UARTDR,a        ; transmission de la valeur 55h par la liaison RS232
    
```

### Sous-programme de transmission

```

rxd : jrr    7,UARTCR,rxd   ; attente de la réception d'une donnée
ld          a,UARTDR        ; chargement de la valeur reçue dans a
    
```

BR2	BR1	BR0	Fosc Division	Fosc = 8 MHz	Time Delay Required
0,00	0,00	0,00	6656	1200	9,6 ms
0,00	0,00	1,00	3328	2400	4,8 ms
0,00	1,00	0,00	1664	4800	2,4 ms
0,00	1,00	1,00	832	9600	1,2 ms
1,00	0,00	0,00	416	19200	600 $\mu$ s
1,00	0,00	1,00	256	31200	400 $\mu$ s
1,00	1,00	0,00	208	38400	300 $\mu$ s
1,00	1,00	1,00	Reserved	Reserved	Reserved

BR2	BR1	BR0	Fosc Division	Fosc = 4 MHz	Time Delay Required
0,00	0,00	0,00	6656	600	19,2 ms
0,00	0,00	1,00	3328	1200	9,6 ms
0,00	1,00	0,00	1664	2400	4,8 ms
0,00	1,00	1,00	832	4800	2,4 ms
1,00	0,00	0,00	416	9600	1,2 ms
1,00	0,00	1,00	256	15600	740 $\mu$ s
1,00	1,00	0,00	208	19200	600 $\mu$ s
1,00	1,00	1,00	Reserved	Reserved	Reserved

## 6 Temps nécessaire par rapport à la vitesse de transmission

support dans le cas l'on désire changer son programme interne. A noter qu'il existe deux versions : le ST62E30 qui est une version reprogrammable car il contient une EPROM effaçable et le ST62T30 qui est une version programmable une seule fois. La **figure 8** représente le circuit côté composants et la **figure 9** côté pistes. Les lignes du programme décrites ci-dessus sont à insérer dans le programme principal de l'utilisateur afin qu'il puisse développer sa propre application.

## Conclusion

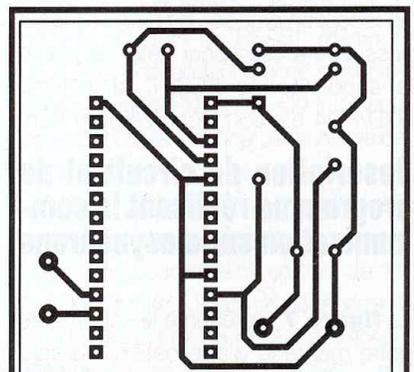
Bien que très peu encombrant dans un montage, le ST6230 possède de nombreux périphériques internes qui permettent de réaliser de nombreuses applications dans des domaines très variés. Dans cet article, nous avons réalisé, grâce à l'emploi du port de communication série asynchrone (U.A.R.T.) interne du microcontrôleur, un circuit simple permettant d'échanger des données par une liaison RS232. Le lecteur qui désire approfondir toutes les

possibilités ainsi que les registres de ce compteur 16 bits auto-rechargeable peuvent se référer au livre écrit par l'auteur de cet article aux éditions ETSF sur le microcontrôleur ST6230. De plus, il est à noter que toute une gamme d'outils de développement pour le ST6230 est disponible dont un assembleur, un compilateur C, un Debugger, un programmeur et un Starter-Kit qui permet à l'utilisateur de faire fonctionner ses applications sur une petite carte d'évaluation et aussi de programmer son composant à l'aide d'un ordinateur.

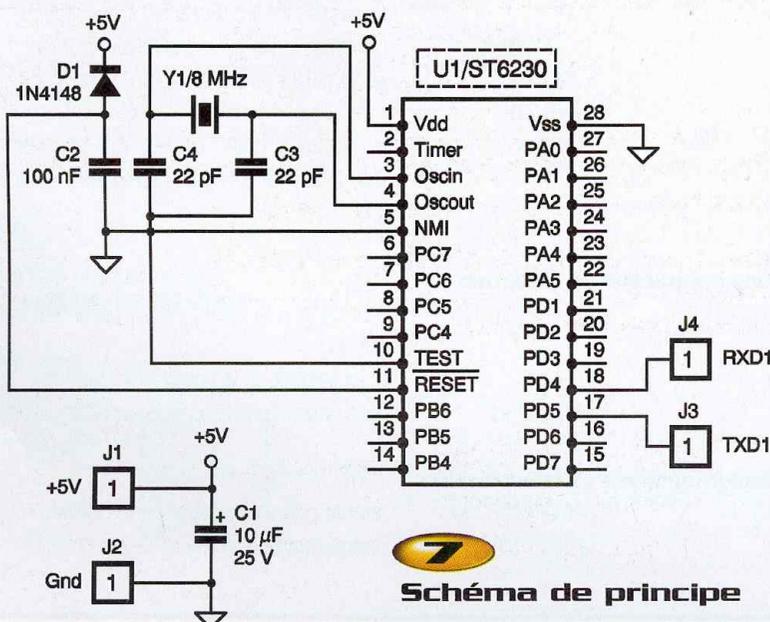
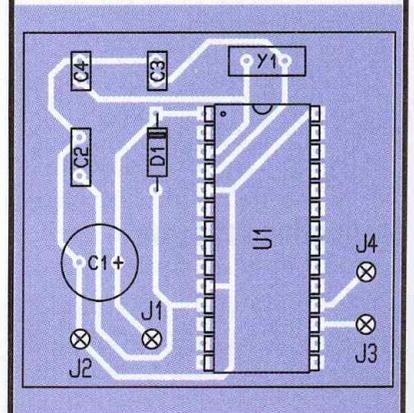
**M. LAURY**

## Nomenclature

- J<sub>1</sub> à J<sub>4</sub> : 4 picots**
- U<sub>1</sub> : ST6230 + support DIL28 broches**
- Y<sub>1</sub> : quartz 8 MHz**
- D<sub>1</sub> : diode 1N4148**
- C<sub>1</sub> : 10  $\mu$ F/25V**
- C<sub>2</sub> : 100 nF**
- C<sub>3</sub>, C<sub>4</sub> : 22 pF**



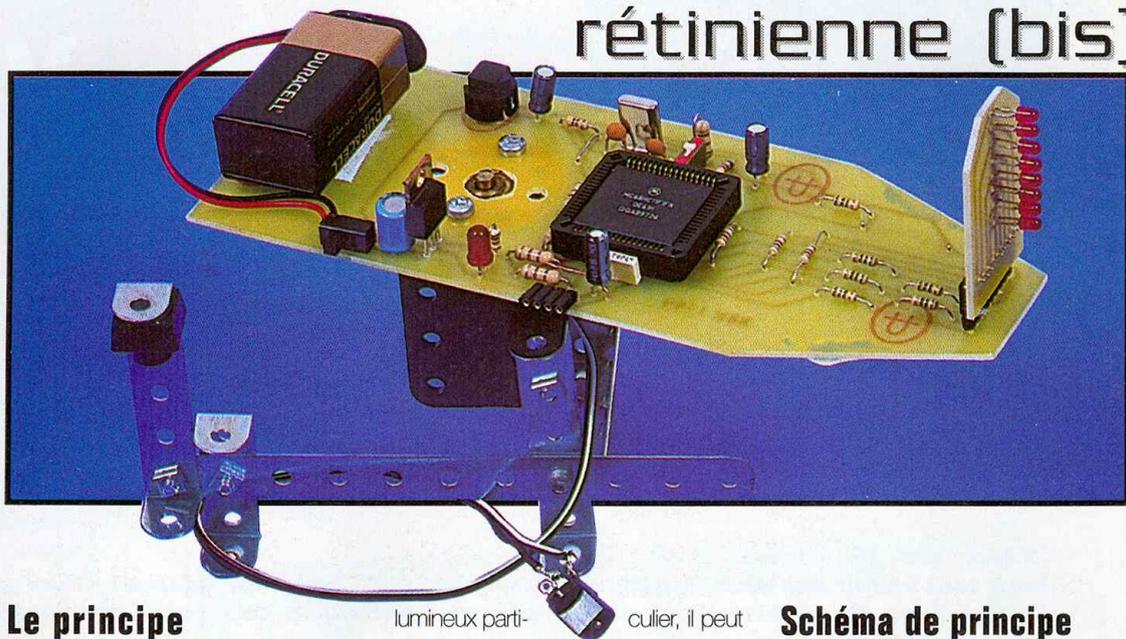
## 8/9 Tracé du circuit imprimé et implantation des éléments



## 7 Schéma de principe

# Journal lumineux

## à persistance rétinienne (bis)



### Le principe (figure 1)

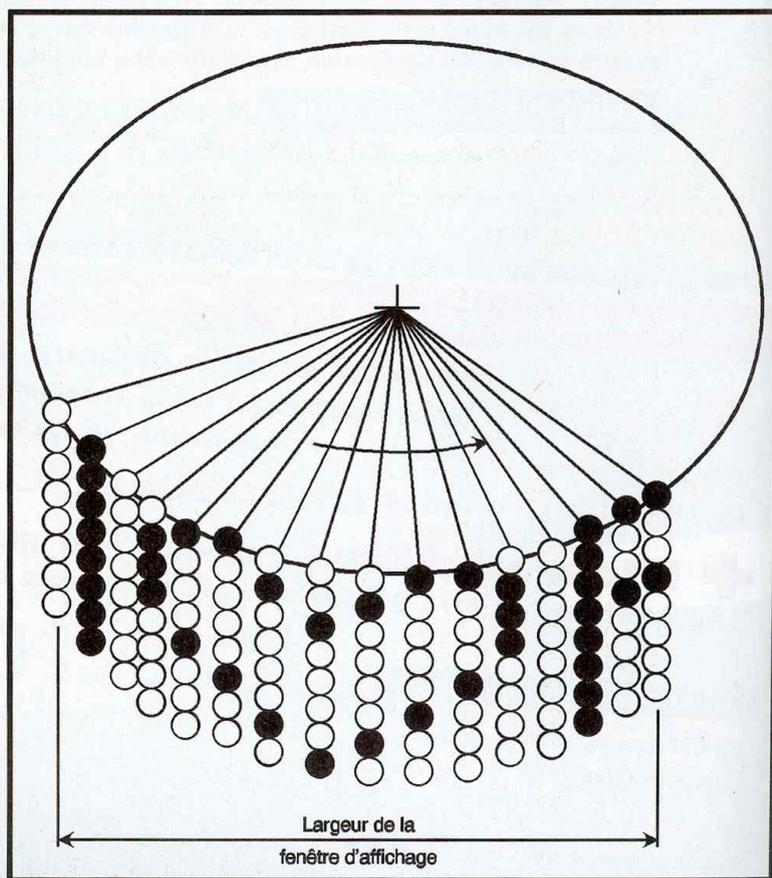
Une colonne de 8 LED se déplace dans un plan horizontal sur la circonférence d'un cercle à une vitesse de rotation donnée. L'état de chaque LED, allumée ou éteinte, dépend de la position angulaire de la colonne. À chaque tour, plusieurs fois par seconde, les LED sont allumées ou éteintes au même endroit : la rétine de l'œil d'un observateur va avoir l'impression d'un affichage fixe permanent. Le signal déclenchant l'allumage de la colonne de LED est produit par un interrupteur ILS fixé sur le circuit et fermé à chaque tour sous l'influence d'un aimant fixé sur le bâti moteur. Sans trop s'éloigner de ce principe, le 68HC11 sera programmé de telle sorte que le texte ne soit pas totalement fixe devant les yeux de l'observateur, mais défile doucement de droite à gauche : les mots successifs du texte n'apparaissent plus et ne disparaissent plus un par un par à-coup comme dans la version précédente mais progressivement lettre par lettre. La souplesse du logiciel permettra aussi de choisir la largeur en nombre de colonne de l'affichage : suivant la localisation et l'emploi de ce panneau

lumineux parti-culier, il peut en effet être intéressant de choisir une largeur particulière de l'affichage.

### 1 Principe du journal lumineux

### Schéma de principe (figure 2)

Il est des plus réduits, ce qui est souvent le cas dans les montages à base de microcontrôleurs. Au cœur



Dans le numéro 225, nous vous présentons déjà un système utilisant la persistance rétinienne. La gestion des LED était confiée à 2 circuits intégrés et le message lumineux était stocké dans une EPROM 2716. Dans cette nouvelle version, tout est confié à un microcontrôleur 68HC11F1 qui contient dans son EEPROM les données de l'affichage et le programme de gestion. Pour un emploi simplifié et une utilisation immédiate, la programmation du 68HC11F1 est prise en charge par le logiciel "TOURNIQUET", disponible en version MAC et PC sur notre site Internet.



LED allumée = ■      Valeur bit = 1

LED8 bit 7	■ ■ ■		
LED7 bit 6	■ ■ ■		
LED6 bit 5	■ ■ ■		
LED5 bit 4	■ ■ ■		
LED4 bit 3	■ ■ ■		
LED3 bit 2	■ ■ ■		
LED2 bit 1	■ ■ ■		
LED1 bit 0	■ ■ ■		

hexa 81 FF 81 00 00 00 00 70 88 84 42 21 42 84 88 70 00 00 00 00 FF 90 90 F0 00 00 FF 90 90 FF 00 00 FF 98 94 F3 00 00 FF 00 00 F1 91 9F 00

**3** Conversion du message lumineux vers hexadécimal

l'intermédiaire des résistances R<sub>11</sub> à R<sub>18</sub> de 560 Ω limitant le courant débité par le 68HC11 à quelques mA, sans surcharge donc pour le microcontrôleur. L'entrée PE5 est reliée à l'interrupteur ILS.

Les 2 broches PDO (Rx) et PD1 (Tx) du 68HC11 sont reliées à la carte interface MAC/PC par l'intermédiaire de deux petits connecteurs. L'adaptation des signaux, réalisés par un très classique MAX232, est confiée à une carte particulière qui ne sert qu'à la programmation et permet d'avoir une platine mobile plus petite et plus légère que si ces composants y avaient été intégrés.

### Programmation du 68HC11

Le microcontrôleur 68HC11F1 possède une RAM de 1 koctets et une EEPROM de 512 octets. Malgré une taille plus réduite, il a été choisi d'utiliser l'EEPROM plutôt que la RAM pour enregistrer le programme d'exploitation des données et les données de l'affichage : la programmation en RAM est volatile et il aurait été nécessaire de recharger les données après tout arrêt.

