



Mono-carte  
à PIC 17C756



OSCAR : lecteur  
de CD-MP3

- Cours  
« Electronique & PC »
- Systèmes Surround modernes
- Tout sur le Power Mac G4-500
- Digital Railrunner
- Testeur de câbles de réseau

## Essayez- vous à l'USB



# Au cœur du Power Mac G4 500

## Les secrets du dernier-né d'Apple

Guy Raedersdorf

Si vous êtes un lecteur assidu d'Elektor, les cartes-mères pour PC ne devraient plus, après les 3 articles que nous leur avons consacré, avoir le moindre secret (technique) pour vous. Il nous a paru intéressant, bien que nous ayons bien moins de lecteurs possédant un Apple que de lecteurs disposant d'un PC, de jeter un coup d'oeil au cœur d'un Power Mac G4 500, l'ordinateur le plus récent d'Apple, dont la puissance n'a rien à envier à ses concurrents de « l'autre bord », les PC (dits IBM compatibles).



Le Mac est l'ordinateur préféré de nombreux professionnels de l'édition. Cela ne devrait pas, avec l'arrivée du Power Mac G4 changer, bien au contraire, nous n'en voulons que pour prouver les regards d'envie qui ont accompagné l'arrivée de l'ordinateur et les mines de regret qui ont suivi son départ. Ce n'était pas encore le moment...

L'un des attraits majeurs des ordinateurs d'Apple est qu'ils ont un comportement prévisible, tous les G4 d'un moment donné et d'un type donné d'un même continent étant constitués des mêmes sous-ensembles, ce qui est lui d'être le cas d'un PC où l'un trouvera telle carte-mère, telle carte graphique, cette standardisation appliquée par Apple se traduisant inévitablement par une réactualisation (*updating*) plus facile à automatiser. Les nouveaux pilotes concernent tous les ordinateurs de la même famille et enveloppe chronologique, de sorte que la mise à niveau n'est jamais un problème.

Ouvrons la porte sur le sanctuaire.

### Le Power Mac G4, en quelques traits de crayon

Le synoptique de la **figure 1** permet de se faire une bonne idée sur l'agencement des différents sous-ensembles constituant ce (super-)ordinateur personnel.

Le Power Mac G4 se caractérise par la présence, non seulement d'un processeur G4 de Motorola, d'où son nom, mais également d'un processeur de vecteur baptisé Velocity Engine, qui dispose d'un set d'ins-

tructions de 162 instructions SIMD (*Single Instruction Multiple Data*). La largeur de la mémoire interne est de 128 bits. Le bus-système travaille à 100 MHz et atteint des taux de transfert de 400 Mbps. La carte du processeur comporte une antémémoire de niveau 2 de 1 Moctet (dite *back-side*) travaillant à la moitié de la fréquence d'horloge du processeur, soit 250 MHz pour le dernier modèle de G4.

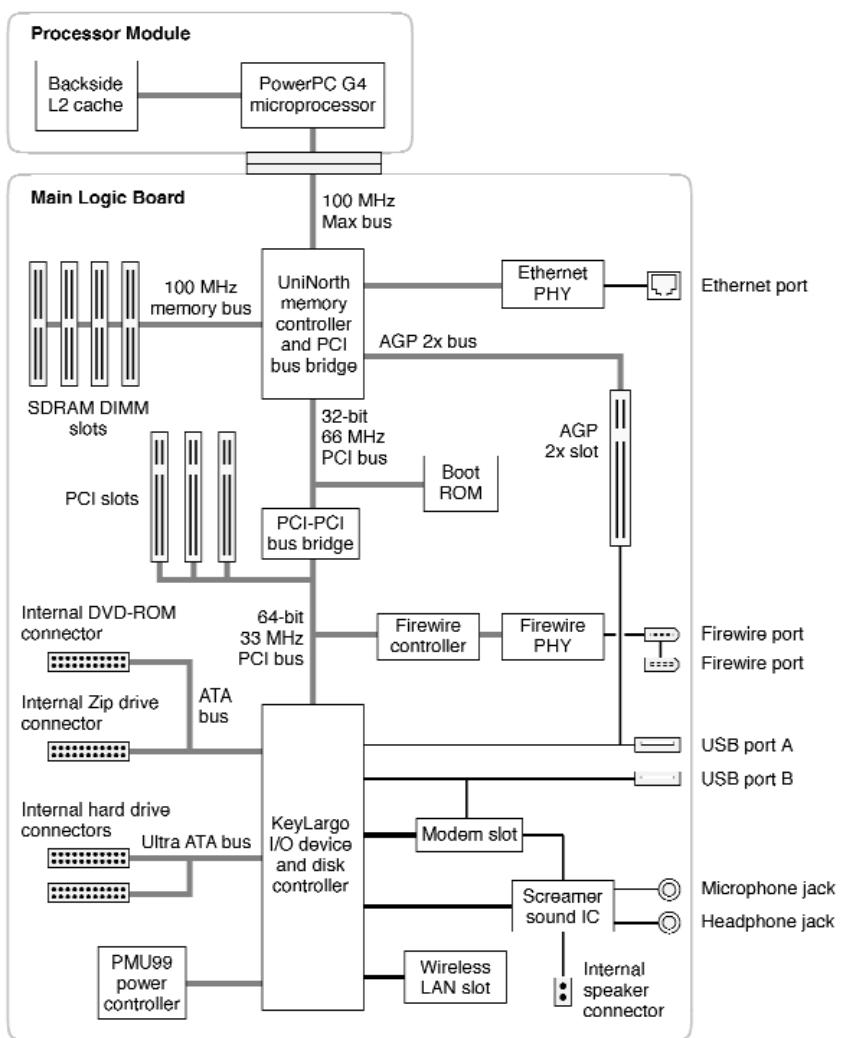
La carte-mère du Power Mac G4, baptisée Sawtooth (dent de scie) dispose de 4 supports de mémoire pouvant recevoir des modules DIMM à 168 contacts identiques à ceux utilisés sur le PC. Tous les modules à 64 bits de large, du 32 au 512 Mocets, en passant par 64, 128 et 256 Mocets peuvent être utilisés. La taille mémoire maximale théorique serait donc de 2 Goctets, mais le Système d'Exploitation (SE) Mac OS 9 travaille avec une taille maximum de mémoire de 1,5 Goctets. Une version future de Mac OS mettra à profit cette taille de 2 Goctets. D'ici là les modules de 1 Goctet pièce seront sans doute devenus monnaie plus courante. La taille de mémoire minimum est de 64 Mocets.

Le bus système : données sur 64 et adresses sur 32 bits, horloge de 100 MHz, supportant le protocole MaxBus. Le Power PC G4 est livré avec 256 Mocets de RAM, de quoi pouvoir commencer à travailler, le besoin de mémoire supplémentaire se fera sans doute sentir dans un proche avenir.



**Les contacts avec le monde extérieur :**

- 3 ports FireWire 400 Mbps dont un interne,
- 2 ports USB (12 Mbps chacun)



000036 - 4 - 11

Figure 1. Synoptique élagué de la carte-mère du Power Mac G4. (Source Apple.com)

## En savoir plus sur le PowerPC G4

Le PowerPC G4, G4 en raccourci, produit par Motorola, est, comme le suggère son nom, un processeur de la 4<sup>ème</sup> génération. Il existe actuellement en version 400, 450 et 500 MHz. Il utilise un bus système en pipeline, baptisé MaxBus, au rendement supérieur à celui du PowerPC G3.

### Autres caractéristiques marquantes du G4 :

Il est nommé premier « super-ordinateur » sur une puce, car capable de 1 milliard d'opérations en virgule flottante par seconde (gigaflop) en mode soutenu, 3,6 Gflops théoriquement en pointe.

Il est doté d'un noyau PowerPC superscalaire.

Le Velocity Engine (VE = moteur de vitesse) qui l'épaule sert à accélérer l'exécution des tâches courantes du système. Ce dernier peut traiter les données par morceaux de 128 bits au lieu des 32 ou 64 bits des processeurs classiques. Il peut en outre exécuter 4 (voire 8 dans certains cas) calculs 32 bits en virgule flottante par cycle (2 à 4 fois plus que les processeurs traditionnels). Il est prétendu, après tests comparatifs, que le G4 avec VE est jusqu'à 3 fois plus rapide qu'un Pentium III 800 MHz (!!).

Il utilise la technologie AltiVec : unité d'exécution vectorielle de 128 bits de large,

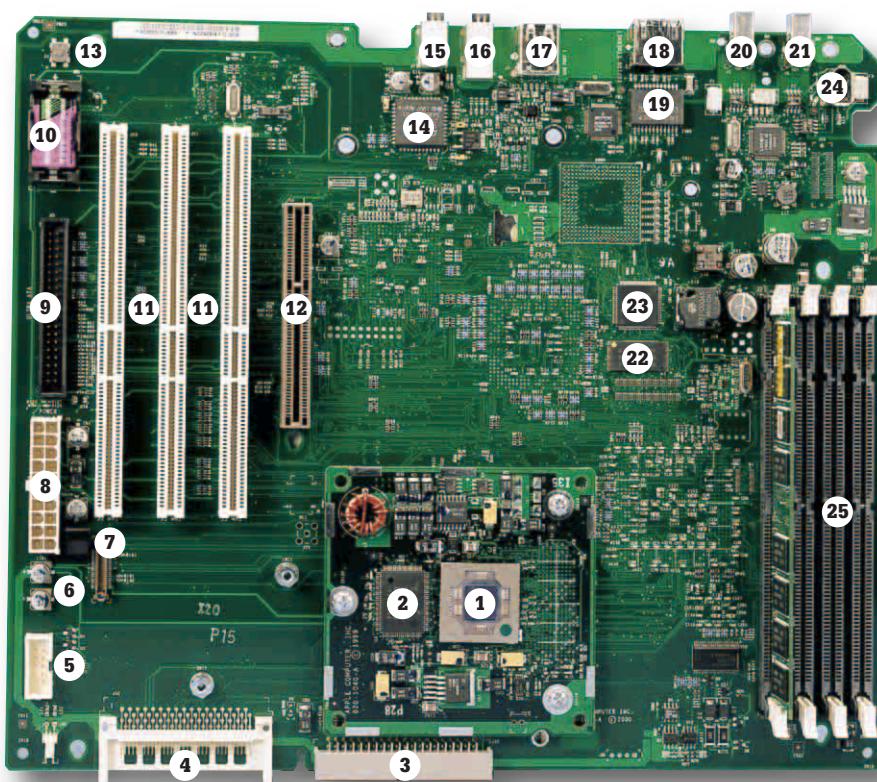
Possède un double cache de données et d'instructions de 32 Koctets

Il s'accorde avec jusqu'à 2 Mocets d'antémémoire L2 (backside)

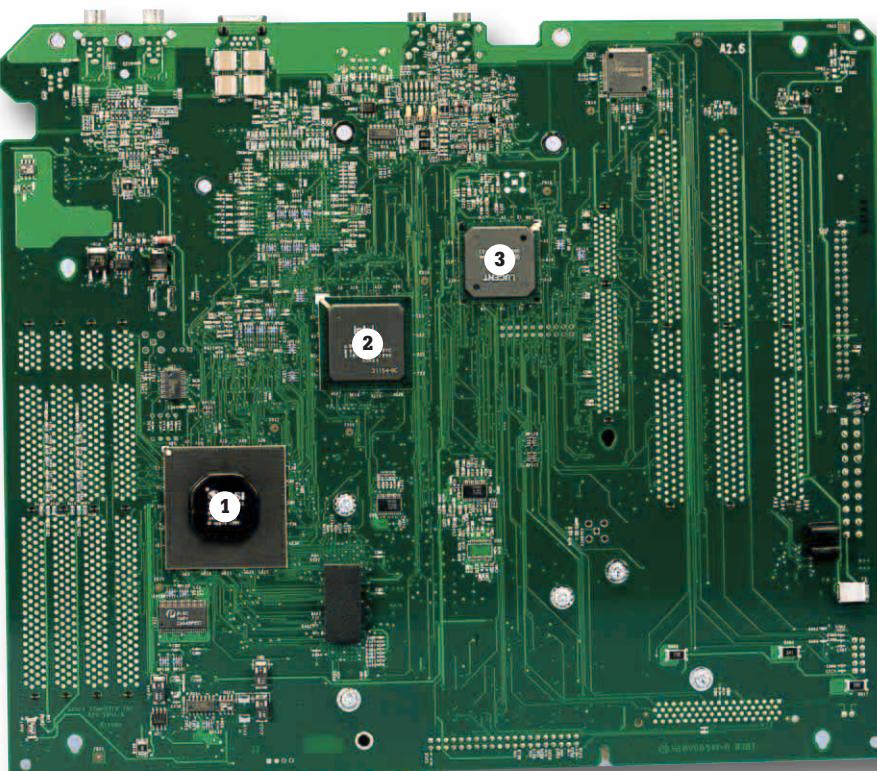
Une dernière caractéristique intéressante, la possibilité de démarrer le système depuis un périphérique de stockage connecté au réseau FireWire, voire au port USB si le périphérique est prévu pour cela.

# INFORMATIONS

## Le recto



## Sur le verso



### Connecteurs d'extension :

- 3 PCI 64 bits 33 MHz
- 1 AGP 2x dans lequel se trouve une carte graphique d'ATI du type RAGE 128 PRO dotée de 16 Moctets de SDRAM sur bus de 128 bits; elle possède et une embase mini-DB-9 à 15 contacts pour moniteur analogique (max 1 600 x 1 200 32 bits à 65 Hz, voire 1 920 x 1 080 à 16 bits de résolution) et une embase DVI à 24 contacts pour un moniteur numérique (tel que le superbe TFT 21" d'Apple, 1 600 x 1 024 à 32 bits max, qui à lui seul coûte encore, pour le moment, plus cher que l'ordinateur lui-même).

### - 2 sorties numériques

#### *Communications*

- Connecteur Ethernet 10/100BASE-T en option Aiport Base Station incorporant un modem 56K et un port 10Base-T Ethernet
- À noter l'absence, sur le modèle 500, de modem en standard, l'accent étant mis sur l'accès à Internet par le biais d'un réseau Ethernet. Rien n'interdit bien entendu d'en (faire) monter un en option.

- Ethernet intégré avec connecteur RJ-45 à 8 contacts pour modes 10 ou 100 Mbps.

### - Option Wireless LAN

- 3 ports WireFire (IEEE-1394a) dont 1 interne
- 2 ports USB à 4 contacts indépendants qui remplacent les ports sériels traditionnels; le port B de l'un des ports USB est en liaison avec la paire de signaux USAB du connecteur AGP. De même, le port B du

## Vue d'aigle sur la carte-mère du G4

### Le recto

Commençons par le composant le plus important, la

- 1 carte à G4 sur laquelle on trouve également
- 2 Backside, antémémoire (mémoire-cache) de niveau 2 (1 Moctet)
- 3 connecteur ATA standard pour lecteurs ZIP
- 4 connecteur pour module Wireless LAN (Local Area Network)
- 5 sortie vers haut-parleur
- 6 Extern Sound

# Le côté technique des choses

## AirPort

Le Power Mac G4 peut recevoir une carte AirPort. Ce module Wireless LAN comporte un contrôleur d'accès média, un processeur de signal numérique et une section HF. Le module comporte un connecteur destiné à recevoir le câble vers les antennes qui ont été intégrées dans le boîtier du Power Mac G4.

Il s'agit d'un port de connexion ouvrant la porte sur Internet et permettant également un transfert, par les éthers, de fichiers de toutes sortes, et cela jusqu'à de l'ordre de 50 m. Les signaux radio émis par l'AirPort traversent les murs... AirPort transmet 11 Mbits/s, soit 10 fois plus que la plupart des réseaux domestiques. Il dispose également de ports destinés à une ligne téléphonique et à Ethernet 10/100 Base-T.

AirPort permet une interconnexion de plusieurs ordinateurs. Il devient possible ainsi, par exemple, de surfer simultanément à plusieurs sur Internet tout en cherchant son courrier électronique. La carte AirPort peut également être utilisée pour le partage d'une imprimante, l'échange de fichiers et comme nous le disions quelques lignes plus haut, l'accès à Internet avec tout ce que cela sous-entend.

C'est pour les portables qu'AirPort présente le plus grand intérêt; où que vous soyez chez vous, vous avez, par son biais, accès quasi-instantanément à Internet ou un autre ordinateur.

Travailler en AirPort implique la présence d'une carte AirPort dans l'ordinateur concerné et d'une station de base, dite AirPort Base, à la forme de « soucoupe volante » (épaisse) ou son équivalent logi-

ciel, AirPort Software Base Station, l'ordinateur concerné faisant alors office de Station AirPort.

## FireWire

Il s'agit d'une technologie d'entrées/sortie série à haute vitesse servant à la connexion de périphériques. Ce standard officiel est connu sous la dénomination de IEEE 1394. Citons, parmi les applications les plus parlantes de FireWire, les caméscopes vidéo qui peuvent ainsi transférer les informations numériques pour un traitement ultérieur sur l'ordinateur, montage vidéo, création de clips, etc.

En fait, FireWire, dont l'embase comporte 6 contacts, est l'USB du riche; ces 2 technologies offrent l'intérêt majeur de permettre une connexion et une déconnexion à chaud (sans avoir à couper l'ordinateur); point n'est besoin d'identificateur de périphérique (les fameux ID des appareils SCSI) ni de terminateurs, des simples câbles suffisent, le système se configurant automatiquement. Si USB peut, théoriquement, piloter une chaîne comportant un maximum de 127 périphériques, FireWire se limite à 63. Le nombre de types de périphériques différents connectables à FireWire ne cesse de croître, caméscopes, synthétiseurs, disques durs, Jukebox DVD, graveurs de CD,

Comme l'indique son nom, il n'y a rien, actuellement, de plus « hot » que FireWire.

Que reste-t-il à signaler d'intéressant.

La version la plus récente du système d'exploitation du G4, le Mac OS 9, dispose d'une fonction de réactualisation (update) automatique.

second port USB est relié au connecteur destiné à recevoir un éventuel modem. Le clavier et la souris sont connectés aux ports USB.

## Périphériques

- Disque dur de 27 Goctets Ultra ATA 66
- Lecteur DVD-RAM avec lecture de DVD-RAM double face (5,2 Goctets max), de CD et de DVD-Vidéo
- Lecteur/enregistreur ZIP 100 Moc-tets (remplace le lecteur de dis-

quette considéré (à raison ???) comme dépassé).

Signalons, pour terminer, la présence de 2 baies libres pouvant recevoir des disques durs 3½ additionnels ou autres périphériques.

Nous osons espérer que la lecture de cet article consacré au dernier-né d'Apple permettra d'atténuer les ressentiments mutuels injustifiés entre les membres des 2 communautés du grand monde de la micro-informa-

tique, les possesseurs de PC (IBM) et les admirateurs inconditionnels d'un ordinateur au logo préféré d'Eve, la pomme (Apple).

*Nous remercions, en la personne de Michel Roelofs, la société Apple sise à Bunnik (NL) pour le matériel mis à notre disposition et son démontage.*

(000036-4)

*Une paire d'adresses intéressantes sur le Net :*  
<http://developer.apple.com/techpubs/>  
[http://www.apple.com/powermac/processor.html](http://www.apple.com/powermac/)

connecteur pour modem (modem en option)

18 Ethernet 10/100 Base-T

connecteur d'alimentation

19 Ethernet PHY (PHYsical Layer)

connecteur disque dur selon Ultra ATA 66

20 embase FireWire 2

0 pile au lithium (de grande taille)

21 embase FireWire 1

1 connecteurs PCI

22 Boot Rom. 1 Mocet d'EPROM Flash. En vue de réduire le nombre de broches de Uni-N, la ROM de boot est connectée au bus PCI 66 MHz.

2 connecteur AGP (vidéo)

23 contrôleur FireWire (Texas Instruments)

3 CUDA Reset, pour un redémarrage de la machine en cas de (très gros) problèmes

24 connecteur FireWire interne

4 Screamer, puce son de Crystal; il s'agit d'un codec (codeur/décodeur) audio 16 bits doté de 2 entrées et de 2 sorties stéréo analogiques.

25 connecteurs pour mémoire PC100

5 Sound Out

## Sur le verso

6 Sound In

1 UniNorth, une puce de VLSI; connue sous la dénomination de Uni-N, cette puce joue un rôle important dans le

Power Mac G4; elle assure des fonctions de pont entre le processeur, la mémoire, le système d'E/S à base de PCI, le connecteur AGP et l'interface Ethernet.

2 circuit des Entrées/Sorties (Intel) dit KeyLargo; il fournit toutes les fonctions d'entrées/sorties hormis Ethernet et FireWire; KeyLargo met à disposition 2 hubs USB indépendants, une interface Ultra DMA / 66, une interface EIDE (Extended IDE) et assure le support du connecteur de communication et du circuit-son. KeyLargo dispose d'un port série de type Macintosh traditionnel.

3 PCI-PCI bus bridge (Lucent). Le Power Mac G4 présente la caractéristique étonnante de disposer de 2 bus PCI ne travaillant pas à la même fréquence, le premier pris directement à la sortie du Uni-N, tournant à 66 MHz, le second, en aval du circuit PCI-PCI bus bridge qui porte ainsi bien son nom, tournant à 33 MHz.

# Orgues avec connexion à Internet

Peter Coenders

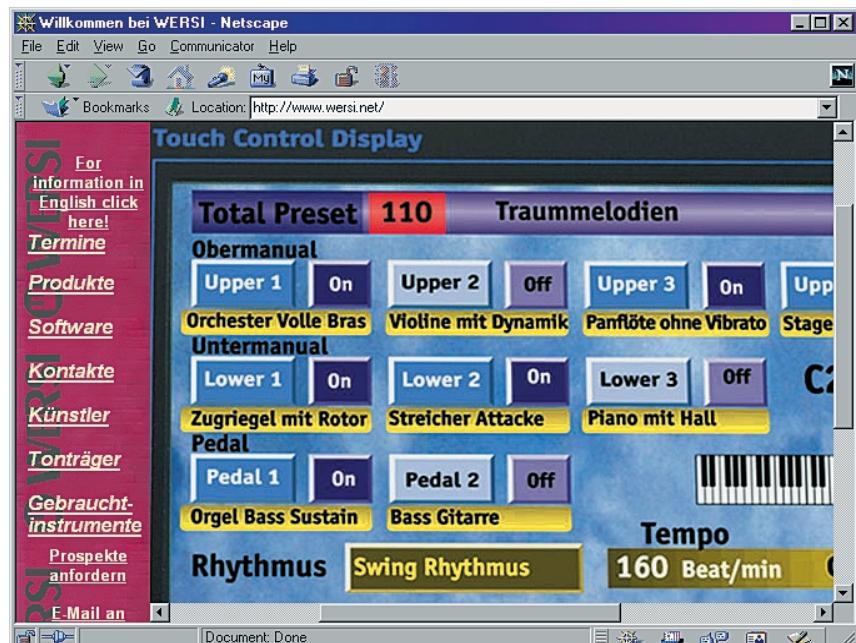
Aujourd’hui, même les orgues électroniques se retrouvent branchés sur Internet. Cela, dans bien des cas, en vue de la réactualisation des progiciels (update) du système, fichiers qu’il est possible de télécharger par le biais d’un PC. Wersi est allé plus loin encore. Leur nouvelle génération d’orgues est dotée d’une interconnexion à Internet autonome.

Wersi, une firme allemande, est l’un des fournisseurs d’orgues à réaliser soi-même les plus connus du marché. Pendant les années 70 et au début des années 80 en particulier, Wersi proposait des orgues analogiques très performantes et au prix relativement abordable. Des modèles tels que Helios et Galaxis se sont fait une renommée mondiale.