Les 512 octets de l'EEPROM devront être utilisés judicieusement : faire le plus petit programme d'exploitation possible pour avoir la plus grande place disponible possible pour les données. Les données sont stockées à suivre à partir de l'adresse FEC0, le programme d'exploitation étant lui placé à partir de l'adresse FE00 du début d'EEPROM.

#### Format des données

Le tableau de la **figure 3** illustre à titre d'exemple la conversion d'un message lumineux en son équivalent hexadécimal.

#### Routines principales

##### Test de l'état de l'ILS

L'affichage du message lumineux ne débute qu'après le passage du circuit mobile au-dessus de l'ILS. C'est l'objet de la boucle d'attente suivante :

```
TESTE    LDAA PORTE
          ANDA #%00100000
          CMPA #%00100000
          BEQ  TESTE
avec PORTE EQU $100A, adresse du port E
```

#### Affichage sur les LED

Après avoir quitté la boucle d'attente précédente, le programme va afficher sur le port B le contenu des données successives du message lumineux. La durée de l'affichage est générée par la boucle de temporisation TEMPO. L'adresse du début d'affichage est stockée dans le registre d'index X. La largeur de la fenêtre d'affichage est stockée dans la variable LARGEUR.

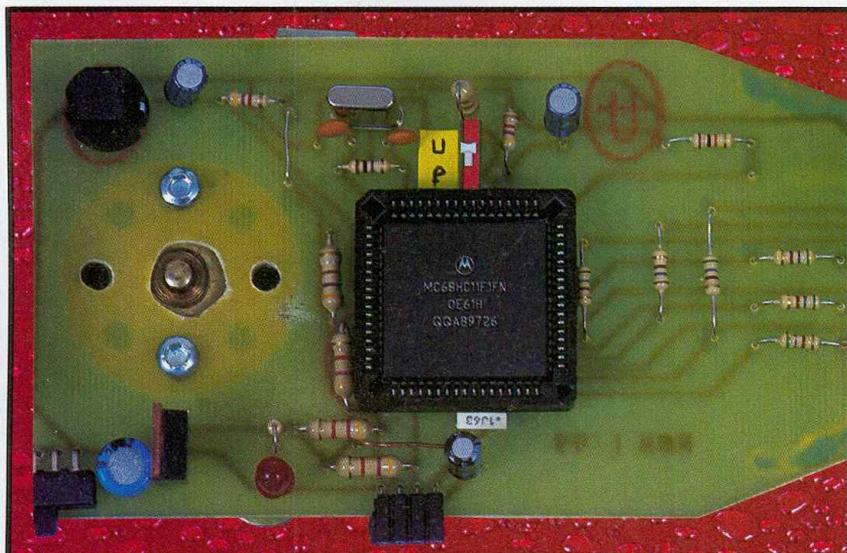
```
          CLRB
SUIITE   ABX
          LDAA X
          STAA PORTB
          BSR TEMPO
          INCB
          CMPB LARGEUR
          BNE SUIITE
          BRA DEBUT (du programme, test de l'ILS)
```

avec PORTB EQU \$1004, adresse du port B

```
TEMPO   LDY# Durée de la temporisation
          NOP
          NOP
          NOP
          DEY
          BNE WAIT
          RTS
```

#### Autres routines :

- le programme va nécessiter quelques lignes supplémentaires en particulier :
- pour incrémenter le registre d'index X à chaque tour de manière à créer un défilement du message lumineux de la droite vers la gauche,
  - pour remettre le registre d'index X à la valeur FEC0 après qu'il ait atteint la valeur de fin des données,
  - pour revenir en début de zone de donnée et afficher la première colonne du message après la dernière colonne.



**gros plan sur le 68HC11 F1FN**

## Partie fixe, support moteur et alimentation

La platine tournante est mise en rotation par un moteur de MECCANO fixé sur un petit support. Dans la réalisation de l'auteur de l'article, le support est tout simplement constitué d'une armature de pièces de MECCANO. L'alimentation du moteur est confiée à un bloc secteur classique fournissant des tensions de 3V et 4,5V.

## Description du logiciel "TOURNIQUET"

Pour l'utilisateur du logiciel "TOURNIQUET", programmer la gestion de l'allumage des LED et convertir le message lumineux en données hexadécimales est particulièrement simple. Après avoir ouvert l'application en double cliquant sur l'icône, il suffit de colorier les ronds figurants les LED sur le panneau qui s'affiche en suivant des règles simples :

- pour dessiner une forme quelconque, il faut la colorier, LED par LED, en cliquant dans les ronds,
- pour écrire une lettre, il faut soit la colorier, LED par LED, en cliquant dans les ronds, soit cliquer dans la colonne où on veut qu'elle débute et frapper la touche du clavier qui correspond,
- il est possible en appuyant sur les boutons ad hoc d'insérer ou de supprimer une colonne en ayant préalablement cliquer sur

la colonne à supprimer ou à l'endroit où faire apparaître l'insertion.

D'autres paramètres sont intéressants à modifier suivant les besoins :

### Nombre total de colonnes :

Le message lumineux tourne en boucle fermée : après l'affichage de la dernière colonne déclarée, s'affiche à nouveau la première colonne. Si le message lumineux occupe 100 colonnes, il est possible par exemple :

- de déclarer que la longueur du message fait 100 colonnes : après la centième, l'affichage revient à la première,
- de déclarer que la longueur du message est supérieure (par défaut 200) : après l'affichage des 100 premières colonnes avec certaines LED allumées, toutes les suivantes seront éteintes jusqu'au retour de la première colonne.

### Nombre de colonnes affichées :

Si le message lumineux ne doit être vu que suivant un angle donné par rapport à une position centrale, il est possible de limiter le nombre de colonnes et donc la largeur de l'affichage. Au contraire, si l'on désire que le message soit vu sur 360°, il est possible d'agrandir la fenêtre d'affichage en augmentant le nombre de colonne de l'affichage.

Il n'est pas possible d'avoir un message lumineux de plus de 410 colonnes mais ce n'est déjà pas si mal. Cette limitation est due à la taille de l'EEPROM qui, sur le 68HC11F1, est de 512 octets : le pro-

gramme de gestion occupe 102 octets, il n'en reste que 410 octets pour des données.

### Durée de l'affichage :

Certains lecteurs trouveront peut-être que les lettres sont trop larges ou trop minces, d'autres modifieront peut-être le montage pour augmenter le rayon du cercle parcouru par la colonne lumineuse. Il est bien sûr possible de jouer sur la vitesse de rotation du moteur mais modifier la durée d'affichage des LED permet de garder une vitesse de rotation donnée et d'optimiser la perception de persistance rétinienne.

### Sauvegarde des messages lumineux

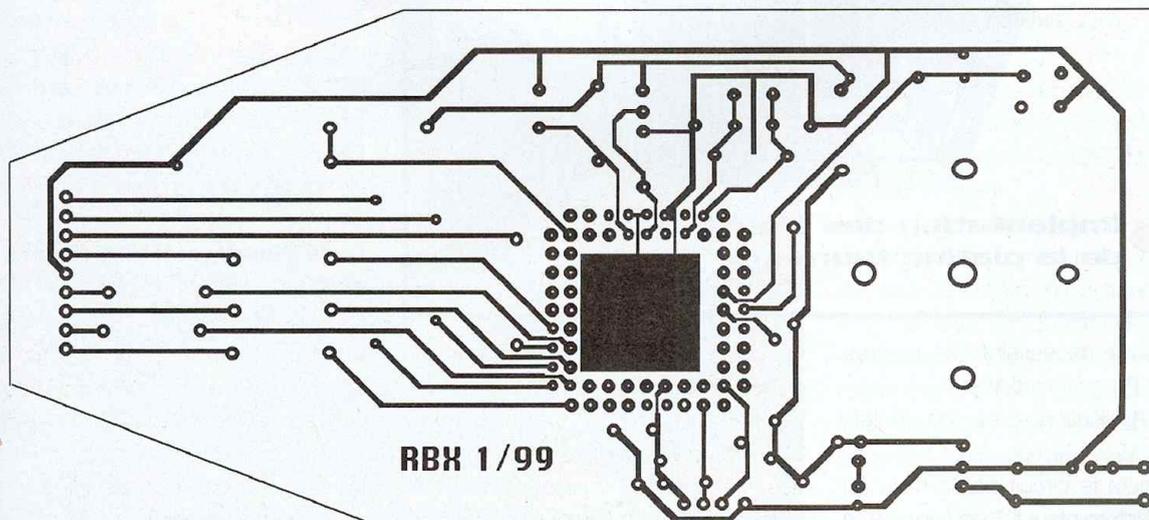
Ils sont enregistrés au format .txt et peuvent donc être lus par tout traitement de texte. Vous remarquerez qu'en plus du message proprement dit sont aussi sauvegardés les 3 paramètres précédents en hexadécimal.

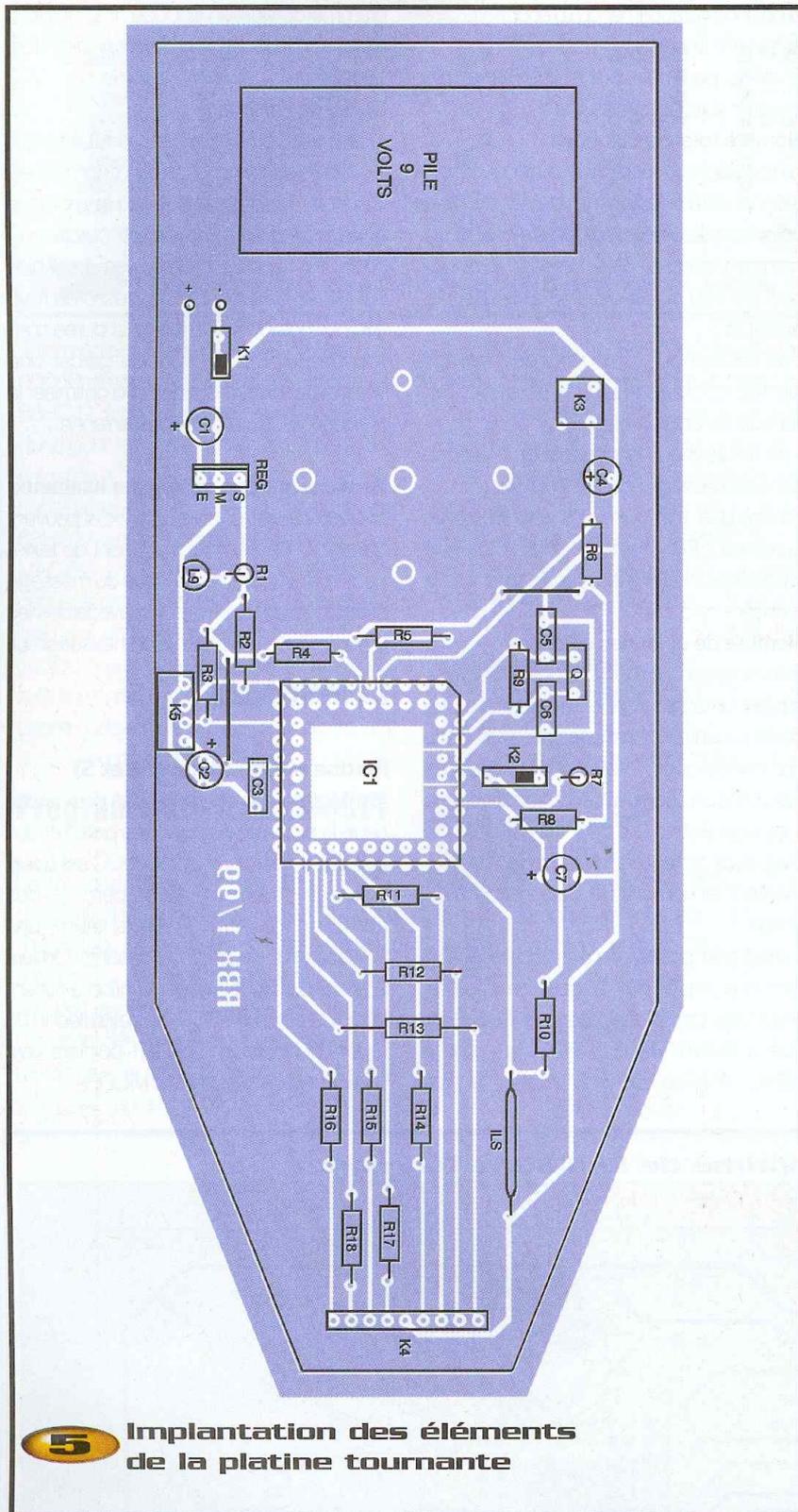
## La réalisation

### Platine mobile (figures 4 et 5)

Elle ne pose pas de problèmes particuliers, seule la proximité de quelques pistes et des pastilles autour du support PLCC est à surveiller attentivement. Après perçage des trous, on passera à l'implantation des 2 straps et des autres composants. On fera là aussi particulièrement attention à l'orientation du support PLCC : la petite flèche de repère qui indique la patte 1 pointera vers l'interrupteur à glissière de MODB.

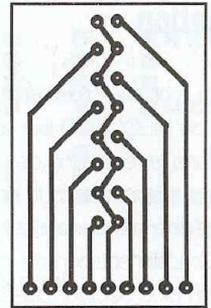
## 4 Tracé du circuit imprimé de la platine tournante





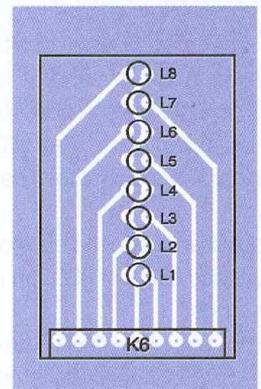
On inscrira si nécessaire des repères MODB=0/**P**rogrammation et MODB=1/**U**tilisation près de l'interrupteur  $K_2$ .  
On terminera le circuit par la mise en place de l'interrupteur ILS en faisant très attention à la capsule de verre. On pourra

même enrober cet interrupteur dans une grosse goutte de colle pour bien le protéger des chocs.  
On achèvera la platine mobile en fixant des contreponds de manière à ce qu'elle soit parfaitement équilibrée autour de l'axe de rotation.