C'est aux alentours de 1985 que Wersi mit ses premières orgues numériques sur le marché. La miniaturisation sans cesse croissante de l'électronique fit blémir l'aspect « réalisation personnelle » de ce nouvelle génération. Au début des années 90, Wersi introduisit la série CD dont les Spectra et Atlantis furent les modèles haut de gamme. Cette série utilisait une technique d'échantillonnage sur 16 bits, garante d'une reproduction très naturelle des sons.

L'un des inconvénients de cette approche était une limitation de la polyphonie. Le modèle Spectra à 2 claviers et un pédalier disposait d'un maximum de 32 voix (rythmique exclue), l'Atlantis avec ses 3 claviers et son pédalier, de 56 voix maximum. Pour l'époque, ces caractéristiques techniques étaient loin d'être ridicules, mais elles ne faisaient pas le poids face aux orgues analogiques contemporaines.

Jusqu'à l'an dernier, Wersi a continué de développer le principe CD avec des instruments très attrayants tels que PhonX et Pegasus. Le grand écran LCD tactile dont sont dotés les instruments les plus récents, ainsi d'ailleurs qu'un certain nombre de claviers et



d'expanseurs, constitue une caractéristique pratiquement unique dans le monde des orgues. Cet écran associe un confort indiscutable tout en facilitant l'utilisation de ces appareils. Si on compare cette nouvelle génération d'instruments à des instruments haut de gamme tels que le Galaxis analogique, le nombre d'organes de réglages tels qu'interrupteurs, touches et autres drawbars a sensiblement diminué sans que la

palette des possibilités disponibles n'ait, elle, été réduite en conséquence, bien au contraire. Signalons au passage que Wersi s'est du même coup attaqué au problème de la polyphonie limitée. Les versions les plus récentes de Pegasus peuvent aller jusqu'à 64 voix polyphoniques. En début d'année, Wersi a présenté sa série d'instruments la plus récente. Il s'agit là d'une série très remarquable basée sur les technolo-



gies multimédia pour ordinateur les plus modernes et caractérisée par les connaissances des instruments propres à Wersi. Il semblerait à première vue que les limitations de la série CD aient été levées. Tous les modèles d'orgues et même le clavier sont dès lors polyphoniques à 208 voix. Ces instruments disposent

en outre d'un enregistrement en temps réel tel que le connaissent les instruments analogiques. La nouvelle série compte au nombre de ses membres Abacus (un *keyboard*), Vegas (un orgue portatif), Monet, Da Vinci (2 orgues de salon) et, le nec plus ultra, le Louvre doté lui de 3 claviers et d'un pédalier à 25 touches.

Cette série se distingue par son écran tactile couleur de grande taille. La caractéristique remarquable du clavier est qu'il dispose, comme c'est le cas des instruments de chez Hammond, de vrais *drawbars*. Signalons, en guise de conclusion, que Wersi propose, en option sur tous les instruments de sa nouvelle série, une connexion à Internet. Wersi veut ainsi offrir de nouvelles possibilités d'actualisation de sons et de rythmes, voire une sorte de service technique par une approche et médium différents. Il est vrai que des marques de renom telles que Yamaha et Roland ont opté pour une approche similaire, à ce près que les dites firmes proposent leurs mises à jour sur Internet où elles sont déchargeables à l'aide d'un PC pour ensuite être recopiées, par le biais d'une disquette, vers l'instrument concerné.

Bien que ces nouveaux instruments aient déjà été présentés en public à plusieurs reprises, ils ne sont pas encore disponibles. La production en série ne devrait cependant pas tarder.

Si vous voulez en savoir plus, voire écouter une avant-première (de Yamaha), rendez-vous sur Internet à l'une des adresses suivantes :

<http://www.wersi.net>

<http://www.yamaha.co.uk>

[http://www.yamaha.co.jp/english/product/pk\\_nonflash/products/psr9\\_000/psr9000.htm](http://www.yamaha.co.jp/english/product/pk_nonflash/products/psr9_000/psr9000.htm)

<http://www.rolandcorp.com/>

(005077)

## LE TORT D'ELEKTOR

### Chipsets MP3

**Elektor mai 2000,  
page 62 et suivantes**

Dans la figure 6, le K517C4000 est le microprocesseur chargé de la gestion du système et de la transmission des données. Le décodeur, qui est d'ailleurs également un convertisseur A/N, est le TL7231 (sa dénomination complète est DS TL7231MD comme le dit la légende, le DA3550 ayant été mentionné par erreur. Dans la figure 7, le microprocesseur n'est pas un PIC, mais un composant de la famille ST7 de SGS Microelectronics.

### Adaptateur d'EEPROM

**Elektor juillet/août 2000, page 98**

Comme nous l'on signalé plusieurs lecteurs, le schéma de l'adaptateur d'EEPROM comporte une double erreur. C'est la broche 14 de ICI (et non sa broche 15) qui doit être reliée à la patte supérieure de RI. Cette ligne 14/ICI – RI doit également aller à la broche 8 de

IC2. Il faut également corriger la platine sur ces points. Le dessin de platine modifié a été placé sur le site Internet d'Elektor à l'adresse : [www.elektor.presse.fr](http://www.elektor.presse.fr).

### Clavier volubile

**Elektor Avril/Mai 2000,  
pages 60/70 et suivantes**

La liste des composants donne un numéro de nomenclature erroné en ce qui concerne les capuchons pour les boutons-poussoirs. Le bon numéro est 707023-01 et non pas 700860-55.

La photo en début de second article comporte une erreur : le quartz a été monté à l'endroit où l'on devrait trouver en fait les connexions du résonateur.

Il semblerait qu'il soit difficile de mettre la main sur un résonateur céramique de 8 MHz. On peut fort bien le remplacer par un quartz qui viendra se connecter aux 2 contacts extérieurs des points prévus pour la connexion du résonateur.

Il faudra alors placer 2 condensateurs de

27 pF côté « pistes » pris respectivement entre le point central et chacun des contacts extérieurs.

### Super décodeur de loco pour EDITS Pro

**Elektor octobre 1999, page 34 et suivantes**

Le schéma de l'auxiliaire de programmation servant à programmer le décodeur comporte 2 transistors baptisés T2. L'un d'entre eux s'appelle bien évidemment T3. Les 2 transistors en question ne sont pas des BD560, mais des BD680. Il faut en outre modifier la valeur de C4 qui passe, de 1 nF à 2nF2.

### Mono-carte à 537 « Lite »

**Elektor janvier 2000, page 16 et suivantes**

L'auteur pense que les informations de cartographie mémoire données ci-après sont d'une importance capitale pour ceux d'entre nos lecteurs qui utilisent la mono-carte à 537 « Lite ». Les adresses de Chip Select sont les suivantes :

CS0 E000-E7FF

CS1 E800-EFFF

CS2 F000-F7FF

LCD F800-FFFF

## Chipsets MP3

**Elektor mai 2000,**

**page 62 et suivantes**

Dans la figure 6, le KS17C4000 est le microprocesseur chargé de la gestion du système et de la transmission des données. Le décodeur, qui est d'ailleurs également un convertisseur A/N, est le TL7231 (sa dénomination complète est DS TL7231MD comme le dit la légende, le DA3550 ayant été mentionné par erreur. Dans la figure 7, le microprocesseur n'est pas un PIC, mais un composant de la famille ST7 de SGS Microelectronics.

## Adaptateur d'EEPROM

**Elektor juillet/août 2000, page 98**

Comme nous l'on signalé plusieurs lecteurs, le schéma de l'adaptateur d'EEPROM comporte une double erreur. C'est la broche 14 de ICI (et non sa broche 15) qui doit être reliée à la patte supérieure de RI. Cette ligne 14/ICI – RI doit également aller à la broche 8 de

IC2. Il faut également corriger la platine sur ces points. Le dessin de platine modifié a été placé sur le site Internet d'Elektor à l'adresse : [www.elektor.presse.fr](http://www.elektor.presse.fr).

## Clavier volubile

**Elektor Avril/Mai 2000,**

**pages 60/70 et suivantes**

La liste des composants donne un numéro de nomenclature erroné en ce qui concerne les capuchons pour les boutons-poussoirs. Le bon numéro est 707023-01 et non pas 700860-55.

La photo en début de second article comporte une erreur : le quartz a été monté à l'endroit où l'on devrait trouver en fait les connexions du résonateur.

Il semblerait qu'il soit difficile de mettre la main sur un résonateur céramique de 8 MHz. On peut fort bien le remplacer par un quartz qui viendra se connecter aux 2 contacts extérieurs des points prévus pour la connexion du résonateur.

Il faudra alors placer 2 condensateurs de

27 pF côté « pistes » pris respectivement entre le point central et chacun des contacts extérieurs.

## Super décodeur de loco pour EDITS Pro

**Elektor octobre 1999, page 34 et suivantes**

Le schéma de l'auxiliaire de programmation servant à programmer le décodeur comporte 2 transistors baptisés T2. L'un d'entre eux s'appelle bien évidemment T3. Les 2 transistors en question ne sont pas des BD560, mais des BD680. Il faut en outre modifier la valeur de C4 qui passe, de 1 nF à 2nF2.

## Mono-carte à 537 « Lite »

**Elektor janvier 2000, page 16 et suivantes**

L'auteur pense que les informations de cartographie mémoire données ci-après sont d'une importance capitale pour ceux d'entre nos lecteurs qui utilisent la mono-carte à 537 « Lite ». Les adresses de Chip Select sont les suivantes :

CS0 E000-E7FF

CS1 E800-EFFF

CS2 F000-F7FF

LCD F800-FFFF

APPLIKATOR est une rubrique servant à la description de composants intéressants récents et de leurs applications; par conséquent, leur disponibilité n'est pas garantie. Le contenu de cette rubrique est basé sur les informations fournies par les fabricants et les importateurs, ne reposant pas nécessairement sur les expériences pratiques de la Rédaction.

# MCP120/I30

## Moniteur de microcontrôleur

**De nombreuses applications modernes à microcontrôleur ont besoin d'être protégées contre des tensions système trop faibles, sachant qu'elles peuvent se traduire par un comportement imprévisible du système à  $\mu$ C.**

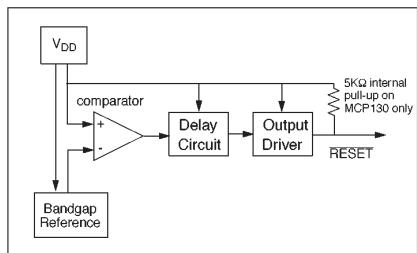


Figure 1. Structure interne du MCP1x0.

Les conséquences de l'application de niveaux de tension trop faibles vont de ce que l'on appelle le BOS (*Brown Out Syndrom*) processus au cours duquel on a une écriture involontaire de l'EEPROM (lors d'un effondrement bref de la tension d'alimentation) jusqu'au redémarrage du programme bien que le niveau de la tension d'alimentation présente ne soit pas suffisant pour garantir un fonc-

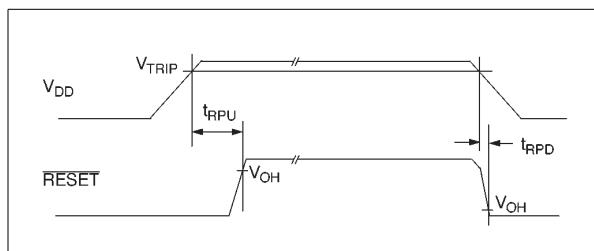


Figure 2. Chronologies cruciales : de la détection d'une sous-tension à la remise à zéro et le processus inverse.

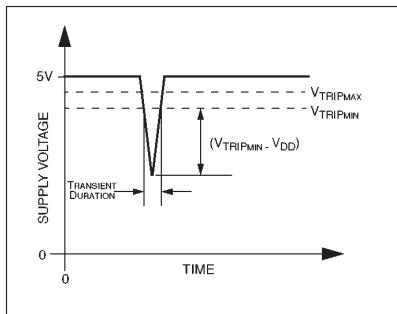


Figure 3. Des creux de tension rapides et peu accentués ne déclenchent pas de RAZ.

tionnement correct de la mémoire SRAM (RAM statique) du système au cas où il en serait doté (situation que l'on constate dans le cas d'un rétablissement trop progressif de la tension système).

C'est très précisément pour éviter que des effondrements de tension n'aient de conséquences incontrôlables sur le système à microcontrôleur que l'on fait appel à des circuits intégrés de surveillance (des moniteurs). Le MCP120/130 de Microchip Technology fait partie de cette famille de composants; sa fonction est de forcer le microcontrôleur dans un état de remise à zéro (RAZ) et de l'y maintenir jusqu'à ce que la tension système ait atteint un niveau suffisant et qu'elle soit bien stable. La **figure 1** montre la structure interne de ce composant. Le MCP120, tout comme son frère le MCP130 d'ailleurs, existe en 7 variantes caractérisées par la

valeur de la tension à laquelle chacun d'entre eux entre en fonction. Ces composants sont également présentés sous diverses formes de boîtiers et avec des brochages différents.

Quelque soit le modèle concerné, le circuit force la sortie de RAZ (*Reset*) au niveau bas (actif) dès lors que la tension système,  $V_{DD}$ , tombe en-deçà de la tension de consigne (*trip point voltage*). Quelle que soit la version de MCP120 concernée, sa sortie de RAZ est du type à collecteur ouvert (plus exactement à drain ouvert), alors que dans le cas du MCP130 cette sortie est dotée d'une résistance de forçage au niveau haut (*pull up*) de  $5\text{k}\Omega$  interne.

La **figure 2** nous propose le chronodiagramme du MCP120/130. Lors de la montée de la tension système, une fois que celle-ci a dépassé la valeur de tension de consigne  $V_{TRIP}$ , il faut au composant un certain temps,  $t_{RPU}$ , avant qu'il ne désactive la sor-

Identification composant : MCPI20/I30(T) - xxx b ml /pp				
Pièce	MCPI20:	Drain ouvert		
	MCPI20T:	Drain ouvert, fourni en bandes et rouleaux		
	MCPI30:	Drain ouvert & forçage au niveau haut		
	MCPI30T:	Drain ouvert & forçage au niveau haut, fourni en bandes et rouleaux		
Seuils xxx	270	$2,55 \leq V_{TRIP} \leq 2,70$		
	300	$2,85 \leq V_{TRIP} \leq 3,00$		
	315	$3,00 \leq V_{TRIP} \leq 3,15$		
	450	$4,25 \leq V_{TRIP} \leq 4,50$		
	460	$4,35 \leq V_{TRIP} \leq 4,60$		
	475	$4,50 \leq V_{TRIP} \leq 4,75$		
	485	$4,60 \leq V_{TRIP} \leq 4,85$		
	D	I: RST	2: $V_{DD}$	3: $V_{SS}$
Brochage (TO-92 uniquement, vu de face et du haut)	F	I: $V_{SS}$	2: RST	3: $V_{DD}$
	G	I: $V_{DD}$	2: $V_{SS}$	3: RST
Plage de températures I	H	I: $V_{DD}$	2: RST	3: $V_{SS}$
	I	$-40 \text{ à } +85^\circ\text{C}$		
Boîtier /pp	TO	TO-92, 3 broches	cf. brochage	
	TT	SOT-23, 3 broches	I: RST	$I: V_{DD}$
	SN	SOIC, 8 broches	I: RST	$2: V_{DD}$
			(toutes les autres broches restent non connectées)	

tie de RAZ et dans le sens inverse, un délai de  $t_{RPD}$  jusqu'à ce qu'il remette le microcontrôleur à zéro (réinitialisation). L'inertie du circuit intégré est sensiblement moindre, de l'ordre de 10  $\mu\text{s}$ , lorsqu'il lui faut désactiver le microcontrôleur que lorsqu'il décide, après remontée de la tension système à la valeur de service, la temporisation étant alors de l'ordre d'un 1/3 de seconde.

Un des paramètres importants dont il faut tenir compte lors de la configuration du réseau de monitoring, est la largeur d'impulsion minimale requise pour le déclenchement d'une remise à zéro. Nous voyons en **figure 3** une transitoire de tension typique en-deçà de la tension de consigne ( $V_{TRIP} - V_{DD}$ ) en fonction de la durée de la transitoire. On voit que des effondrements de tension « plats » doivent avoir une durée plus longue que des effondrements importants avant de se traduire par le déclenchement d'une RAZ. Il faudra partant parer les transitoires plates ne se traduisant pas par une RAZ à l'aide d'un condensateur de découplage (100 nF) placé le plus près possible de la broche d'alimentation positive ( $V_{DD}$ ).

(005057)

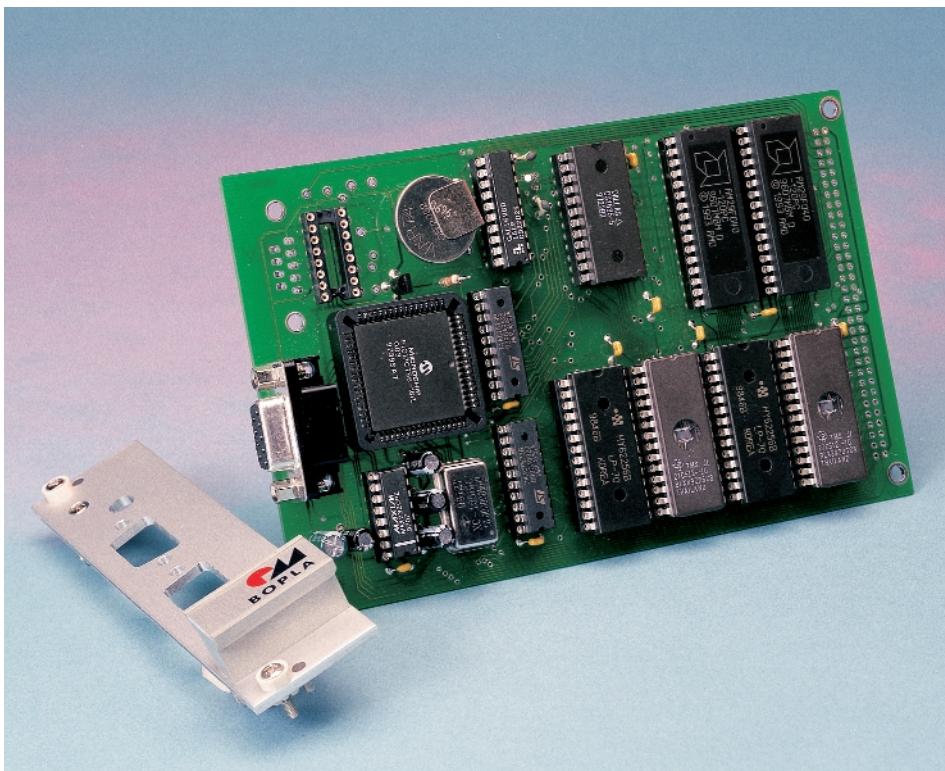
Données typiques ( $V_{DD} = 1,0 \text{ à } 5,5 \text{ V}$ , $T = -40 \text{ à } +85^\circ\text{C}$ , valeurs typiques pour $5,0 \text{ V}, 25^\circ\text{C}$ )						
Paramètre	Symbol	Conditions	Min.	Typ.	Max.	Unité
Tension d'alimentation	$V_{DD}$		1,0		5,5	V
Courant d'alimentation	$I_{DD}$	$V_{DD}=5,5 \text{ V}$ , sans charge		45	60	$\mu\text{A}$
Tension de consigne	$V_{TRIP}$	MCPIx0-270	2,55	2,625	2,70	V
		MCPIx0-300	2,85	2,925	3,00	
		MCPIx0-315	3,00	3,075	3,15	
		MCPIx0-450	4,25	4,375	4,50	
		MCPIx0-460	4,35	4,475	4,60	
		MCPIx0-475	4,50	4,625	4,75	
		MCPIx0-485	4,60	4,725	4,85	
		MCPIx0-...315, $I_{OL}=3,2 \text{ mA}$ , $V_{DD}=V_{TRIP,MIN}$			0,4	V
Tension de RAZ basse	$V_{OL}$	MCPIx0-450..., $I_{OL}=8,5 \text{ mA}$ , $V_{DD}=V_{TRIP,MIN}$			0,6	V
		MCPI30-xxx	$V_{DD}-0,7$			V
Tension de RAZ haute	$V_{OH}$	MCPI30-xxx		5		$k\Omega$
Résistance de forçage (pull-up)		MCPI30-xxx				
Courant de fuite en sortie		MCPI20-xxx		1		$\mu\text{A}$
Hystérésis de la tension de consigne	$V_{HYS}$			50		mV
Détection VDD pour RESET inactive	$t_{RPU}$		150	350	700	ms
Détection VDD pour RESET active	$t_{RPD}$	$V_{DD}$ de $V_{TRIPMAX}+250 \text{ mV}$ à $V_{TRIPMIN}-250 \text{ mV}$		10		$\mu\text{s}$
Valeurs maximales absolues						
$V_{DD}$		7,0 V				
Toutes entrées et sorties		$-0,6 \text{ V à } V_{DD}+1,0 \text{ V}$				
Température de service		$-65 \text{ à } +125^\circ\text{C}$				
Prtection DEM sur toutes les entrées		$\geq 2 \text{ kV}$				

# Mono-carte à PIC17

## Platine à microcontrôleur PIC17C756

Projet : Prof. Dr.-Ing. Francesco P. Volpe & Bernd Wegmann

La mono-carte à PIC17 au cœur de laquelle règne un microcontrôleur du type PIC17C756 en boîtier PLCC68 de chez Microchip Technology, est un système à la fois universel et extensible, qui, de plus, se laisse programmer depuis un PC. On peut ainsi se passer de programmeur spécifique.



Sur la plupart des systèmes à microcontrôleur se laissant programmer « en circuit », le programme Moniteur se trouve grillé dans une EPROM externe. Le contrôleur doit disposer d'une capacité d'occulter son domaine de

mémoire de programme interne et pouvoir être forcé de chercher le programme uniquement dans la mémoire externe. Il faut en outre que cette mémoire externe ait très exactement la même organisation que la mémoire interne (largeur de 16 bits dans le cas présent).

Le cœur et la périphérie

Le cœur de notre mono-carte à PIC17, dont le schéma représenté en **figure 1** montre qu'il s'agit d'un système à microcontrôleur classique, est un microcontrôleur de l'écurie Microchip Technology, un PIC17C756. Ce microcontrôleur connaît 58 instructions d'un mot (single word) qui ne requièrent, exception faite des instructions de saut et d'accès aux tableaux, qu'un seul cycle d'horloge.

Le PIC17C756 travaille avec une tension d'alimentation comprise entre 3,0 et 6,0 V, la plage de sa fréquence d'horloge s'étend elle du continu à 33 MHz. On procède à une division par 4 interne de la fréquence d'horloge de sorte que la durée d'un cycle est de 121 ns au minimum. Cette fréquence d'horloge implique l'utilisation de circuits intégrés de mémoire au temps d'accès inférieur à 120 ns ! Le noyau du microcontrôleur comporte un multiplieur 8 x 8 câblé (matériel), 18 sources d'interruption

## Caractéristiques techniques de la mono-carte à PIC17 :

- La CPU est un PIC17C756
- Comporte une EPROM avec programme Moniteur
- Le programme utilisateur est transféré en mémoire Flash sous la houlette du Moniteur
- Possède de la SRAM
- Horloge en temps réel et sauvegarde par pile
- Format carte Europe avec connecteur d'extension VG-64
- Compatible à 100% avec l'environnement de développement MPLAB de Microchip Technology
- Décodage d'adresse par GAL, flexible et modifiable.

et une pile matérielle comportant pas moins de 16 niveaux. Le PIC17C756 possède, en interne, une mémoire de données de 902 x 8 bits et une mémoire de programme de 16 K x 16 bits.