**6/7**

**Tracé du circuit imprimé et implantation des éléments de l'affichage**



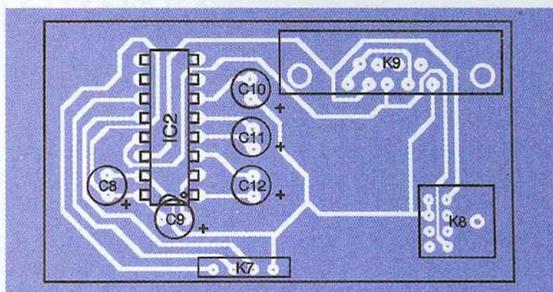
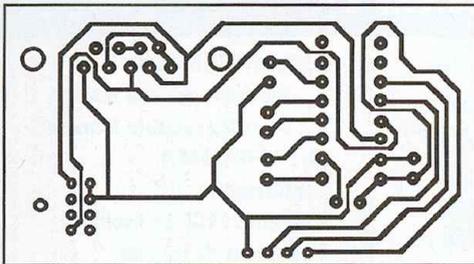
**Colonne de LED (figures 6 et 7)**

Le tracé compliqué du circuit mobile simplifie d'autant le tracé du circuit de la colonne de LED. On notera la position centrale de la masse et l'orientation alternée des LED.

**Remarque :** nous avons compliqué à dessein le tracé du circuit mobile pour que l'état 1 ou 0 du bit n de chaque octet de données visualise l'état allumé ou éteint de la LED n. Il était possible de tracer le circuit plus simplement mais la correspondance Bit/LED n'aurait pas été immédiate et aurait compliqué la programmation.

**Carte d'interface MAC/PC (figures 8 et 9)**

La réalisation de cette carte ne doit pas poser de problèmes particuliers. On choisira de souder le connecteur DB9 pour PC ou le connecteur mini DIN 8 pour MAC suivant son équipement informatique. La carte interface et l'ordinateur seront reliés par un câble de rallonge série.



8/9

## Tracé du circuit imprimé et implantation des éléments de la carte interface Mac/PC

### Partie fixe

#### Structure du support :

Chacun est libre d'imaginer le support qu'il veut. La réalisation de la photo représente un minimum mais respecte quelques besoins indispensables : le système doit être stable pour s'affranchir d'un éventuel défaut d'équilibrage de la partie mobile. L'aimant fermant l'ILS doit être convenable positionné, un peu à l'extérieur de la trajectoire de l'ILS (voir **figure 10**).

#### Alimentation du moteur :

Celle-ci étant confiée à un bloc secteur du commerce, il n'y a plus qu'à brancher le moteur en s'assurant que la platine mobile, vue de haut, tournera dans le sens trigonométrique.

### Mise en œuvre du tourniquet lumineux

#### Programmation

- Connectez l'interface série entre votre ordinateur et la carte tourniquet,

- Assurez-vous que l'interrupteur à glissière  $K_2$  est sur  $MODB=0$  (**P**) pour obtenir le mode BOOTSTRAP,

- Alimentez la carte tourniquet en fermant l'interrupteur  $K_1$ . La LED  $L_3$  doit s'allumer, - Double-cliquez sur l'icône du programme "TOURNIQUET",

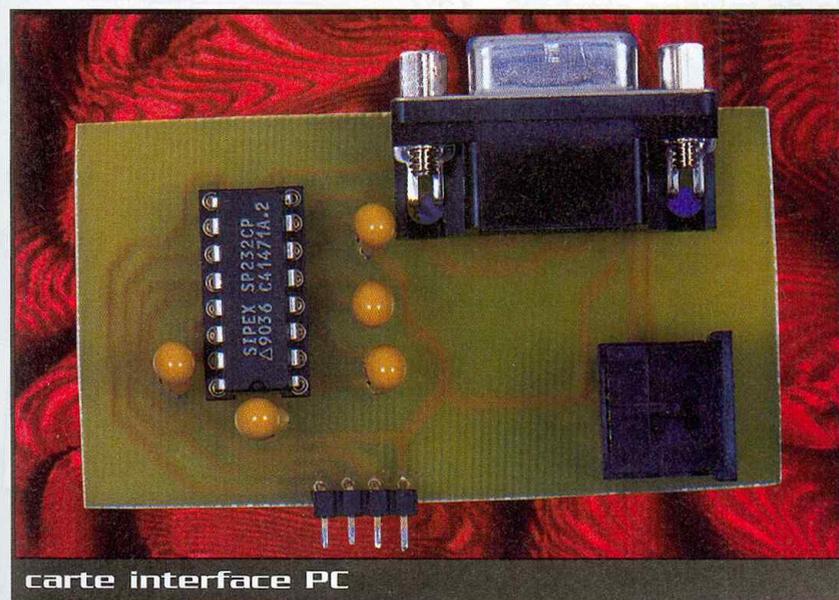
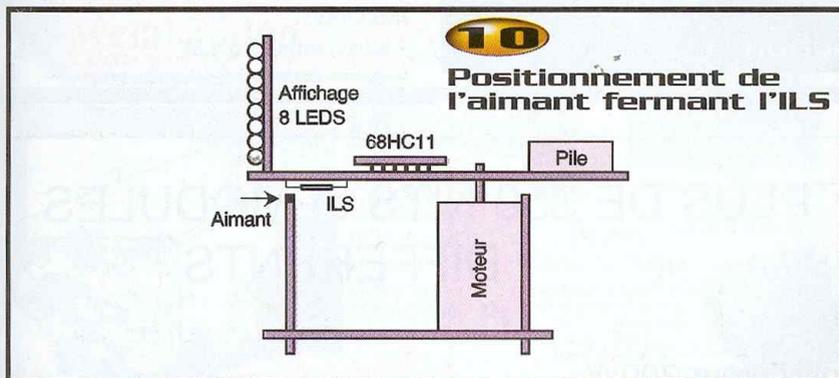
- Sélectionnez le port série COM1 ou COM2. Test éventuel de la liaison série et du montage en sélectionnant l'item essai du menu «Transmission». Après quelques secondes, les LED de la colonne lumineuse doivent alors clignoter. Refaire un reset en appuyant sur le bouton poussoir  $K_3$ ,

- Frappe d'un message lumineux suivant les indications du paragraphe 5,

- Envoi du programme à la carte tourniquet en sélectionnant l'item envoi du menu «Transmission». La programmation dure quelques instants (15 à 30 secondes),

- Déplacer l'interrupteur à glissière  $K_2$  pour obtenir le mode circuit seul avec  $MODB=1$  (**U**),

- Déconnectez l'interface série. Mettez la carte hors tension.



carte interface PC

#### Utilisation

- Mettez votre montage dans un endroit dégagé,

- Alimentez la carte tourniquet,

- A la main, faites défiler l'ILS au-dessus de l'aimant permanent : les LED doivent s'allumer brièvement,

- Alimentez le moteur, votre message lumineux doit défiler comme par magie devant vos yeux (la lumière ambiante ne doit pas être trop forte). On pourra jouer sur l'affichage des mots en modifiant la vitesse de rotation par le choix de la tension d'alimentation de 3V ou 4,5V ou en modifiant le paramètre "durée de l'affichage".

A. RIBOUX

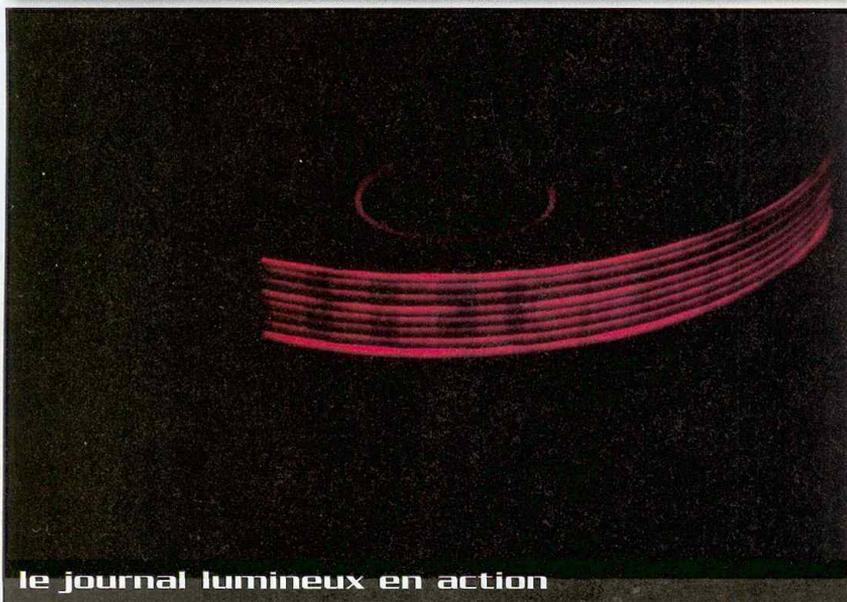
## Nomenclature

### Platine mobile

$R_1$  : 1 k $\Omega$  (marron, noir, rouge)  
 $R_2$  à  $R_8$  : 4,7 k $\Omega$  (jaune, violet, rouge)  
 $R_9$  : 10 M $\Omega$  (marron, noir, bleu)  
 $R_{10}$  : 10 k $\Omega$  (marron, noir, orange)  
 $R_{11}$  à  $R_{18}$  : 560  $\Omega$  (vert, bleu, marron)  
 $C_1$  : 220  $\mu$ F/25V  
 $C_2$  : 10  $\mu$ F/25V

$C_3$  : 100 nF  
 $C_4, C_7$  : 1  $\mu$ F  
 $C_5, C_6$  : 18 pF  
 $Q$  : quartz 8 MHz  
 $REG$  : régulateur 7805  
 $L_9$  : LED 3 mm  
 $K_1$  : interrupteur d'alimentation pour CI  
 $K_2$  : interrupteur DIL simple

$K_3$  : bouton poussoir  
 $K_4$  : barrette sécable male  
 $K_5$  : barrette sécable femelle  
 $IC_1$  : 68HC11F1FN  
 1 interrupteur ILS  
 1 support PLCC 68 broches  
 1 coupleur de pile 9V  
 2 straps



### Carte interface MAC/PC

$K_7$  : barrette sécable mâle soudée  
 $K_8$  : connecteur mini DIN 8 pour MAC  
 $K_9$  : connecteur DB9 pour PC  
 $IC_2$  : MAX232  
 $C_8$  à  $C_{12}$  : 1  $\mu$ F  
 1 support 16 broches

### Carte LED

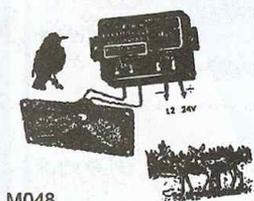
$K_6$  : barrette sécable femelle soudée  
 $L_1$  à  $L_8$  : LED 3 mm haute luminosité

### Autres

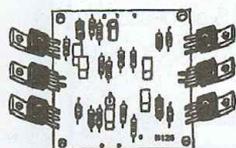
1 aimant permanent  
 1 moteur MECCANO 3 à 6V  
 1 support moteur (quelques pièces de MECCANO)  
 1 alimentation 3 à 4,5V

# KEMO

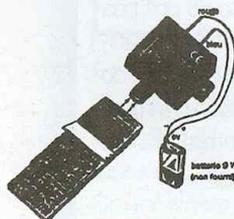
## PLUS DE 350 KITS et MODULES DIFFERENTS



M048

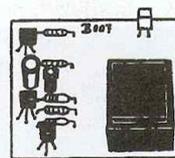


B125

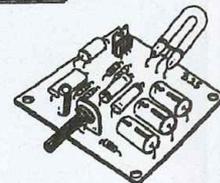


M085

- B125 Amplificateur 200W
- B131 Table de mixage stéréo 6...12V
- M032 Amplificateur 12W, universel
- B025 Stroboscope
- B160 Vu-mètre à leds - 5 leds, 12V
- B007 Appareil à électriser 9...12V=
- B188 Mini-générateur de tonalité 6...12V
- M048 Générateur d'ultra-sons 12...24V
- B216 Barrière de lumière infra rouge 3m
- B048 Interrupteur thermique 12V
- B136 Acupuncture électronique
- B192 Contrôleur de niveau d'eau 9V
- B217 Alerte à la fumée
- M085 Détecteur d'infra-rouge 9V
- M013 Interrupteur crépusculaire 230V~



B007



B025



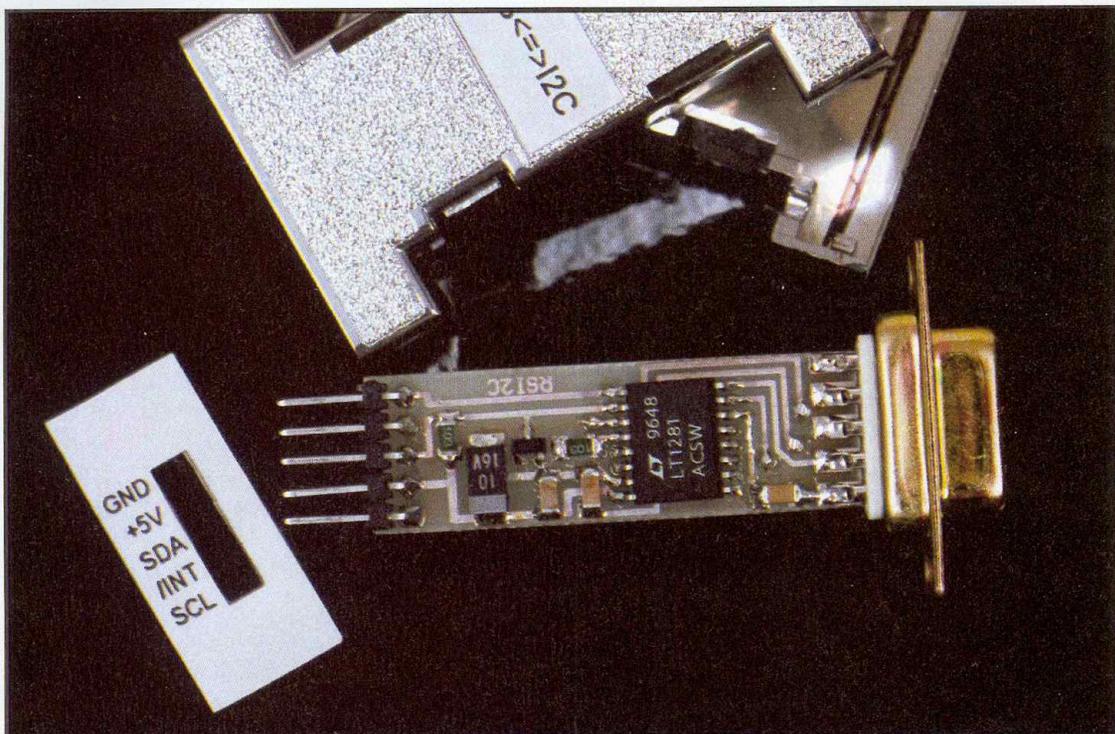
B136

Catalogue et liste des dépositaires contre 30,00 F en timbres frais d'envoi compris.