Cette dernière n'est pas nécessairement à utiliser sachant que le microcontrôleur peut travailler en 3 modes différents dont on retrouve les spécificités dans le tableau récapitulatif de la **figure 2**. Voici ces modes :

### *– Le mode microcontrôleur :*

La totalité de la mémoire de programme se trouve à bord du processeur.

**– Le mode microcontrôleur étendu :** La mémoire de programme située entre les adresses 0000<sub>H</sub> et 3FFF<sub>H</sub> se trouve sur le circuit intégré, le domaine de mémoire situé au-delà de 4000<sub>H</sub> jusqu'à FFFF<sub>H</sub> étant adressé en externe. À cet effet, les PortC et PortD servent de lignes d'adresses et de données multiplexées. La distinction se fait par le biais du signal ALE (*Address Latch Enable*) et de 2 verrous D (*latch*). On notera que le bus d'adresses et le bus de données possèdent une largeur de 16 bits.

### *– Le mode microprocesseur :*

L'ensemble de la mémoire de programme se trouve en externe; la taille maximale de la mémoire adressable est de 64 Koctets. La mémoire de programme propre du microcontrôleur reste inutilisée. L'accès à la mémoire externe se fait comme en mode microcontrôleur étendu à la différence près que l'adresse de la mémoire en externe se fait dès

l'adresse 0000<sub>H</sub>. Ce mode ne permet ni l'accès aux bits de configuration et à la ROM de boot ni le test de mémoire.

Dans le cas de notre mono-carte à PIC17, le PIC17C756 travaille en mode microprocesseur sachant que, de plus, il se trouve à l'état non programmé. C'est très exactement ce que nous voulons vu qu'il n'est pas dans nos intentions de programmer le microcontrôleur.

Enfoui dans son boîtier PLCC à 68 broches, le PIC17C756 possède nombre de périphériques. Il est possible ainsi d'avoir accès, individuellement, à 50 lignes d'entrées/sorties (I/O). Chacune de ces lignes est en mesure de drainer ou de fournir un courant maximum de 25 mA, cette intensité pouvant même aller jusqu'à 60 mA (sous 12 V) dans le cas des 2 lignes RA2 et RA3, ceci en vue de pouvoir piloter directement des LED. 4 des broches sont utilisables en entrée de capture avec une résolution d'un cycle (121 ns), 3 autres comme sorties MLI (**M**odulation en Largeur d'**I**mpulsion = PWM pour *Pulse Width Modulation* en anglais) avec une résolution de 1 à 10 bits.

Le PIC17C756 possède un temporisateur/compteur (Timer/Counter dit T/C) à 16 bits avec prédiviseur 8 bits programmable, deux autres T/C à 8 bits et un dernier T/C à 16 bits. Un générateur de taux de transmission autonome assure l'horloge requise par la communication qui pourra s'établir par le biais des 2 interfaces universelles synchrone/asynchrone (USART/SCI); on dispose en outre d'un port sériel synchrone (SSP) qui connaît différents modes (SPI et I<sup>2</sup>C,

utilisable également en maître I<sup>2</sup>C). Un convertisseur Analogique/Numérique (CAN) à 12 canaux et possédant une résolution de 10 bits permet une conversion directe de grandeurs analogiques.

Le PIC17C756 dispose en outre d'un temporisateur de mise en fonction (*Power-up Timer*), d'un temporisateur de lancement d'oscillateur et d'un circuit de remise à zéro à la mise sous tension (POR = *Power On Reset*), d'une remise à zéro en cas de niveau de tension d'alimentation trop faible (*Brown-out reset*), d'une protection du logiciel; il peut être mis dans un mode de sommeil très économique en énergie et connaît différentes options au niveau de l'oscillateur utilisable. Il ne faudra pas oublier que le temporisateur chien de garde (*WatchDog Timer*, WDT en raccourci) se trouve, par défaut, en mode actif. Sa désactivation n'est possible qu'à l'aide d'un programmeur et donc malheureusement pas par logiciel. Il faudra partant, lors de la programmation, toujours avoir le contrôle du WDT et le cas échéant le remettre à zéro pour éviter qu'il ne déclenche une remise à zéro imprudente. Cette exigence peut paraître, à première vue, ennuyeuse et demander une attention soutenue, mais, elle ne peut avoir qu'une influence favorable pour forcer à adopter un style de programmation correct et garantir un déroulement plus stable du programme.

2 autres circuits intégrés font partie de l'entourage immédiat du microcontrôleur, à savoir l'oscillateur IC15 placé tout près et un circuit de suivi de microcontrôleur, IC14, qui n'hésite pas à remettre le microcontrôleur à zéro lorsque la tension d'alimentation tombe à une valeur trop faible. Nous avons intercalé, sur la ligne de remise à zéro maître MCLR, un bouton-poussoir relié à la masse qui permet une remise à zéro manuelle de la mono-carte à PIC17. Le circuit intégré utilisé, un MCP120/130 est décrit ailleurs dans ce numéro. Notons que le MCP130 comporte une résistance de forçage au niveau haut (*pull up*) interne de sorte que l'on peut, si on utilise le dit type de circuit, supprimer la résistance R1.

## Cartographie mémoire

On retrouve en **figure 3** la cartographie de la mémoire. Comme nous le disions plus haut, en mode Microprocesseur, l'ensemble de la mémoire de programme allant de 0000<sub>H</sub> à FFFF<sub>H</sub> se trouve hors-composant (*off chip*). Cette figure illustre clairement la répartition de la mémoire externe. L'EPROM Moniteur se situe dans l'espace mémoire allant de 0000<sub>H</sub> à 1FFF<sub>H</sub> et occupe partant 8 Koctets. On utilise pour cela une paire d'EPROM du type 27C512 (le standard actuel, plus facile à trouver et

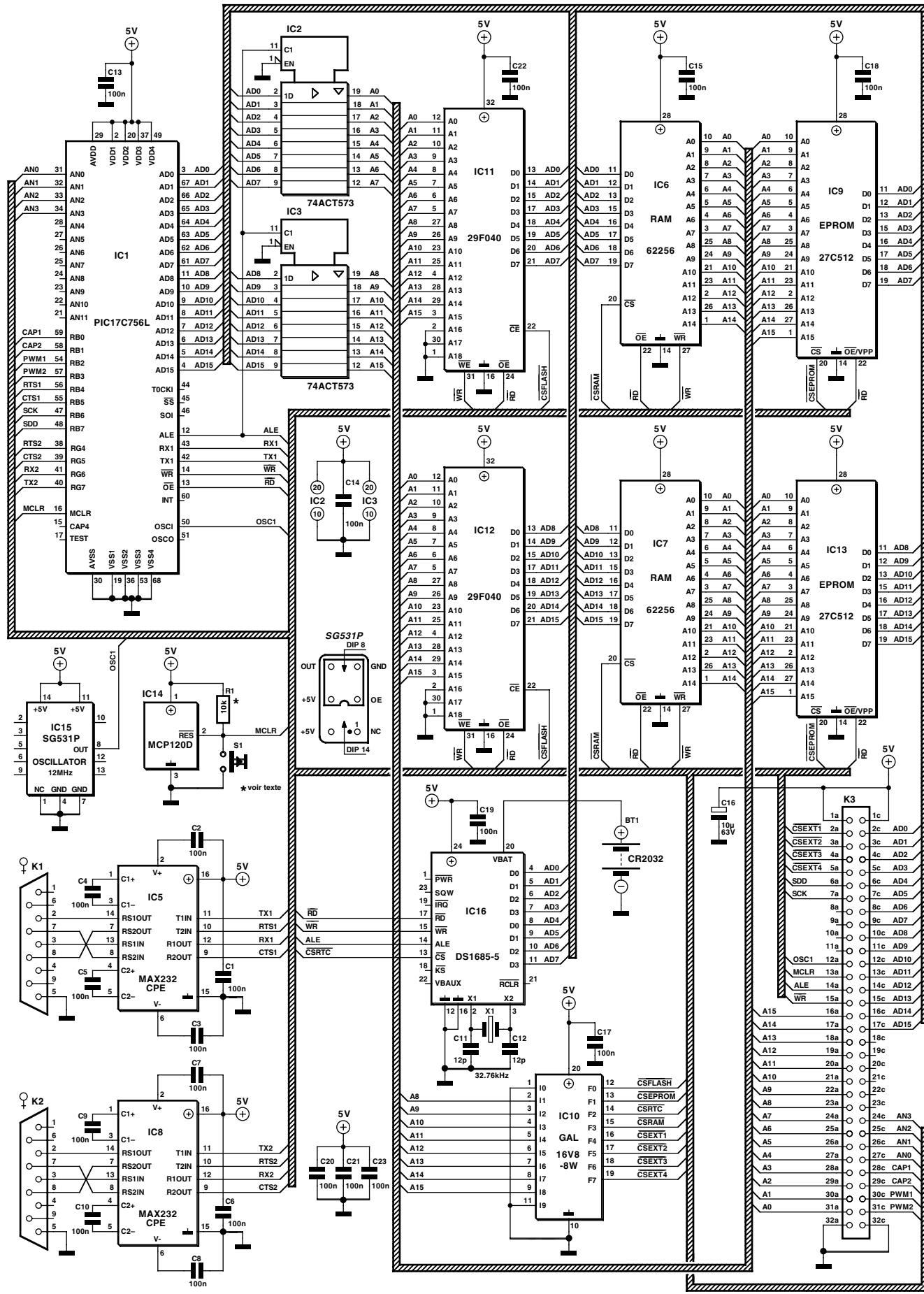


Figure 1. L'électronique de la mono-carte à PIC17.

pas, ou à peine, plus chère qu'une 27C64 et qui ne sera utilisée qu'en partie), IC9 et IC13, d'une largeur de 8 bits chacune, de sorte que le programme Moniteur est subdivisé en un octet Haut (*High*) et un octet Bas (*Low*). Ces 2 EPROM sont disponibles auprès des adresses habituelles et sont dénommées **EPS000061-21b** pour l'EPROM de poids fort (*High*) et **EPS000061-21a** pour celle de poids faible (*Low*). Il ne saurait bien évidemment être question d'inverser ces EPROM lors de leur mise en place sur la platine.

Les vecteurs d'interruption se trouvent, dans le domaine d'EPROM, aux adresses 0008, 0010, 0018 et 0020<sub>H</sub>. De manière à pouvoir utiliser ces interruptions dans le cadre d'un programme stocké en mémoire Flash, ces adresses du moniteur effectuent un saut, allant aux adresses 9F08, 9F10, 9F18 et 9F20<sub>H</sub> situées elles dans le domaine de mémoire Flash. Le programme utilisateur est placé dans les 2 circuits intégrés Flash, du type AM29F010 ou AM29F040. Ici encore on se trouve en présence d'une paire de circuits de mémoire de 8 bits de large. Le programme utilisateur dispose d'un domaine de 32 Koctets situé entre les adresses 2000 et 9FFF<sub>H</sub>. Cette taille devrait être suffisante. Il vous faudra, s'il vous faut plus d'espace utilisateur, reprogrammer la GAL d'adressage. Les circuits de mémoire Flash du type 29F040 ont une capacité de 4 Mbits, ce qui devrait constituer une réserve suffisante. Ceci est également le cas de la mémoire de données, qui est constituée par 2 SRAM du type 62256, IC6 et IC7, et que nous avons volontairement limitée à 12 Koctets (adresses A000<sub>H</sub> à CFFF<sub>H</sub>). Il existe une réserve de 20 Koctets mais il ne faut pas perdre de vue que l'espace adressable total est, dans tous les cas de figure, limité à 64 Koctets.