DISTREL : 8 av. du 18 Juin 1940 - 92500 RUEIL MALMAISON  
 aucune vente directe,

# Interface

# RS232 - I2C



## Le matériel

Cette interface peut servir soit à commander des périphériques I2C, soit à commander de manière simple des composants lors des étapes de mise au point. Elle se caractérise par une bonne fiabilité et l'absence de risque de détérioration des composants testés. En effet, si on ne prend pas un certain nombre de précautions, les tensions présentes sur la RS232 ( $\pm 12V$ ) peuvent détruire les composants I2C (0/+5V). Pour notre réalisation, ces précautions se sont matérialisées dans un driver RS232 (IC<sub>1</sub>) qui permet d'interfacer sans risque

les deux types de signaux (c'est sa fonction). Pour ceux qui connaissent le fonctionnement du bus I2C, notre réalisation permet de simuler une station maître. La possibilité de fonctionner en mode «multi-mâtres» n'est pas possible, ce qui n'est pas gênant pour le type d'applications visées. Le signal d'horloge SCL (serial clock) est généré directement par le signal DTR de la liaison RS232. De même,

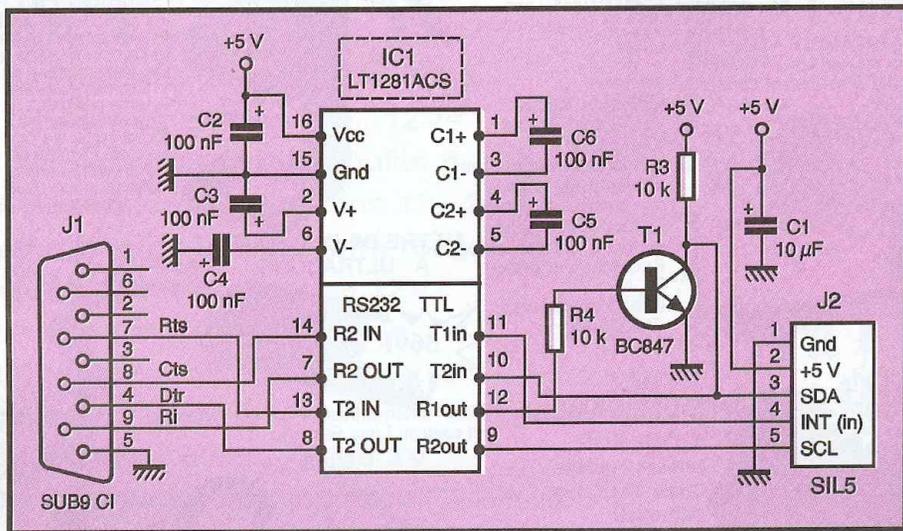
le signal d'interruption /INT disponible sur certains composants (comme le PCF8574) est récupéré par l'intermédiaire du signal CTS. La seule petite difficulté concerne le signal SDA (serial data).

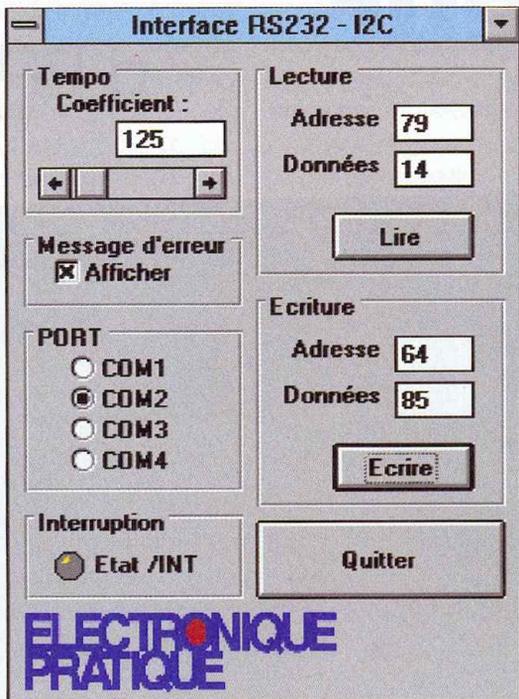
Ce signal est bidirectionnel contrairement aux buffers du driver qui ne le sont pas. Il est donc nécessaire de réaliser artificiellement la commutation, c'est le rôle du transistor T<sub>1</sub>.

De plus en plus de composants sont pourvus d'une interface I2C. Il est donc indispensable de pouvoir facilement tester ces composants. L'interface que nous vous proposons utilise la liaison RS232. De plus, elle présente l'avantage d'être assez compacte. De nombreuses interfaces I2C ont été décrites dans notre revue. Néanmoins, nous vous en proposons une nouvelle qui permet de transformer une liaison RS232 en bus I2C.

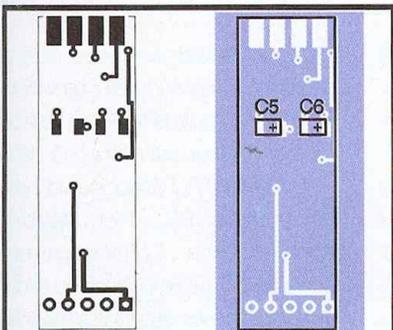


**Schéma de principe**



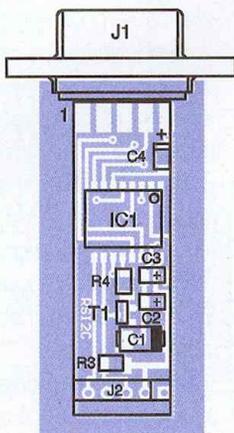
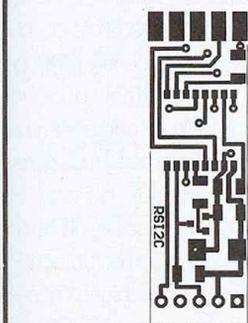


**2** Écran...



**3** Tracé du circuit imprimé et implantation des éléments

**4** Tracé du circuit imprimé et implantation des éléments côté RS232



Lorsque l'on veut écrire sur le bus I2C, il faut agiter RTS (SDAout) en fonction des données et de manière synchrone avec DTR (SCL). Et lorsque l'on veut lire, il faut placer RTS (SDAin) à «1» et lire l'état du signal RI (SDAin) en synchronisation avec DTR (SCL).

Pour en finir avec le matériel, le driver doit être en boîtier SOP.

Pour notre part, nous avons utilisé un circuit de chez Linear Technology, le LT1281ACS, mais vous pourrez utiliser un MAX232 de Maxim ou un HIN232 de chez Harris. La seule précaution à prendre est

de bien utiliser les bonnes capacités pour le convertisseur à pompe de charge. Le transistor est un BC847 en boîtier SOT23.

**Le logiciel**

La simplicité du matériel implique un logiciel un peu plus complexe. En effet, la gestion du bus I2C est entièrement confiée au logiciel. Pour cela nous avons réalisé une DLL pour Windows et un petit utilitaire permettant de lire et d'écrire sur le Bus I2C.

La zone coefficient permet de régler la vitesse du bus I2C en fonction des performances du PC.

- Pour lire, il faut placer l'adresse et cliquer sur le bouton «lire».

- Pour écrire, il faut placer l'adresse, ainsi que la donnée puis cliquer sur le bouton «Écrire».

Les autres zones permettent de choisir le port série, de valider l'affichage des erreurs et de voir l'état du signal /INT disponible sur certains composants I2C.

Si l'utilisation de composants CMS vous semble hors de portée et/ou que vous désirez en savoir plus sur la liaison série, nous vous conseillons la lecture de l'ouvrage «La liaison RS232» chez ETSF. Dans ce livre, vous trouverez une version en composants standards (non CMS) de l'interface RS/I2C ainsi que d'autres réalisations et enfin tous les renseignements techniques sur la liaison série RS232.

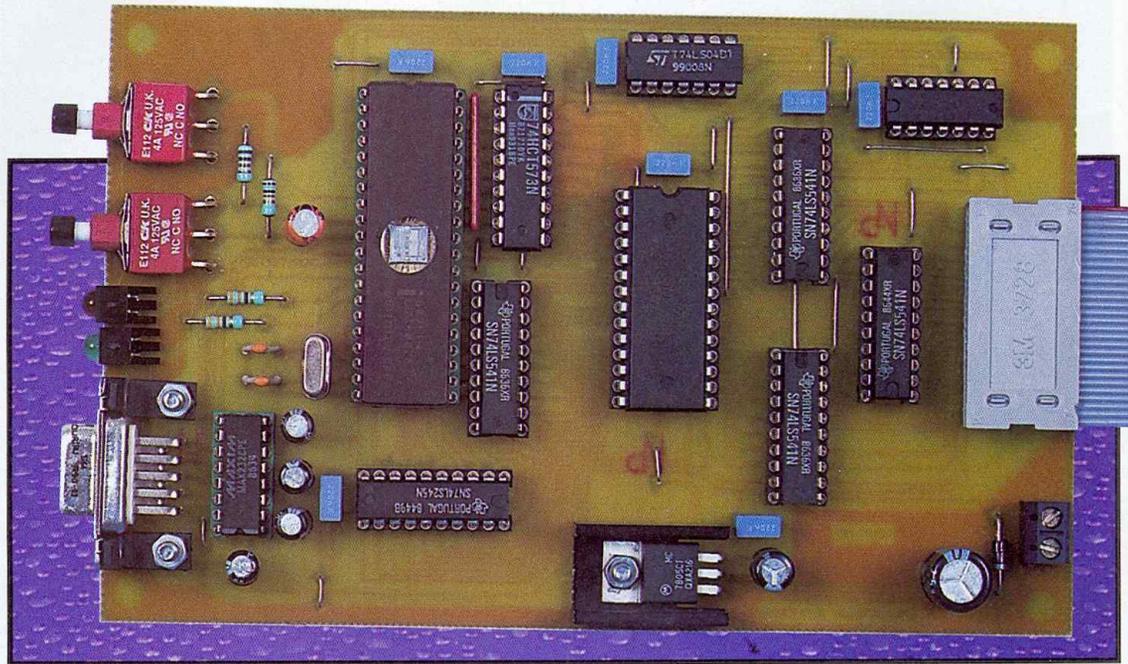
PH. ANDRE

**Nomenclature**

- C<sub>1</sub> : 10 µF/6032
- C<sub>2</sub> à C<sub>6</sub> : 100 nF/1206
- R<sub>3</sub> et R<sub>4</sub> : 10 kΩ/1206
- T<sub>1</sub> : BC847/SOT23 ou équiv.
- IC<sub>1</sub> : LT1281ACS ou MAX232SOP ou équivalent
- J<sub>1</sub> : SUB9 femelle
- JO<sub>2</sub> : 5 broches coudées

# Emulateur

# d'EPROM



Pour mettre au point un système à microprocesseur, on fait souvent appel à un émulateur temps réel. Il s'agit d'un appareil très coûteux que l'amateur éclairé a bien du mal à rentabiliser. Il est pourtant possible de se passer de ce type d'appareil en ayant recours à un émulateur d'EPROM qui est un outil bien moins coûteux.

L'émulateur d'EPROM que nous vous proposons de réaliser ce mois-ci est capable de se substituer aux EPROM 27C64, 27C128 et 27C256. Le chargement du code s'effectue via un port série à 19200 bauds. Notre émulateur reconnaît un seul format de fichier, le format HEXA INTEL. Fort heureusement, c'est le plus répandu. De plus, il existe de nombreux programmes de transformation des fichiers dans différents formats qui sont disponibles gratuitement sur Internet.

## Schémas

Le principe d'un émulateur d'EPROM est très simple. On s'arrange pour placer le code dans une RAM qui remplacera l'EPROM sur le circuit. Il faut donc 2 accès au contenu de la RAM et 1 arbitre pour sélectionner la source qui prend le contrôle du bus de la RAM. Il va de soit que du côté du montage cible, l'accès à la RAM ne permet que les opérations de lecture.

Les schémas de notre émulateur

d'EPROM sont visibles en **figures 1** et **2**. Comme vous pouvez le constater sur la figure 1, le cœur du montage est un microcontrôleur 87C51. Le choix de ce microcontrôleur est facile à expliquer. Il nous fallait tout d'abord disposer d'un port série intégré dans le microcontrôleur, la famille 8051 étant bien sympathique pour cette raison. Ensuite, pour permettre une implantation raisonnable du montage, il nous a fallu opter pour un microcontrôleur qui dispose d'une EPROM interne. Vous remarquerez sur le schéma que le microcontrôleur est câblé pour adresser une RAM externe. De ce fait, on pourrait penser que l'ajout d'une EPROM externe ne doit pas être si compliqué. Ceci est vrai sur le papier, mais il en va tout autrement lors de l'implantation et du routage du circuit imprimé. Lorsque vous verrez le nombre de straps qui sont nécessaires pour réaliser le circuit imprimé, vous ne pourrez qu'approuver notre choix.

En définitive, le schéma n'est pas vraiment compliqué. Le microcontrôleur U<sub>2</sub> dispose d'un oscillateur interne qui

nécessite seulement un quartz et deux condensateurs pour être fonctionnel. Le circuit de remise à zéro du microcontrôleur est composé de la simple cellule RC associée au bouton poussoir BP<sub>2</sub>. Rien de bien méchant donc. L'adaptation des signaux RS232 est confiée au circuit U<sub>1</sub> qui n'est autre que le très répandu MAX232. Notez que la connectique retenue (CN<sub>1</sub>) permet de connecter notre montage par un câble direct sur une sortie série de type IBM/PC/AT. Les ports P0 et P2 du microcontrôleur forment le bus externe sur lequel sera connecté une RAM. Le port P0 présente successivement le poids faible du bus des adresses et le bus des données. Le signal ALE indique la présence du poids faible du bus des adresses. Le latch U<sub>3</sub>, piloté par le signal ALE, permet donc de reconstituer le bus adresses. Les résistances RR<sub>1</sub> permettent de garantir l'état des signaux du port P0 puisque ce dernier dispose de sorties à drain ouvert. Les sorties du port P1 contrôlent les quelques lignes nécessaires pour télécharger l'EPROM et

pour sélectionner le type d'EPROM à émuler. Les bits 2 et 3 du port P3 contrôlent les diodes LED DL<sub>1</sub> et DL<sub>2</sub> qui seront utilisées pour indiquer les différents états de notre émulateur. La diode LED rouge DL<sub>1</sub> indiquera les erreurs de transmission ou de commande. La diode LED verte indiquera l'état fonctionnel du montage.

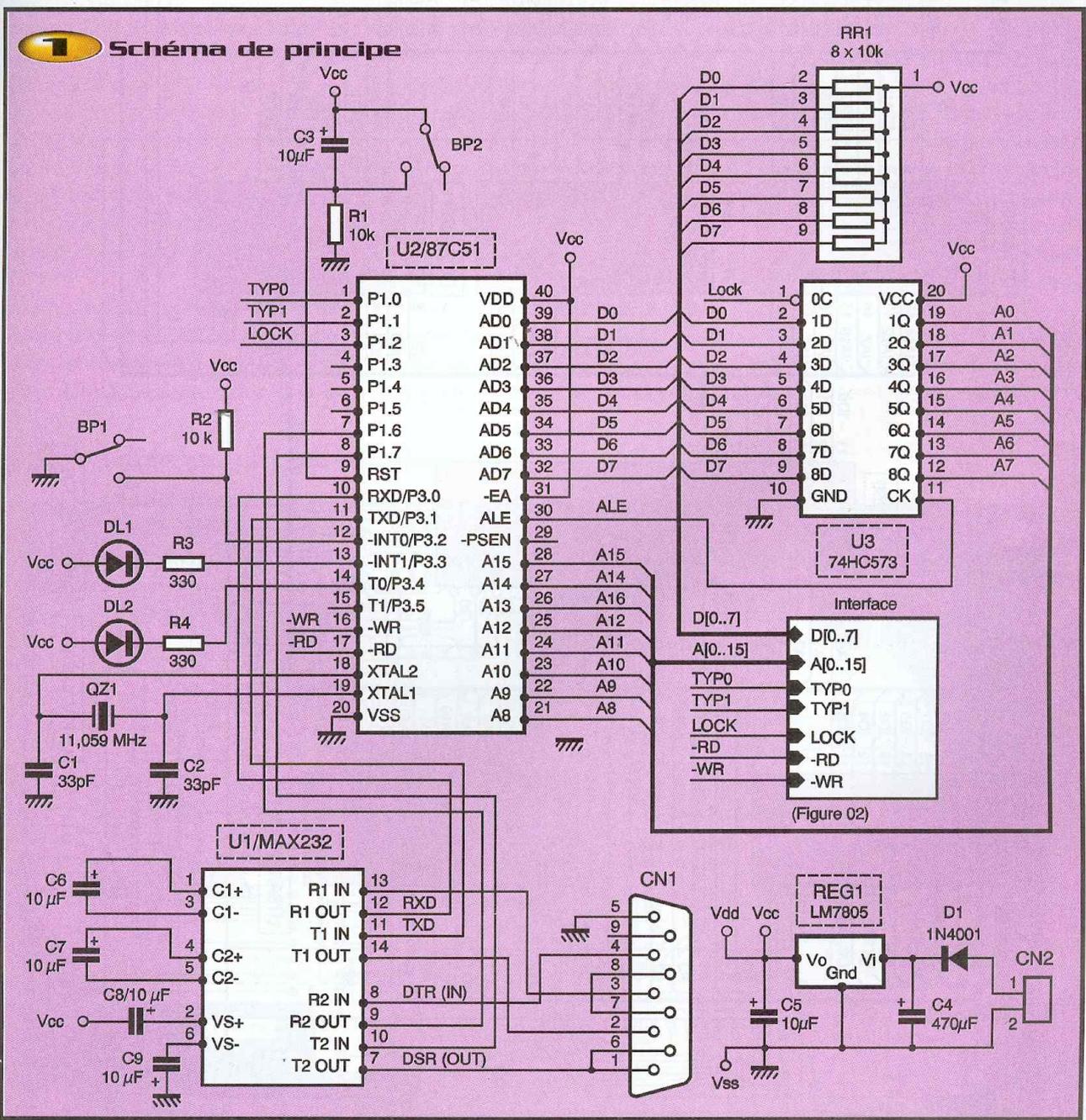
Le bouton poussoir BP<sub>1</sub> génère le signal d'interruption -INT0. Le programme du microcontrôleur utilisera cette interruption pour changer l'état de l'émulateur et le type d'EPROM sélectionnée. Nous reviendrons sur la manipulation de BP<sub>1</sub> un peu plus loin dans cet article.

L'alimentation du montage se fera à partir d'un petit bloc secteur (type alimentation de calculatrice) capable de fournir 200mA sous 12VDC. Notez que la diode D<sub>1</sub> permet de protéger le montage en cas d'inversion du connecteur d'alimentation.

Le schéma d'accès à la RAM qui remplacera l'EPROM à émuler est donné en figure 2. Le poids fort du bus d'adresse du microcontrôleur U<sub>2</sub> aboutit sur un driver 3 états (U<sub>3</sub>). Quant au poids faible du bus d'adresse, il transite par le latch U<sub>3</sub> qui dispose également de sorties 3 états commandées par la broche 1 (OC). Les bus de données et d'adresse du montage cible,

qui reçoit le connecteur CN<sub>3</sub> (en lieu et place de l'EPROM à émuler) aboutissent aux drivers 3 d'états U<sub>3</sub> à U<sub>9</sub>. Entre tous ces drivers 3 états, on trouve la RAM (U<sub>11</sub>) qui sera partagée entre le microcontrôleur et le montage cible. La mémoire retenue pour ce montage est une 62256 qui peut donc couvrir les 32koctets d'espace adressé par une EPROM de type 27C256.

Les drivers 3 états U<sub>3</sub>, U<sub>4</sub> et U<sub>6</sub> sont contrôlés par le signal LOCK qui est issu directement du bit 2 du port P1. Lorsque le signal LOCK est à l'état bas, c'est le microcontrôleur qui prend le contrôle de la RAM. Le signal LOCK est inversé par la porte U<sub>13A</sub>





pour sélectionner les drivers  $U_8$  à  $U_9$ , de sorte que l'émulateur est actif uniquement quand le signal LOCK passe à l'état haut. Les portes  $U_{13B}$ ,  $U_{13C}$ ,  $U_{12C}$  et  $U_{12D}$  contrôlent les signaux de lecture et d'écriture de la RAM. L'arrangement retenu permet d'autoriser l'écriture et la lecture de la RAM par le microcontrôleur tant que le signal LOCK est à l'état bas. Lorsque le signal LOCK passe à l'état haut, le microcontrôleur ne peut plus influencer les lignes -OE et -WR de la RAM. Dans ce cas de figure, le contenu de la mémoire  $U_{11}$  est figé en lecture uniquement, ce qui est heureux.

Le signal de lecture (-OE) et le signal de sélection de boîtier (-CS) issus du montage cible, via  $CN_3$ , commandent les sorties du driver  $U_8$ . Ainsi les données de sortie de l'émulateur seront bien validées par les signaux du montage cible. Le type d'EPROM à émuler est pris en compte grâce aux portes  $U_{12A}$  et  $U_{12B}$  qui masquent les bits d'adresses selon le type actif. Le microcontrôleur commande directement les signaux de sélection de type TYP0 et TYP1.

Vous noterez que l'entrée de sélection du boîtier de la RAM  $U_{11}$  est portée au niveau 0 en permanence. La RAM est donc active en permanence. Cette situation

peut poser un petit problème. Lorsque l'émulateur est en mode LOCK et que le driver de sortie  $U_8$  n'est pas activé (par exemple -CS=1 sur le connecteur  $CN_3$ ) notre mémoire voit quand même défiler les adresses du montage cible. Au moment où le montage cible décide de sélectionner notre émulateur, il peut y avoir apparition d'un octet fugitif. Cette situation se produit si le bus d'adresse n'était pas stable depuis un laps de temps correspondant au temps d'accès de la RAM. Cela signifie que notre émulateur peut poser des problèmes dans un contexte où la cible n'est pas un système à microprocesseur. Le problème est visible, en particulier, si on tente de relire le contenu de notre émulateur sur un programmeur d'EPROM. Ajoutons que notre émulateur ignore les lignes VPP et VCC du montage cible, ce qui n'est pas sans intérêt. Mais en dehors des cas de figure que nous venons d'évoquer, notre émulateur d'EPROM se comporte parfaitement.

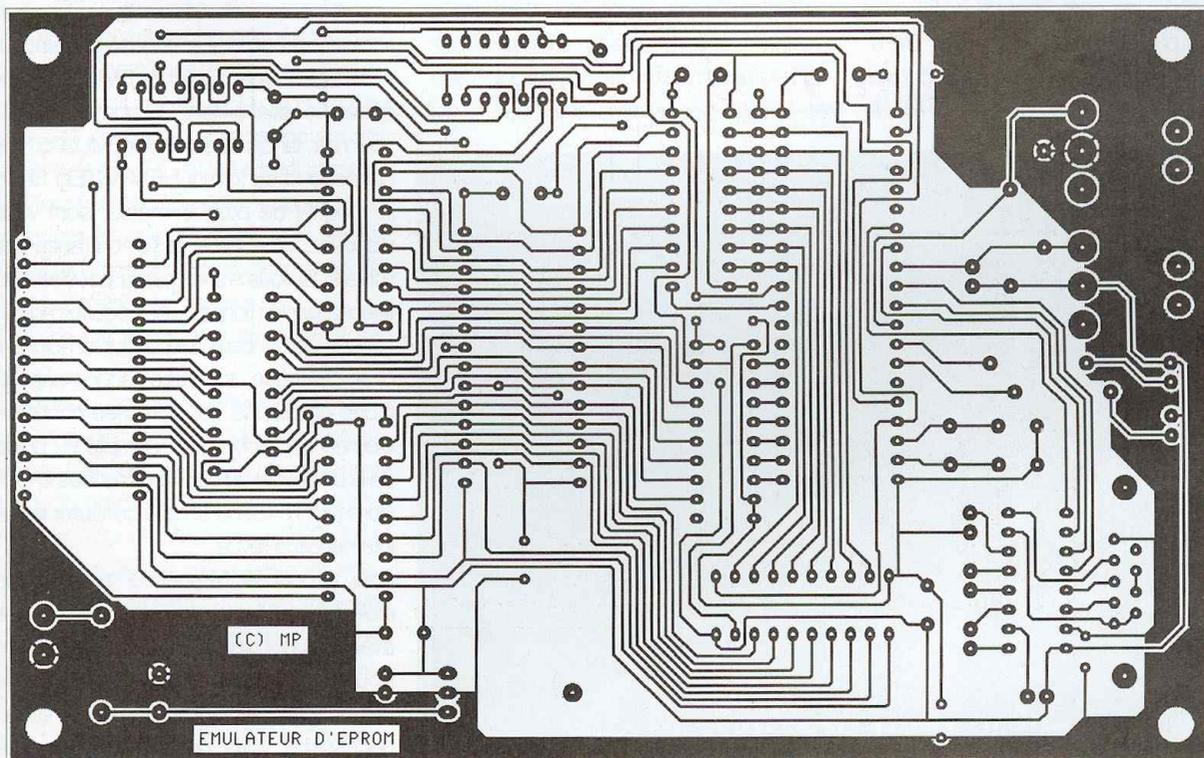
Vous noterez avec intérêt que les circuits  $U_7$  à  $U_9$  sont de technologie TTL-LS sur le schéma. Pour être parfaitement compatible avec des EPROM de type CMOS, vous pourrez utiliser des circuits de technologie HCT. L'auteur a essayé les deux types de

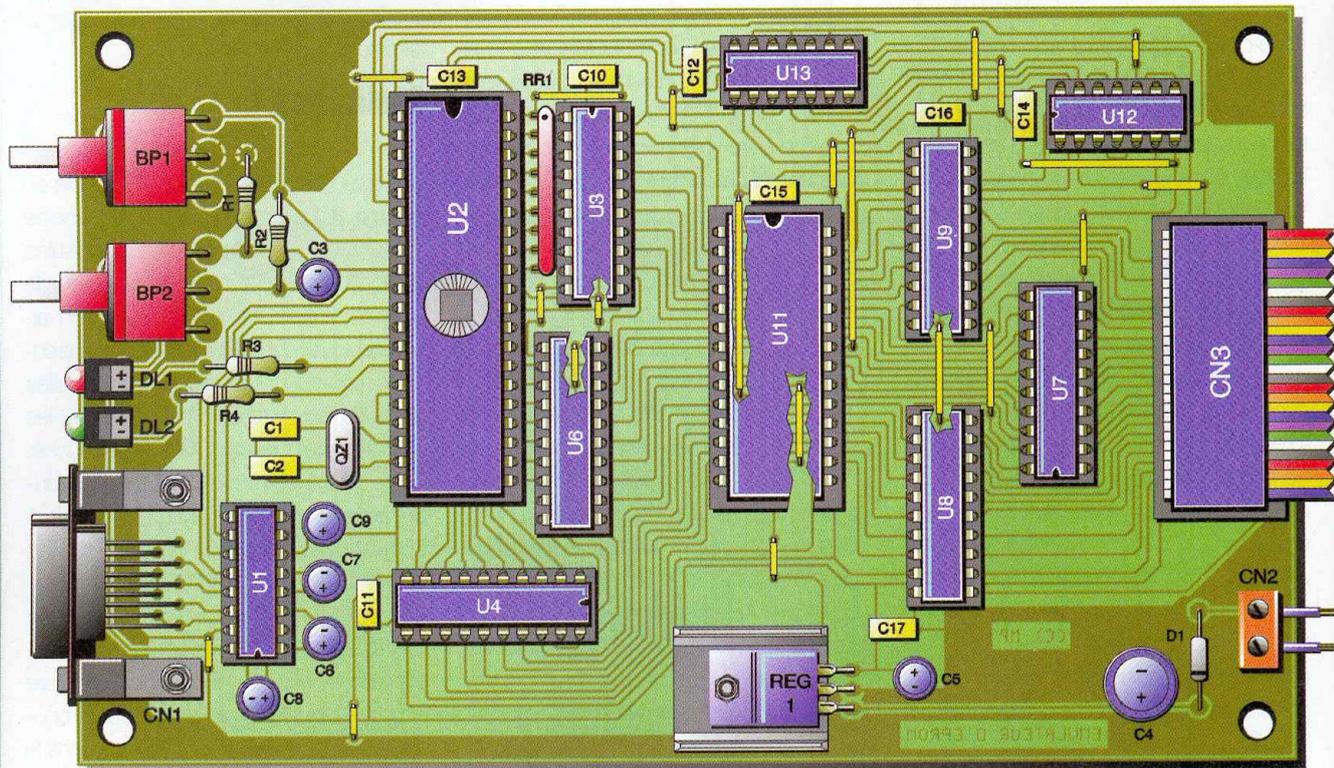
circuit sur sa maquette, cela fonctionne parfaitement.