Le décodage des adresses est l'af-faire d'un circuit intégré programmable, une GAL du type 16V8D, dont le listage de la **figure 4** donne le contenu. À l'intérieur du domaine d'adresses allant de  $D000_H$  à  $EFFF_H$  la GAL décode 4 signaux, Extern1 à Extern4, chacun d'entre eux s'étant vu attribuer un domaine de 2 Koc-tets. Ces signaux sont disponibles sur un connecteur à 64 contacts et pourront ainsi servir à adresser

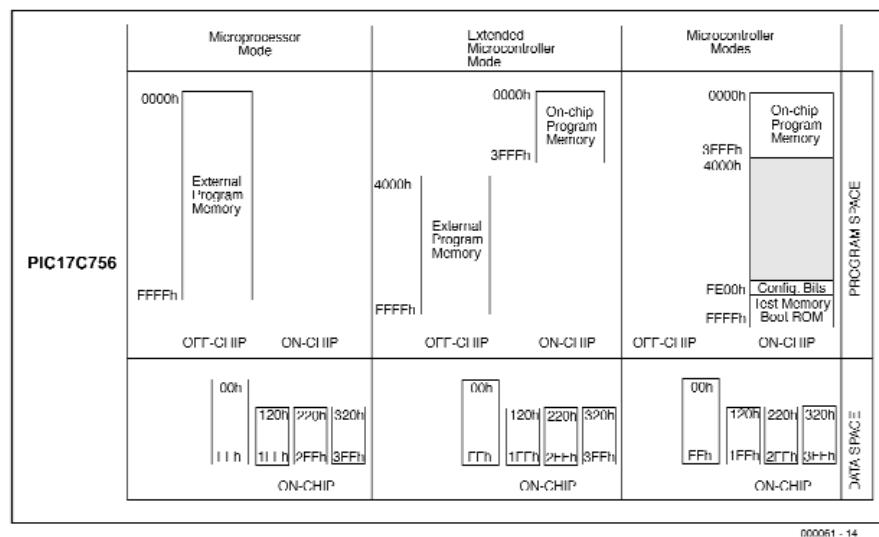


Figure 2. La cartographie mémoire change en fonction du mode de fonctionnement adopté.

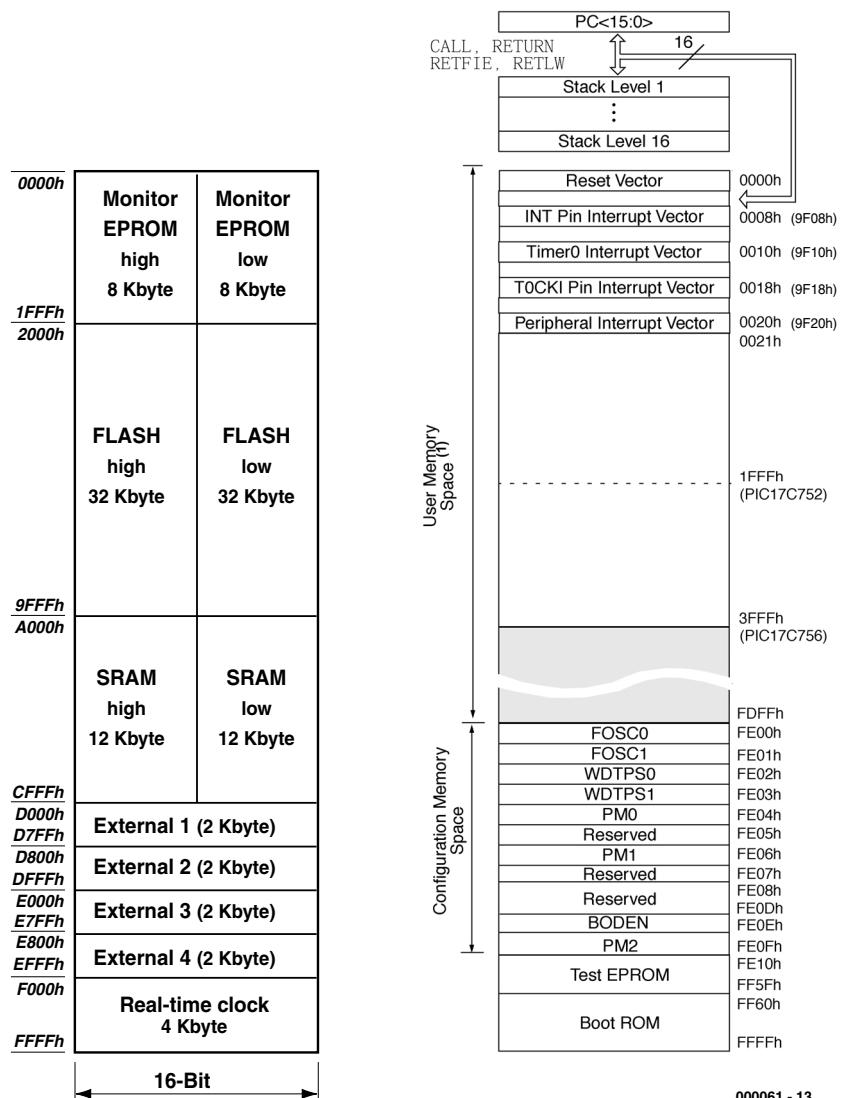


Figure 3. Cartographie de la mémoire du PIC17C756. Le Moniteur démarre à l'adresse  $1FFF_H$ .

```

1:Name        Mon;
2:Partno     IC10;
3:Date        08/11/99;
4:Rev         01;
5:Designer   Bernd Wegmann;
6:Company    FH-Aschaffenburg;
7:Assembly   None;
8:Location   None;
9:Device     g16v8;
10:
11:/*****************************************/
12:/*                                     */
13:/*****************************************/
14:
15:/** Inputs **/
16:
17:Pin [2..9] = [A8..A15];      /* Adressen */
18:
19:
20:/** Outputs **/
21:
22:Pin 12 = !Flash; /* EN von Tastenmatrix */
23:Pin 13 = !Eprom;
24:Pin 14 = !Uhr;
25:Pin 15 = !SRam;
26:Pin 16 = !Ext1;
27:Pin 17 = !Ext2;
28:Pin 18 = !Ext3;
29:Pin 19 = !Ext4;

```

Jedec Fuse Checksum (3373)  
Jedec Transmit Checksum (cffa)

Figure 4. Le listage du contenu de la GAL pourra être modifié facilement pour s'accommoder d'une mémoire de 4 Mbits.

d'éventuelles cartes d'extension. Un dernier domaine, allant de  $F000_H$  à  $FFFF_H$  est réservé à l'horloge en temps réel, IC16, une DS1685-5 de Dallas Semiconductor. Cette horloge dispose d'un tamponnage par pile. Notons que les condensateurs C11 et C12 ne sont pas indispensables et que l'on pourra, partant, omettre de les planter.

## Communication

La mono-carte à PIC17 dispose, pour la communication, de 2 interfaces RS-232 sérielles connectées chacune à l'un des USART (*Universal Synchronous/Asynchronous Receiver/Transmitter*) du microcontrôleur. Le pilotage et le téléchargement de programmes par le biais du PC se fait par l'intermédiaire de l'USART2 du microcontrôleur. La mono-carte à PIC17 communique dans ce cas-là avec le paramétrage suivant :

- 9 600 bauds
- 8 bits de donnée
- sans parité
- 1 bit d'arrêt
- écho local activé
- conversion, à la réception, d'un retour chariot (CR = *Carriage Return*) en un retour chariot + retour à la ligne (CR/LF = *CR/Line Feed*).

La première interface, USART1, peut alors

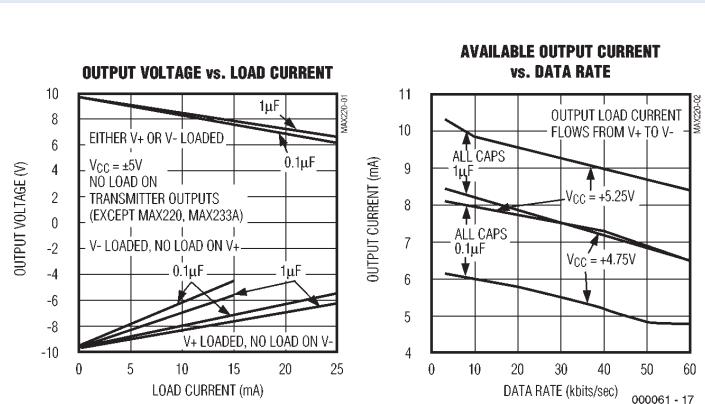
## MAX232

Le convertisseur d'interface MAX232 génère, du côté de l'interface RS-232, une tension dont le niveau dépasse celui de la tension d'alimentation, de même qu'une tension négative de même valeur. Pour ce faire, le MAX232 fait appel à 2 pompes de charge dont les condensateurs sont connectés aux points C1+/- et C2+/-.

Nous utilisons normalement, pour les dits condensateurs, des condensateurs d'une capacité de 1 mF, mais rien ne requiert une valeur aussi élevée, surtout si la consommation de courant du montage n'est pas très élevée.

Les courbes ci-dessous illustrent la relation entre le courant de charge, la tension de sortie et la capacité à donner aux condensateurs des pompes de charge dans le cas d'un MAX232A. Si vous n'avez pas une confiance inébranlable dans des condensateurs de faible valeur, rien n'interdit d'opter pour des condensateurs de 1  $\mu$ F pour un taux de transmission ne dépassant pas 120 Kbauds. La platine dispose de suffisamment de place pour leur implantation.

Le MAX232A CPE atteint, avec ces condensateurs, une vitesse de 200 Kbauds.



être utilisée tout normalement. La conversion des niveaux fait appel à une paire de circuits spécialisés dans l'adaptation de niveaux, à savoir des MAX232 (cf. l'encadré).

Une fois que le programme a été transféré, par le biais de l'interface USART2, dans la mémoire Flash de la mono-carte, cette dernière peut travailler en mode autonome. À par-

tir de cet instant, l'USART2 est mis à la disposition de l'utilisateur.

La mise en place des composants sur la platine double face à trous métallisés, dont on retrouve le dessin des pistes et la sérigraphie