## Réalisation

Le dessin du circuit imprimé est visible en **figure 3**. 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 raison de la taille réduite des pastilles associées aux straps, il vaudra mieux utiliser des forets de bonne qualité pour éviter de les emporter au moment où le foret débouche. En ce qui concerne  $CN_2$ ,  $REG_1$  et  $D_1$ , il faudra percer les pastilles avec un foret de 1 mm de diamètre. En ce qui concerne les boutons poussoirs, il faudra percer les pastilles avec un foret de 2 mm de diamètre. Il n'y a pas de difficulté particulière pour l'implantation. Respectez bien la nomenclature et soyez très vigilant au sens des composants car les circuits  $U_9$  et  $U_7$  sont dans le sens inverse des autres. Vous noterez la présence de 21 straps qu'il est préférable d'implanter en premier pour des raisons de commodité. Respectez scrupuleusement le découplage des lignes d'alimentations si vous voulez éviter les surprises. Le régulateur  $REG_1$  sera impérativement monté sur

### 3 Tracé du circuit imprimé





#### 4 Implantation des éléments

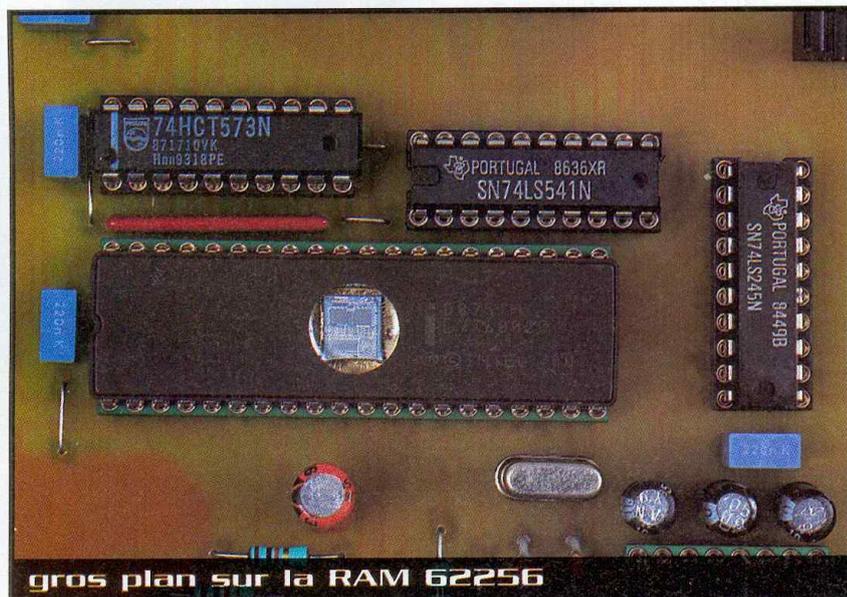
un radiateur ayant une résistance thermique inférieure à 18°C/W pour éviter d'atteindre une température de jonction trop élevée. Veillez bien à choisir un connecteur femelle pour CN<sub>1</sub>. Car un modèle mâle s'implante parfaitement, mais les points de connexions se retrouvent inversés par symétrie par rapport à l'axe vertical. Dans ce cas, il n'y a aucune chance pour que votre montage dialogue avec votre PC, à

moins de fabriquer un câble spécial pour rétablir l'ordre voulu. En ce qui concerne le câble nécessaire pour relier notre montage à un PC de type AT, il vous suffira de fabriquer un câble équipé d'un connecteur DB9 mâle d'un côté et d'un connecteur DB9 femelle de l'autre côté (liaison fil à fil de la broche 1 à la broche 9). L'utilisation de connecteurs à sertir est plus pratique, mais les liaisons nécessaires étant peu nom-

breuses vous pourrez utiliser des connecteurs à souder. Enfin, ajoutons que le connecteur CN<sub>1</sub> sera immobilisé par deux boulons montés dans les passages prévus à cet effet.

Le microcontrôleur U<sub>2</sub> sera programmé avec le contenu d'un fichier que vous pourrez vous procurer par téléchargement sur notre site Internet. Le fichier EMULP5.ROM qui est le reflet binaire du contenu à programmer, tandis que le fichier EMULP5.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 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) ou bien commander le CD-Rom (voir P. 15) ou encore consulter le site Internet de la revue.

Pour connecter l'émulateur au montage cible, vous allez devoir réaliser un câble avec des connecteurs 28 points à sertir. Pour sertir les connecteurs sur du câble plat, si vous ne disposez pas du matériel nécessaire, vous pourrez fabriquer une



gros plan sur la RAM 62256

petite cale en bois (voir la **figure 5**). La cale protégera les pattes du support à sertir. Ainsi vous pourrez sertir les connecteurs avec un petit étau. Choisissez de préférence du bois dur ou du plastique rigide pour fabriquer la cale.

Pour réaliser les dégagements nécessaires aux pattes du support à protéger, vous n'aurez qu'à donner deux coups de scie sur toute la longueur de la cale. Ne prévoyez pas un câble trop long entre l'émulateur et le montage cible, faute de quoi les effets de lignes rendront le montage inopérant. Par exemple, une longueur de câble de 30 à 40 cm est un maximum pour une maquette cible équipée d'un microprocesseur 8031 cadencé à 12 MHz.

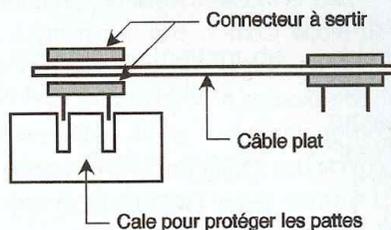
Pour protéger la vie du connecteur qui sera installé sur vos montages cibles, vous pourrez monter un support de type tulipe directement sur le connecteur fraîchement sertir. Les deux supports ne feront plus qu'un d'un point de vue logique. En cas de défection, il est plus simple de changer le support tulipe que de sertir un nouveau connecteur sur le câble. Si, en raison de la fréquence élevée du signal d'horloge le montage cible est sensible aux perturbations, il vaudra mieux vous passer du support tulipe supplémentaire.

## Comment utiliser l'émulateur d'EPR0M

Notre émulateur reçoit les données via un port série. Comme nous l'avons indiqué plus avant dans cet article, la connectique permet de connecter le montage sur un port série de type PC/AT à l'aide d'un câble direct (connecteur DB9 point à point). Pour communiquer correctement avec notre émulateur, le port série devra être configuré avec les paramètres suivants :

19200 bauds, 8 bits, pas de parité, 1 bit de stop.

L'émulateur accepte différentes commandes pour choisir le mode de fonctionnement. Les commandes se composent de 1 ou 2 caractères ASCII à émettre sur le port série. Pour choisir le type d'EPR0M à émuler, vous avez le choix entre les trois commandes suivantes : 'T0' sélectionne le type 27C64, 'T1' sélectionne le type 27C128 et 'T2' sélectionne le type 27C256. La commande 'C' permet d'effacer le contenu de la mémoire de l'émula-



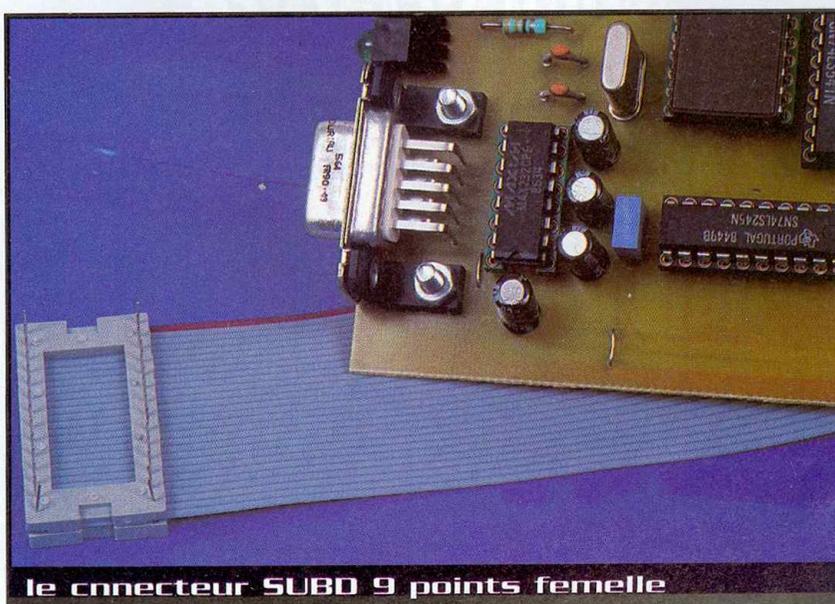
## 5 Fabrication d'une cale à sertir

teur, ce qui peut parfois être utile si le fichier HEXA à charger est constitué de plusieurs blocs discontinus.

Pour envoyer les commandes vers l'émulateur via le port série, vous avez plusieurs solutions. Vous pouvez soit utiliser un programme d'émulation de terminal tel que le programme HYPERTERMINAL qui est livré avec Windows 95/98, soit utiliser la commande COPY du DOS. Vous pouvez également ajouter les commandes au fichier qui contient le code hexadécimal, puisqu'il s'agit d'un fichier ASCII. En ce qui concerne la commande COPY du DOS, il vous suffit de re-diriger le contenu de la console vers le port série souhaité. Par exemple, pour re-diriger les commandes de la console vers le port COM1, il vous faudra taper la commande suivante : 'COPY CON COM1'. Ensuite vous n'avez qu'à saisir les commandes à partir du clavier du PC. Une fois la saisie terminée, utilisez la séquence CTRL-Z pour indiquer la fin de la saisie. C'est seulement après la séquence CTRL-Z que les commandes seront envoyées vers le port série.

Par défaut, après initialisation, l'émulateur est automatiquement configuré pour émuler le type 27C64. En réponse aux commandes 'T0' à 'T1', l'émulateur renvoie un message indiquant que le type demandé est bien pris en compte. Si les commandes sont erronées ou si les paramètres de communication sont incorrects, la diode LED rouge s'allume pour signaler l'erreur. La LED rouge s'éteindra seulement après réception d'une nouvelle commande correcte, cette fois-ci, ou après initialisation de l'émulateur. Avant de pouvoir charger des données dans la mémoire de l'émulateur, vous devrez veiller à ce que ce dernier soit dans le mode 'Commande'. La diode LED vert doit être allumée dans ce mode (feu vert). Si vous tentez de transférer un fichier dans la mémoire de l'émulateur tandis que ce dernier est dans le mode actif, la diode LED rouge s'allume.

Pour passer l'émulateur en mode actif, vous devez envoyer la commande 'L' (L mis pour Lock). Pour repasser en mode Commande vous devrez envoyer la commande 'U' (U mis pour Unlock). Certes le



le connecteur SUBD 9 points femelle

choix des commandes 'L' pour le mode Émulation et 'U' pour le mode Commande n'est pas très homogène. L'habitude des abréviations anglo-saxonnes a été la plus forte. Vous pouvez obtenir le même résultat que les commandes 'L' et 'U' grâce au bouton BP<sub>1</sub>. Un simple appui sur le bouton poussoir BP permet à l'émulateur de basculer successivement entre le mode Commande et le mode Émulation. La diode LED verte indique le mode actif.

Le bouton BP<sub>1</sub> permet également de changer le type d'EPROM à émuler. Pour cela, il faut appuyer deux fois de suite sur le bouton poussoir BP<sub>1</sub> pour passer au type suivant. Pour être certain que l'émulateur accepte la commande à partir du bouton poussoir, appuyez rapidement une première fois sur BP<sub>1</sub>, relâchez le bouton un bref instant, puis appuyez de nouveau sur BP<sub>1</sub> un laps de temps un peu plus long. L'émulateur passe alors au type d'EPROM suivant, en suivant l'ordre 27C64, 27C128, 27C256 puis retour au type 27C64. A chaque changement du type d'EPROM, l'émulateur anime un court instant les diodes LED pour indiquer le type sélectionné. Pour indiquer le type 27C64, seule la diode LED verte clignote. Pour indiquer le type 27C128, seule la diode LED rouge clignote. Enfin, pour indiquer le type 27C256, les deux diodes LED clignent. Après l'animation, les diodes LED reprennent leur état initial.

Notez que le bouton poussoir BP<sub>2</sub> qui permet de remettre à zéro le microcontrôleur du montage peut être gênant du fait qu'il est très proche de BP<sub>1</sub>. Vous pourrez décider de ne pas installer BP<sub>2</sub>, car il n'est pas fréquent d'avoir besoin de BP<sub>2</sub> (il a servi essentiellement à la mise au point du montage).

Pour télécharger du code dans la mémoire de l'émulateur, il suffit simplement de transférer le contenu d'un fichier au format HEXA INTEL vers le port série. Lorsque l'émulateur reçoit la dernière ligne du fichier HEXA INTEL (trame EOF), l'émulateur passe automatiquement en mode émulation. Ainsi vous ne risquez pas d'altérer involontairement le code nouvellement chargé. Pour transférer un fichier au format HEXA INTEL dans la mémoire de l'émulateur, vous pouvez utiliser le programme Hyperterminal de Windows 95/98. Mais vous pouvez utiliser la commande COPY dans une session DOS, tout simplement. Par exemple, pour copier le contenu du fichier «ESSAI.HEX» dans la mémoire de l'émulateur relié au port COM1 de votre PC, vous n'aurez qu'à lancer la commande : COPY ESSAI.HEX COM1. Notez que l'émulateur se charge de piloter les lignes DTR et DSR pour assurer un contrôle de flux Hardware lors des communications avec le PC.

P. Morin

## Nomenclature

**BP<sub>1</sub>, BP<sub>2</sub> :** boutons poussoirs coudés à souder sur circuit imprimé (ex. : C&K E112SD1AQE)

**CN<sub>1</sub> :** connecteur SubD 9 points femelle, sorties coudées, à souder sur circuit imprimé (ex. : HARTING 09 66 112 7601)

**CN<sub>2</sub> :** bornier de connexion à vis 2 plots, au pas de 5,08 mm, à souder sur circuit imprimé, profil bas.

**CN<sub>3</sub> :** support 28 broches + connecteur 28 pattes à sertir sur câble plat

**C<sub>1</sub>, C<sub>2</sub> :** 33 pF céramique au pas de 5,08 mm

**C<sub>3</sub>, C<sub>5</sub> à C<sub>9</sub> :** 10 µF/25V sorties radiales

**C<sub>4</sub> :** 470 µF/25 sorties radiales

**C<sub>10</sub> à C<sub>17</sub> :** 220 nF

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

**DL<sub>2</sub> :** diode LED verte 3mm

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

**QZ<sub>1</sub> :** quartz 11,059 MHz

**REG :** régulateur LM7805 (5V) en boîtier TO220 + dissipateur thermique 18°C/W [ex. : SHAFFNER réf. RAWA 400 9P]

**RR<sub>1</sub> :** réseau résistif 8x10 kΩ en boîtier SIL

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

**R<sub>3</sub>, R<sub>4</sub> :** 330 Ω 1/4W 5% (orange, orange, marron)

**U<sub>1</sub> :** driver de lignes MAX232

**U<sub>2</sub> :** microcontrôleur 87C51 avec EPROM interne (12 MHz)

**U<sub>3</sub> :** 74LS573 ou 74HCT573

**U<sub>4</sub> :** 74LS245 ou 74HCT245

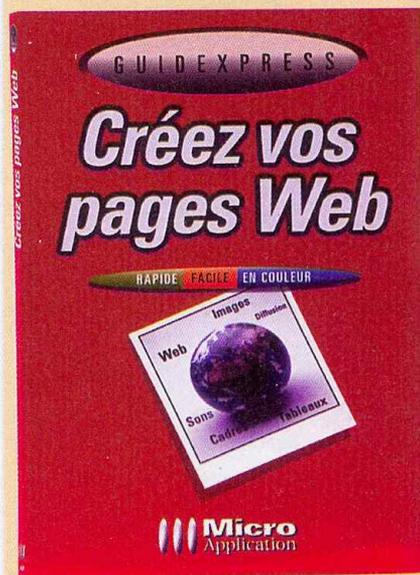
**U<sub>6</sub> à U<sub>9</sub> :** 74LS541 ou 74HCT541

**U<sub>11</sub> :** RAM 62256 temps d'accès 200 ns

**U<sub>12</sub> :** 74LS08 ou 74HCT08

**U<sub>13</sub> :** 74LS04 ou 74HCT04

**Créez vos pages Web Rapide, facile, en couleur**



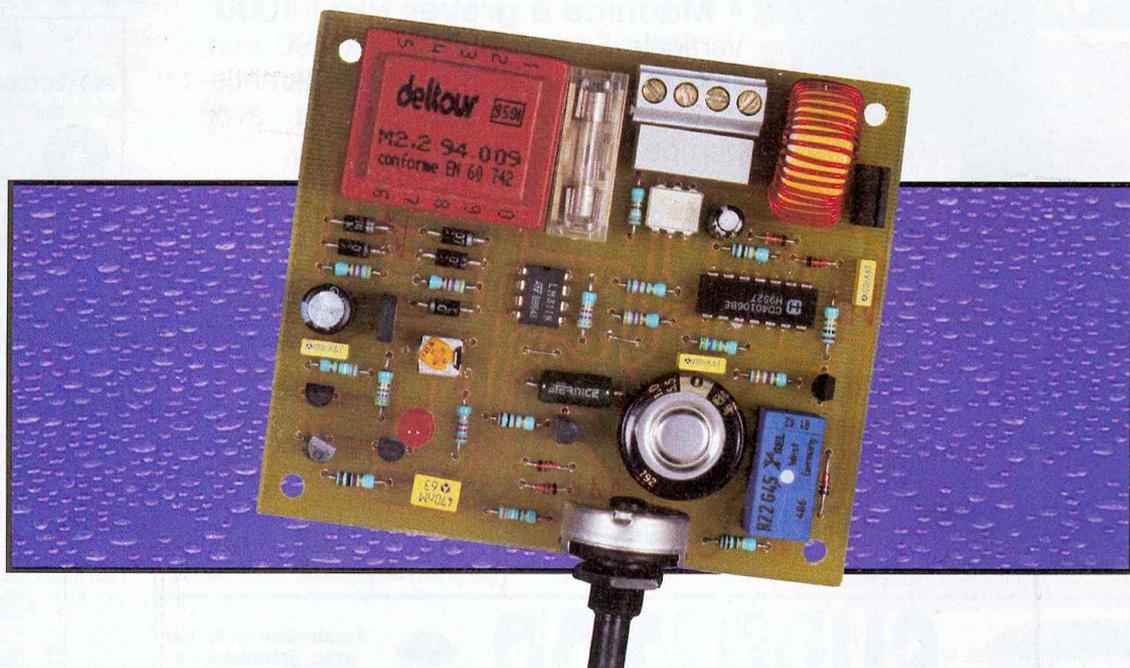
Suivez pas à pas les étapes numérotées et créez facilement des pages Web :

- Les différents logiciels disponibles, la création de titres et de listes, l'intégration d'un son et d'une vidéo...
  - L'édition de texte : alignement, retrait, test d'une page, utilisation de Netscape® Composer...
  - L'insertion d'images : formats GIF et JPG, palette Netscape®, image GIF transparente et animée, image d'arrière-plan...
  - Les tableaux : propriétés d'une cellule, fusion et fraction des cellules, insertion d'images dans les tableaux...
  - Les différents types de liens hypertextes, la création d'un lien à partir d'une image, la procédure de test...
  - La création de frames, l'installation d'un compteur, la mise en place d'un livre d'or...
  - Les fournisseurs Internet, la publication des pages Web, le référencement de votre page...
- Aidez-vous des nombreuses copies d'écran pour effectuer toutes les manipulations très simplement.
- Réalisez des pages Web spectaculaires grâce aux infos complémentaires.

R.WERLE - MICRO Application  
228 pages - 72 F.

# Veilleuse

pour enfant



Heureux les parents qui ne sont pas ou n'ont pas été confrontés aux caprices d'un enfant, qui ne veut pas s'endormir sans lumière ! Cette accoutumance pousse même certains enfants à se réveiller si vous avez pris soin d'éteindre la lumière après qu'ils se soient endormis.

Le principe de fonctionnement de cette veilleuse pour enfant est d'habituer progressivement l'enfant à l'obscurité en diminuant progressivement l'éclairage de sa lampe de chevet.

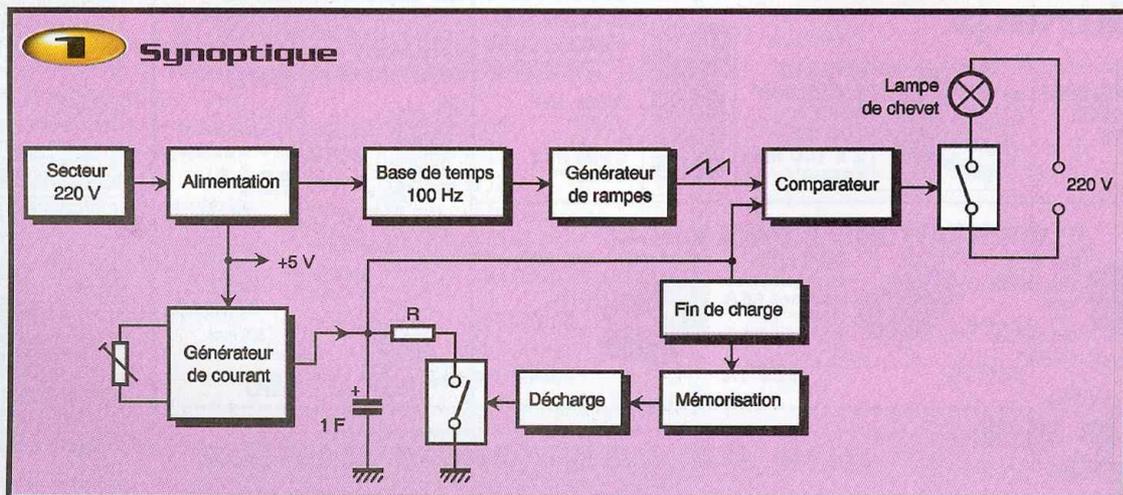
La luminosité de la chambre s'estompera lentement pour s'éteindre au bout d'une durée comprise approximativement entre une vingtaine de minutes et 5 à 7 heures, durée que vous pourrez réduire de semaine en semaine en fonction des progrès de l'enfant.

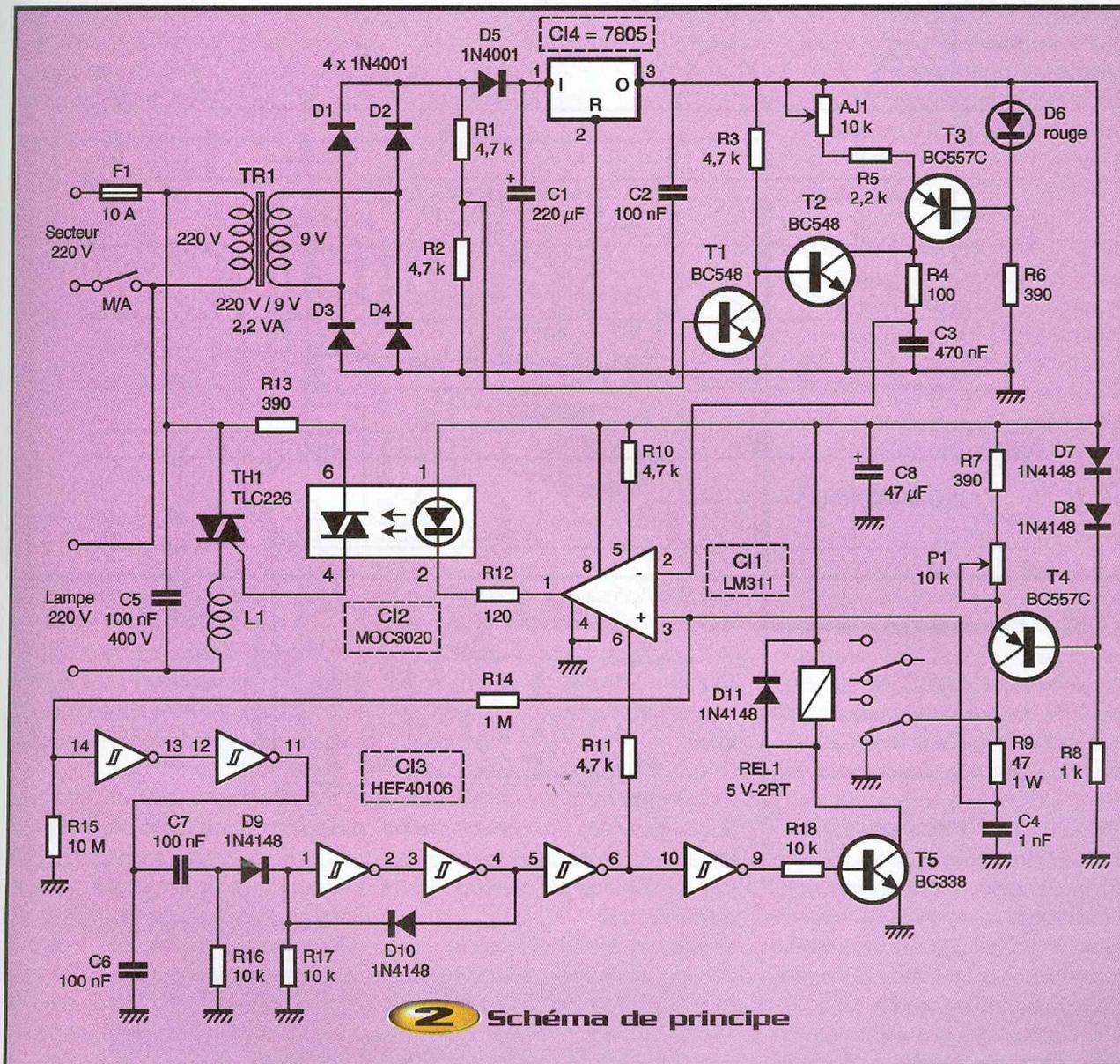
## Le schéma

Le synoptique de la **figure 1** donne la structure de cette veilleuse pour enfant dont la **figure 2** présente le schéma de principe. La gradation de lumière est obtenue grâce à un générateur de rampes de 100 Hz synchronisé sur la fréquence du secteur. La charge très lente d'un condensateur de très forte valeur permet une extinction progressive de la lumière pendant une durée réglable, couvrant une large plage de plusieurs heures.

## L'alimentation

Un transformateur de faible puissance abaisse la tension secteur et assure un isolement galvanique entre le réseau et l'alimentation basse tension du montage. La tension alternative 50 Hz/9Veff du secondaire est redressée par un pont de Graetz constitué des quatre diodes  $D_1$  à  $D_4$  de type 1N400x (si votre transformateur a deux enroulements secondaires, deux diodes seulement suffisent à un redressement double alternance). Dans un



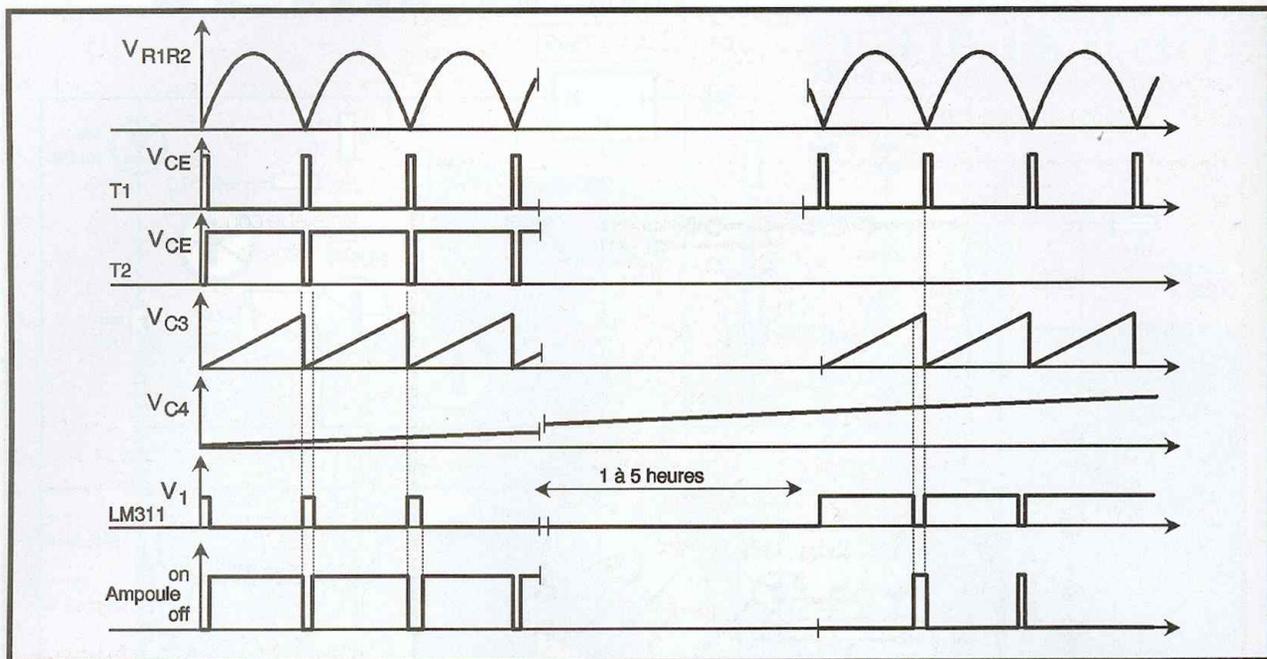


premier temps, ce pont est chargé par le diviseur de tension  $R_1/R_2$ . Par ailleurs, la diode  $D_5$  charge le condensateur de lissage  $C_1$ . La tension continue qui en résulte est ensuite régulée par un classique 7805 stabilisé par le condensateur de découplage  $C_2$ .