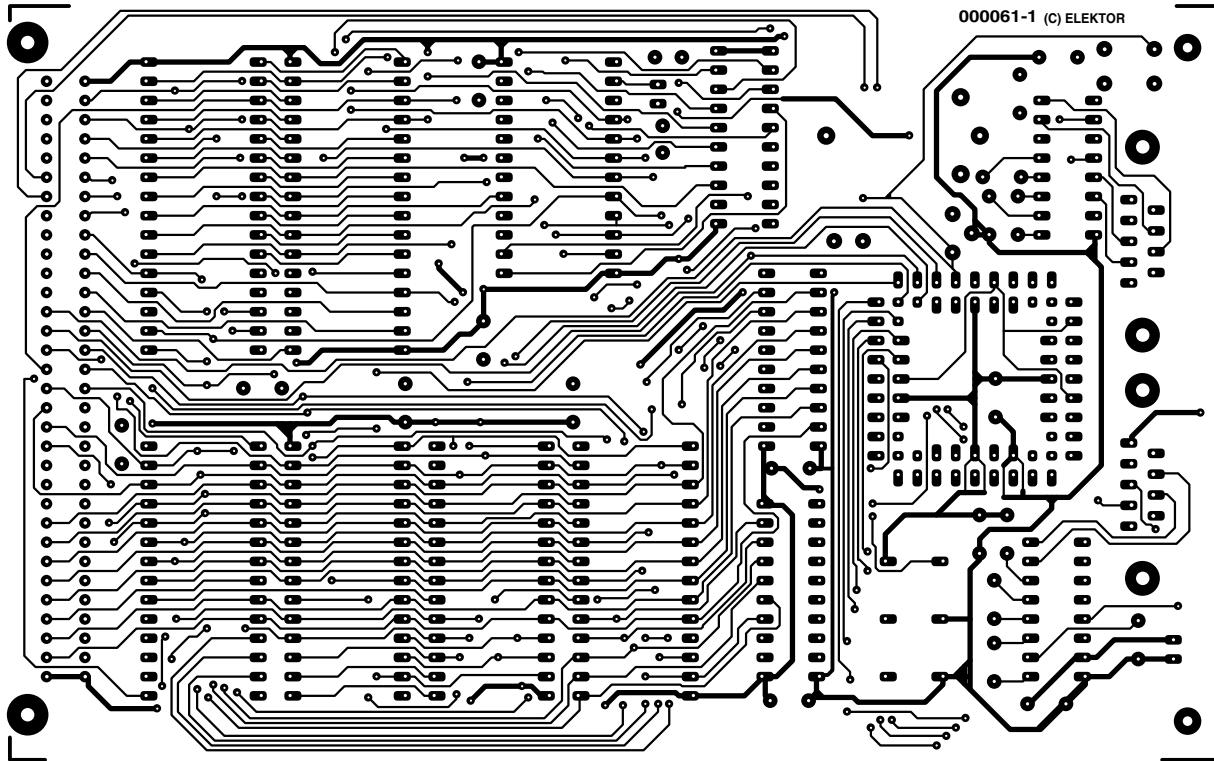
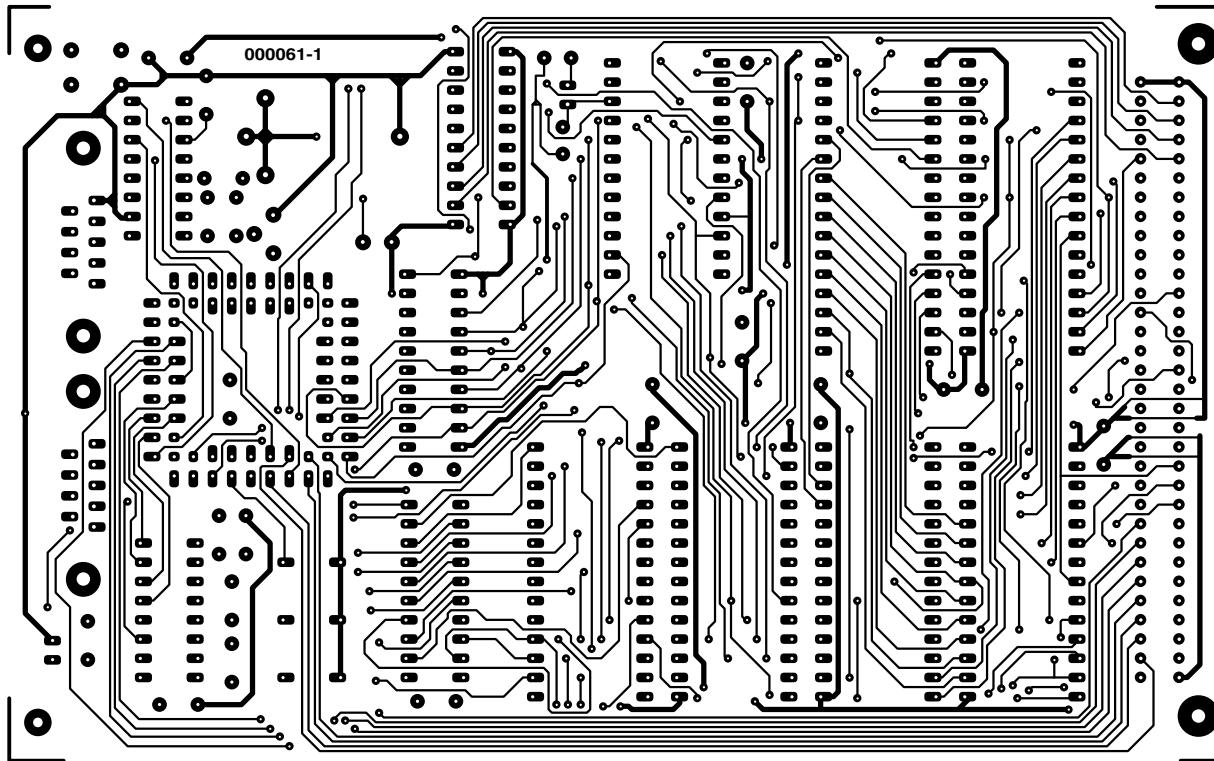
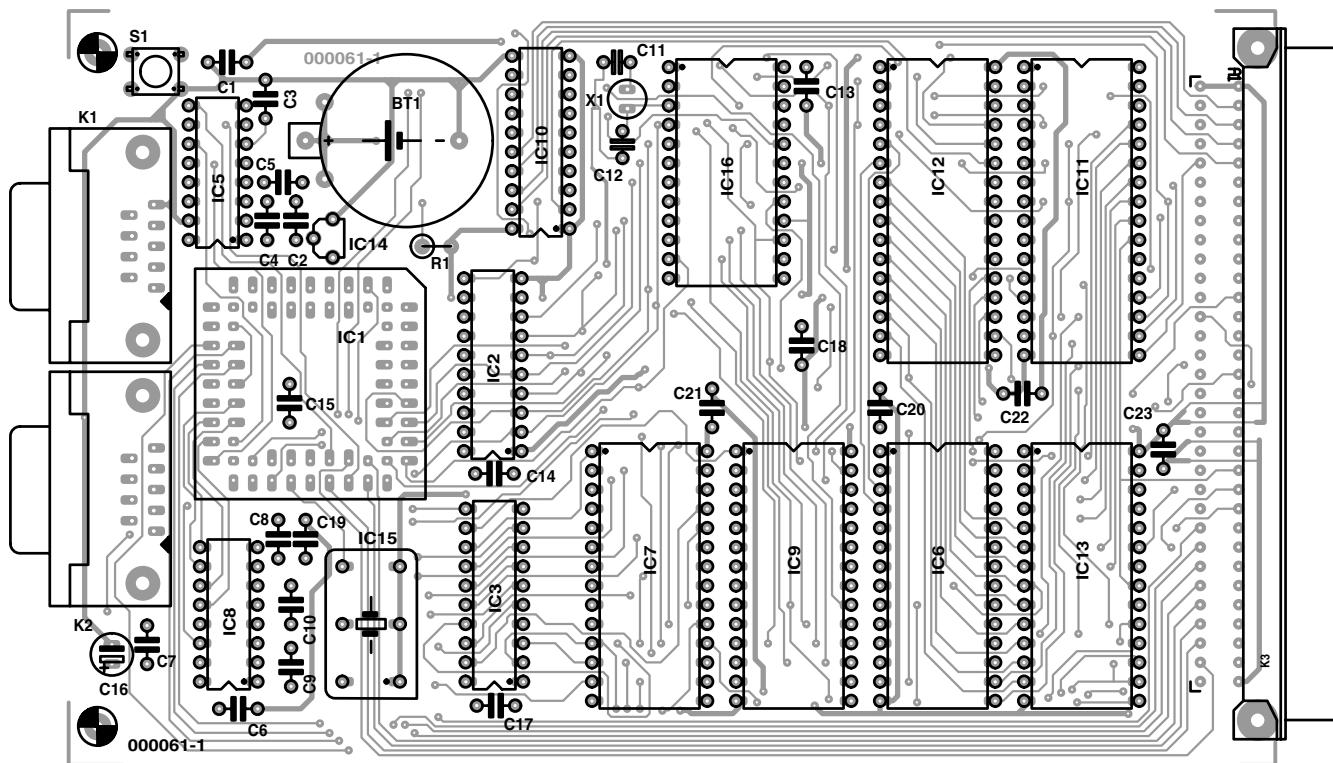


Figure 5. La platine double face à trous métallisés de la mono-carte à PIC17 est au format Europe.



## Liste des composants

### Résistances :

$R_1 = 10 \text{ k}\Omega$

### Condensateurs :

$C_1 \text{ à } C_{10}, C_{13} \text{ à } C_{15}, C_{17} \text{ à } C_{19} = 100 \text{ nF}$

$C_{11}, C_{12} = 12 \text{ pF}$

$C_{16} = 10 \mu\text{F}/63 \text{ V vertical}$

### Semi-conducteurs :

$IC_1 = PIC17C756$  (sans suffixe -A ou -B)

$IC_2, IC_3 = 74ACT573$

$IC_5, IC_8 = MAX232CPE$

$IC_6, IC_7 = 62256$  (HM62256BLPI2),  $= 120 \text{ ns}$

$IC_9, IC_{13} = 27C512$  (programmées **EPS000061-21a/b**),  
 $= 120 \text{ ns}$

$IC_{10} = GAL 16V8-D$  (**EPS000061-31**)

$IC_{11}, IC_{12} = AM29F010 120PC$  ou  $AM29F040 120PC$  (AMD)

$IC_{14} = MCP120D$  ou  $MCP130 450 DITO$

$IC_{15} = \text{bloc oscillateur } 16 \text{ MHz à 8 ou 14 broches}$  (Seiko 531P)

$IC_{16} = DS1685-5$  (Dallas Semiconductor)

### Divers :

$K_1, K_2 = \text{embase sub D à 9 contacts femelle encartable en équerre}$

$K_3 = \text{connecteur 64 contacts en équerre selon DIN64AC}$

$S_1 = \text{bouton-poussoir à contact travail, } 6 \times 6 \text{ mm}$   
(Bourns 7906H-001-000)

$X_1 = \text{quartz } 32,76 \text{ kHz}$

$BT_1 = \text{pile au lithium CR2032}$  (le cas échéant avec porte-pile)  
support PLCC à 68 contacts  
face avant pour rack 19" 5TE/3HE (Bopla)

en **figure 5**, n'appelle que peu de commutaires. Utilisez un fer à souder à pointe fine et veillez à ne pas faire de ponts de soudure involontaires. On utilisera, pour tous les circuits intégrés, des supports (de bonne qualité cela va de soi).

La platine est au format Europe, de sorte qu'il est facile de l'implanter, après l'avoir dotée d'une plaquette face avant de taille adéquate (1 HE), dans un rack 19". Si l'on opte pour cette solution, le bouton-poussoir de remise à zéro ne sera pas monté directement sur la platine mais placé sur la face avant et relié à la platine par le biais d'une liaison souple à 2 conducteurs.

## Éléments requis pour la mono-carte à PIC17

2 EPROM avec programme Moniteur en anglais  
GAL de décodage d'adresse

**EPS000061-21a/b**  
**EPS000061-31**

Le logiciel (tous les codes-source et le dessin de platine au format .pdf) peuvent être téléchargés depuis le site Internet d'Elektor à l'adresse [www.elektor.presse.fr](http://www.elektor.presse.fr).

## Environnement de développement

On pourra utiliser, comme environnement de développement en vue de la programmation de la mono-carte à PIC17, par exemple l'environnement MPLAB-IDE de Microchip Technology. Ce set de programmes est disponible, gratuitement, sur le site Internet de cette société, site qui se trouve à l'adresse suivante :

[www.microchip.com/10/Tools/PICmicro/DevEnv/MPLABi/Software/v50000/index.htm](http://www.microchip.com/10/Tools/PICmicro/DevEnv/MPLABi/Software/v50000/index.htm)

L'environnement de développement intègre l'ensemble des outils nécessaires, à savoir l'assembleur, le lieu (linker), l'éditeur et le simulateur.

Il est également possible d'intégrer un compilateur-C gratuit, qui n'existe malheureusement que sous la forme d'une version d'évaluation de 30 jours. On pourra également télécharger depuis le même site, un manuel et un tutoriel (tous 2 en anglais).

Une fois que l'on a écrit et testé son programme au coeur de MPLAB-IDE, il ne reste plus qu'à télécharger le fichier HEX vers la mono-carte. Grâce au programme Moniteur de la

mono-carte, cette opération se fait sans difficulté. Normalement, l'assembleur produit un fichier HEX pour une EPROM de 16 bits de large. Il est possible, par le biais du Moniteur, de paramétriser le format dans la mémoire Flash. Le choix du format S (*Split*) se traduit par la génération de 2 fichiers 8 bits destinés respectivement à l'octet de poids fort (*High*) et à l'octet de poids faible (*Low*).