### La gradation de lumière

Le pont de résistances  $R_1/R_2$  polarise le transistor  $T_1$  dont le collecteur est chargé par la résistance  $R_3$ . Ce transistor est ainsi conducteur pendant la quasi-totalité d'une alternance secteur, bloquant la jonction base-émetteur de  $T_2$  qui est ainsi bloqué. Par contre, à chaque transition d'alternance secteur, la tension redressée passe par ce que l'on appelle couramment le zéro sec-





### 3 Chronogrammes

teur. La base du transistor  $T_1$  étant à cet instant à 0V, ce dernier est bloqué. La base de  $T_2$  est alors libérée et la résistance  $R_3$  sature le transistor  $T_2$  qui devient conducteur.

Pendant la durée du zéro secteur, la saturation du transistor  $T_2$  entraîne la décharge du condensateur  $C_3$  dans la résistance  $R_4$ . La constante de temps  $R_4/C_3$  étant très faible par rapport à une alternance secteur, la décharge du condensateur  $C_3$  est quasi instantanée. Cette phase correspond au front raide descendant d'une dent de scie. Par contre, dès que le transistor  $T_2$  est blo-

qué, le condensateur  $C_3$  commence à se charger à courant constant car le transistor  $T_3$  est alors polarisé en générateur de courant. L'évolution de la tension aux bornes du condensateur  $C_3$  est donc linéaire.

En effet, le potentiel de la base du transistor  $T_3$  est fixé par la tension directe de la LED  $D_6$ . La tension aux bornes de la résistance  $R_5+A_1$ , étant ainsi constante, le courant de collecteur du transistor  $T_3$  est constant ( $I_c \# I_E$ ).

Le signal en dents de scie résultant de la charge à courant constant du condensateur  $C_3$  (figure 3) est comparé à un seuil, fonction du niveau de charge du condensateur  $C_4$ . Ce condensateur de très forte

valeur (1 Farad, soit l'équivalent de 1000 condensateurs de 1000  $\mu F$  !) est également chargé à courant constant réglable à l'aide du potentiomètre  $P_1$ . Le signal en dents de scie est donc comparé par  $Cl_2$  à un seuil de tension qui augmentera très lentement.

Le comparateur  $Cl_2$  est un LM311 dont l'étage de sortie est constitué d'un transistor de puissance à collecteur ouvert. Ainsi, tant que la dent de scie appliquée sur l'entrée 3 de  $Cl_2$  n'a pas atteint le seuil déterminé par le niveau de charge du condensateur  $C_4$ , le triac  $Q_1$  n'est pas conducteur et aucune tension est appliquée à la lampe. Par contre, dès que la dent de scie dépasse le seuil imposé par  $C_4$ , le triac est amorcé et restera conducteur pour la durée restante de l'alternance secteur. Plus le condensateur  $C_4$  se charge et plus le triac est amorcé tardivement par rapport au début d'une alternance secteur. La lampe est ainsi de moins en moins brillante.  $Cl_8$  est un optotriac assurant l'amorçage du triac et l'isolation galvanique entre le secteur 220V et le circuit de commande basse tension.

#### L'extinction de la lampe

La fin de charge du condensateur  $C_4$ , qui correspond à l'extinction complète de la lampe, est détectée par un inverseur logique à trigger de Schmitt. Pendant la charge du condensateur  $C_4$ , la sortie 11 de  $Cl_3$  est à un état logique 0. Lorsque le seuil de commutation du trigger de



le CD40106 HARRIS et le condensateur C4 de 1F/5,5V

Schmitt de l'entrée 14 de  $C_3$  est atteint, la sortie 11 passe à 1 et une impulsion positive apparaît sur l'entrée 1 de la mémoire constituée de la diode  $D_9$ , de la

résistance  $R_{16}$  et des inverseurs 1/2 et 3/4 de  $C_3$ . Le condensateur  $C_6$  évite que la mémoire soit déclenchée pendant l'établissement de l'alimentation. Désormais

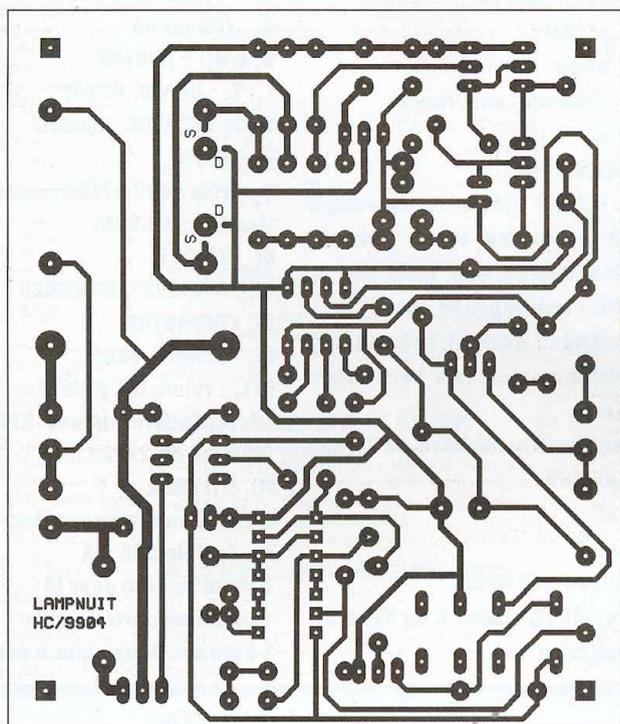
un état bas est présent en sortie 6 de  $C_3$ , ce qui a pour effet de bloquer le comparateur  $C_2$ , empêchant toute commutation du triac. Dans le même temps, le courant de base du transistor  $T_5$  est supprimé et celui-ci se bloque. Le relais n'est plus alimenté et le condensateur  $C_4$  se décharge alors dans la résistance  $R_9$  mise à la masse par le contact repos du relais. Le même processus de décharge se produira si vous coupez l'alimentation du montage alors que la lampe est toujours allumée. Il faudra alors attendre une bonne minute entre deux marches/arrêts de la veilleuse pour que le condensateur  $C_4$  soit totalement déchargé.

## La réalisation

Le tracé des pistes de la **figure 4** a été prévu pour l'utilisation d'un transformateur 2,2VA, aussi bien à simple qu'à double enroulement. Avec un transformateur à double enroulement, seules deux diodes suffisent au redressement, tandis qu'un pont de Graetz de quatre diodes est nécessaire si l'enroulement secondaire est unique. Le choix entre l'un ou l'autre des deux modèles s'effectue par des ponts de soudures adéquates sur le circuit imprimé, au niveau des deux pastilles centrales destinées au secondaire du transformateur. C'est ainsi que pour un transformateur à double enroulement, vous soudez les pistes repérées par un "D" aux pattes du secondaire. Dans le cas d'un simple enroulement, les pistes repérées par un "S" seront soudées aux pattes du transformateur. L'implantation des composants de la **figure 5** montre en pointillés les deux diodes inutiles en cas d'un double enroulement secondaire.

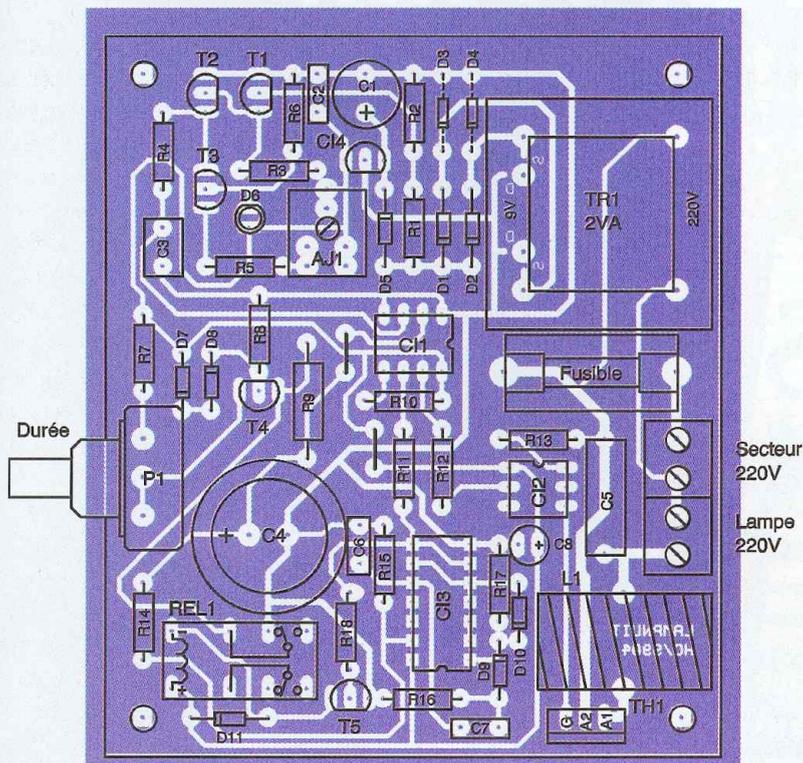
Le triac doit être un modèle à courant de maintien très faible car la lampe de chevet est normalement équipée d'une ampoule de faible puissance, généralement de 40 W. Ce sont généralement des modèles sensibles de faible courant (2 à 4A). Préférez aussi un modèle isolé et munissez le porte fusible d'un cache plastique de protection.

Positionnez à mi-course le curseur de la résistance ajustable  $A_1$ . Dès la mise sous tension, la lampe doit s'allumer. Le seul réglage consiste à régler la pente des dents de scie, soit à l'aide d'un multimètre soit



**4** Tracé du circuit imprimé

## 5 Implantation des éléments



d'un oscilloscope. Avec un multimètre numérique sur un calibre 20VDC,  $A_j$  est réglée pour lire une tension de 1,8V. Avec un oscilloscope, réglez  $A_j$  pour que la pointe des dents de scie soit à peine écrêtée, l'amplitude des rampes est alors proche de 3,5V.

Le potentiomètre  $P_1$  est tourné sur la gauche pour diminuer le temps nécessaire à l'extinction de la lampe et sur la droite pour obtenir la progression la plus lente. Selon la fabrication du 40106, son seuil de commutation peut varier de 2 à 3,6V. Si vous jugez que la durée maximale obtenue avec votre maquette est insuffisante, diminuez la valeur de la résistance  $R_{15}$ . Le montage photographié montre l'utilisation d'un 40106 de chez HARRIS (nouvellement TEXAS Instrument) dont le seuil de commutation a été élevé en diminuant la valeur de  $R_{15}$  à 5,6 M $\Omega$ . L'extinction de la lampe se produit ainsi après plus de 7 heures !

H. CADINOT

## Nomenclature

$R_1$ à $R_3$ : 4,7 k $\Omega$ (jaune, violet, rouge)	$C_6, C_7$ : 100 nF
$R_4$ : 100 $\Omega$ (marron, noir, marron)	$C_8$ : 47 $\mu$ F/10V
$R_5$ : 2,2 k $\Omega$ (rouge, rouge, rouge)	$D_1$ à $D_5$ : 1N4001, 4007
$R_6, R_7, R_{13}$ : 390 $\Omega$	$D_6$ : LED rouge
(orange, blanc, marron)	$D_7$ à $D_{11}$ : 1N4148
$R_8$ : 1 k $\Omega$ (marron, noir, rouge)	$T_1, T_2$ : BC548, BC547
$R_9$ : 22 à 47 $\Omega$ 1W	$T_3, T_4$ : BC558C, BC557C
(jaune, violet, noir)	$T_5$ : BC338
$R_{10}, R_{11}$ : 4,7 k $\Omega$ (jaune, violet, rouge)	$Q_1$ : triac 400V à faible courant de maintien (ex : TLC226)
$R_{12}$ : 120 $\Omega$ (marron, rouge, marron)	$CI_1$ : LM311
$R_{14}$ : 1 M $\Omega$ (marron, noir, vert)	$CI_2$ : MOC3021, MOC3020
$R_{15}$ : 10 M $\Omega$ (marron, noir, bleu)	$CI_3$ : HCF40106
$R_{16}$ à $R_{18}$ : 10 k $\Omega$ (marron, noir, orange)	$CI_4$ : 78M05, 7805
$A_j$ : résistance ajustable horizontale de 10 k $\Omega$	REL $_1$ : relais DIL 5V/2RT
$P_1$ : potentiomètre linéaire de 10 k $\Omega$	TR $_1$ : transformateur 220V/9V/2,2VA (250 mA ou 2x125 mA)
$C_1$ : 220 $\mu$ F/16V	TH $_1$ : TLC226
$C_2$ : 100 nF	L $_1$ : self antiparasites pour triac
$C_3$ : 470 nF	F $_1$ : fusible 500 mA
$C_4$ : 1 F/5,5V (MEGAMOS)	1 porte fusible pour CI
$C_5$ : 100 nF/400V, classe X ou X2, anti-parasitage triac	1 cache de porte fusible
	2 borniers deux plots à souder

# mieux vaut prévenir que guérir!

les dossiers  
le HAUT  
PARLEUR

Des solutions électroniques pour tous

## SPÉCIAL ALARME

sécurité - vidéosurveillance - domotique  
diffusion sonore dans l'habitat  
confort électronique dans la maison

Un numéro à ne pas manquer! en vente dès le 15 juin  
chez tous les marchands de journaux • 30<sup>F</sup>