Il est possible, en se servant de l'option *Send Textfile*, d'envoyer le fichier HEX, à l'aide de n'importe quel programme de terminal, vers la mémoire Flash de la mono-carte à PIC17 à partir de l'adresse 2000<sub>H</sub>. Il est également possible d'écrire le programme à un autre endroit de mémoire, directement dans la mémoire SRAM par exemple. L'adresse de début est toujours définie, dans le code-source, à l'aide de la directive d'assemblage *ORG Address*. Il est à noter cependant que, primo, la mono-carte ne lance, après une instruction G, que le programme démarrant à l'adresse 2000<sub>H</sub> et secundo, que lorsqu'elle se trouve en mode autonome (*stand alone*), la mono-carte s'attend à trouver un programme à partir de cette même

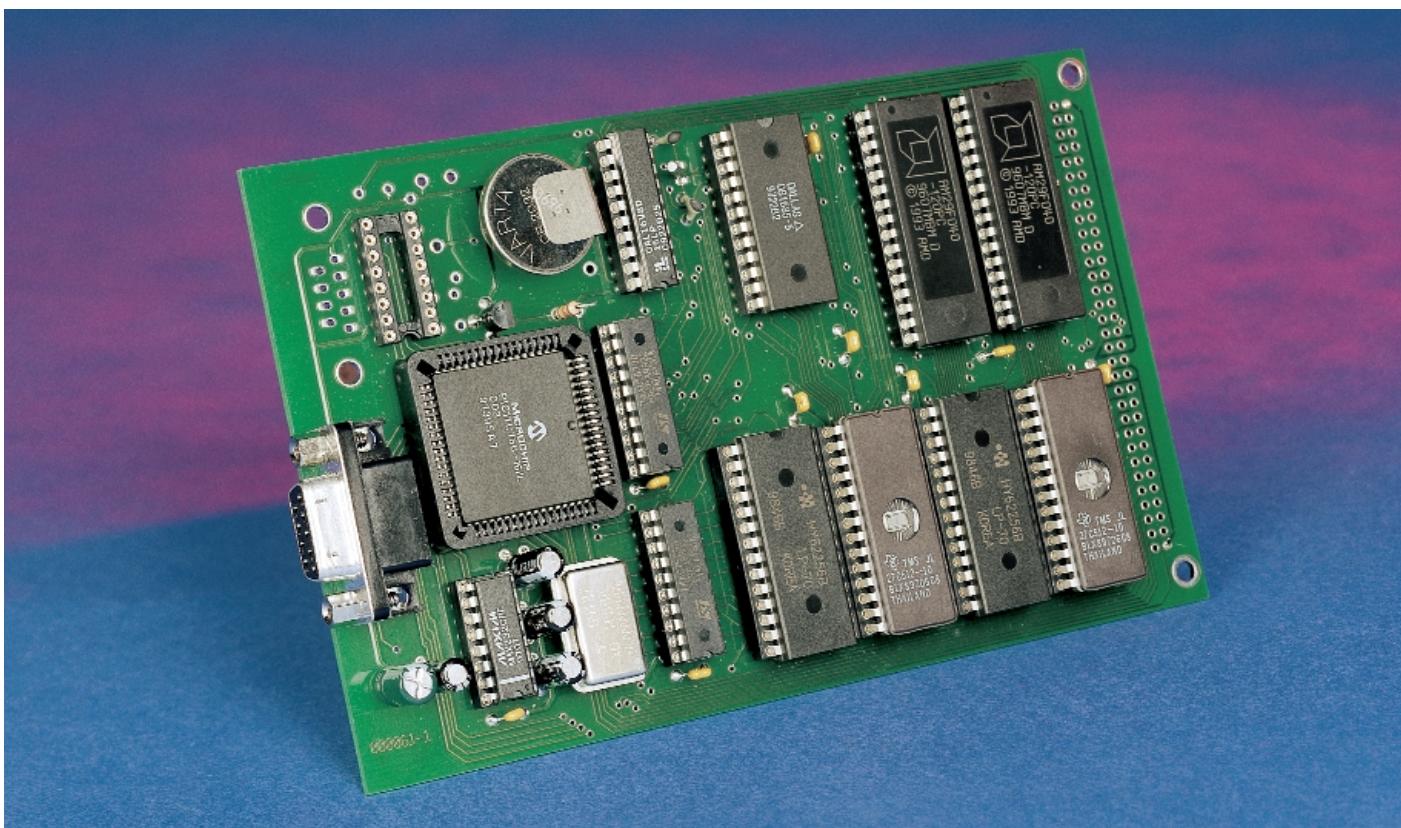
adresse de 2000<sub>H</sub>.

Il est possible d'activer le mode autonome sans avoir à basculer de cavalier de court-circuit (*jumper*). On réinitialise (*reset*) le contrôleur et, pendant 10 s environ, on fait en sorte qu'il n'y ait pas transmission d'instructions vers la mono-carte par le biais de son USART2. Dans ce cas-là, la mono-carte démarre automatiquement le programme commençant à l'adresse 2000<sub>H</sub>. Le Moniteur de la mono-carte à PIC17 possède un nombre d'instructions réduit que l'on aura vite fait d'assimiler :

D :	Download (Téléchargement)
S :	Entrer l'heure, S hh.mm.ss
T :	Donner l'heure
Z :	Entrer la date, Z DD.MM.YY
A :	Donner la date
B :	Lecture de bloc entier, B AAAA AAAA
W :	Écriture d'un mot, W AAAA BBBB
R :	Lecture d'un mot, R AAAA
G :	Démarrer programme en Flash
E :	Effacer Flash
H ou Alt-Z :	Aide concernant toutes les instructions et leur syntaxe

Après mise sous tension de la mono-carte, le Moniteur s'annonce (espérons-le) par l'affichage du message *Boot ok*; il répète ce message toutes les 12 secondes.

(000061)



# L'électronique sur PC

Expériences d'électronique avec l'interface sérielle  
(1<sup>e</sup> partie)

par Burkhard Kainka

Qui n'a jamais eu l'idée d'automatiser ou de surveiller quelque chose avec son PC ? Les expériences d'électronique sur PC ne sont pas forcément compliquées ou coûteuses. Il suffit souvent de se servir directement des interfaces disponibles du PC. Cette série d'articles est consacrée à des expériences basées sur l'interface sérielle qui peuvent être réalisées avec des programmes simples en Visual Basic.

Beaucoup se demanderont : Pourquoi justement l'interface sérielle, alors qu'elle est plus compliquée et possède moins de lignes que l'interface de l'imprimante ?

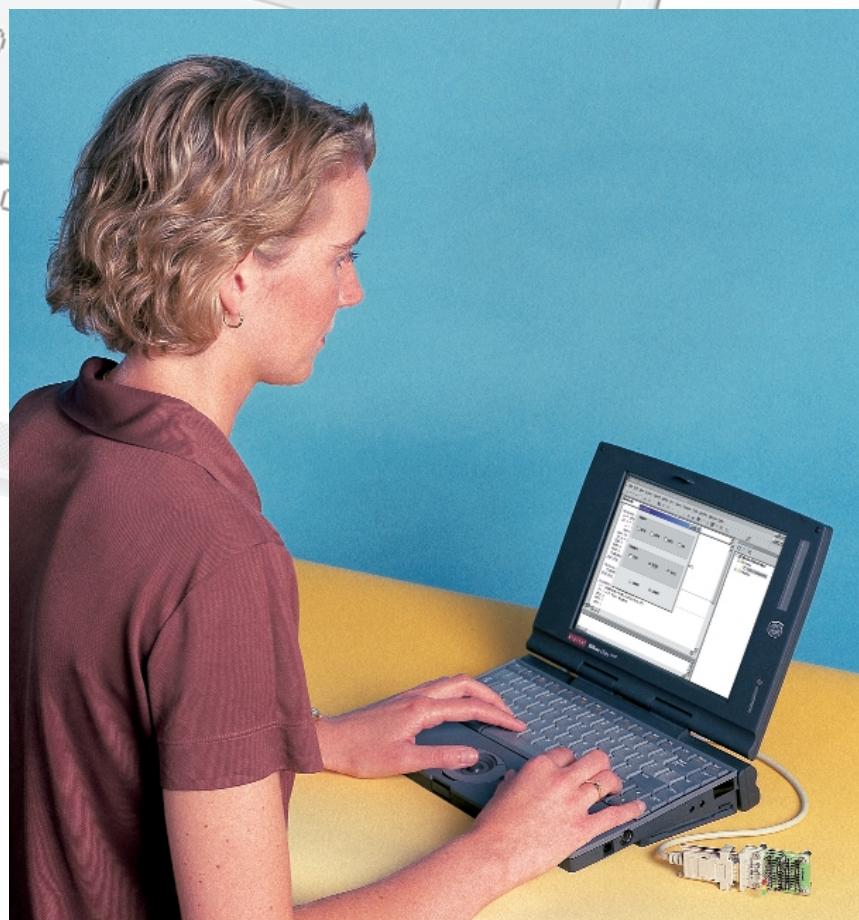
L'utilisation du port-série dans ce cours élémentaire a de nombreuses (bonnes) raisons :

- L'interface sérielle est très résistante aux manipulations qui pourraient causer des dommages involontaires. On peut raccorder les connecteurs pendant que le PC fonctionne sans risquer de l'endommager.

- Une des interfaces sérieelles est encore disponible dans la majorité des cas et peut être utilisée pour des expériences.

- Le courant fourni par l'interface sérielle elle-même est suffisamment élevé pour permettre de réaliser de nombreuses expériences sans l'aide d'un bloc d'alimentation secteur.

Les dépenses occasionnées par ce cours sont minimes, ce qui constitue un avantage quand il s'agit de travaux pratiques dans les écoles et les groupes de jeunesse. On n'a besoin, outre d'une petite platine de raccordement, que de quelques composants de la caisse à bricolage, par exemple des boutons-poussoirs, des transistors NPN, des résistances, des LED, des condensateurs électrolytiques, des diodes et une photo-résistance (connue sous la dénomination de LDR pour *Light*



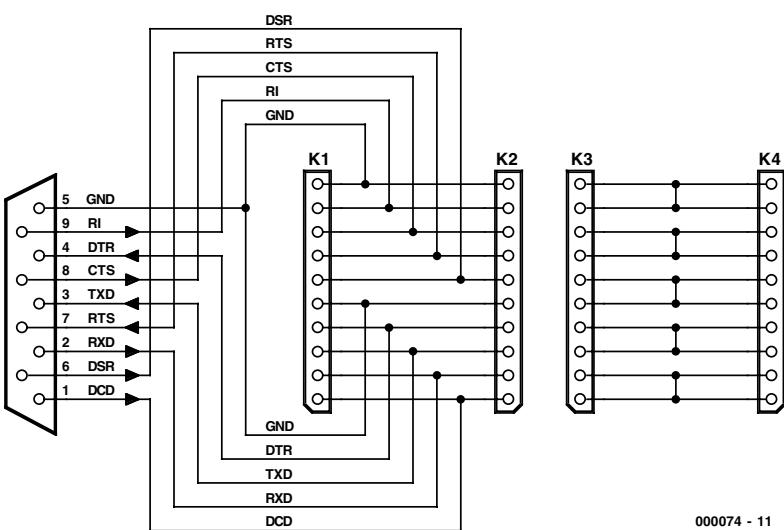


Figure 1. Circuit de la platine expérimentale.

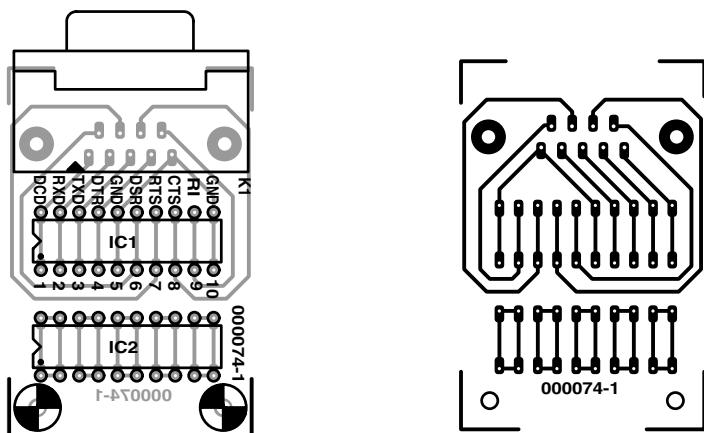


Figure 2. Disposition et plan de montage de la platine expérimentale.

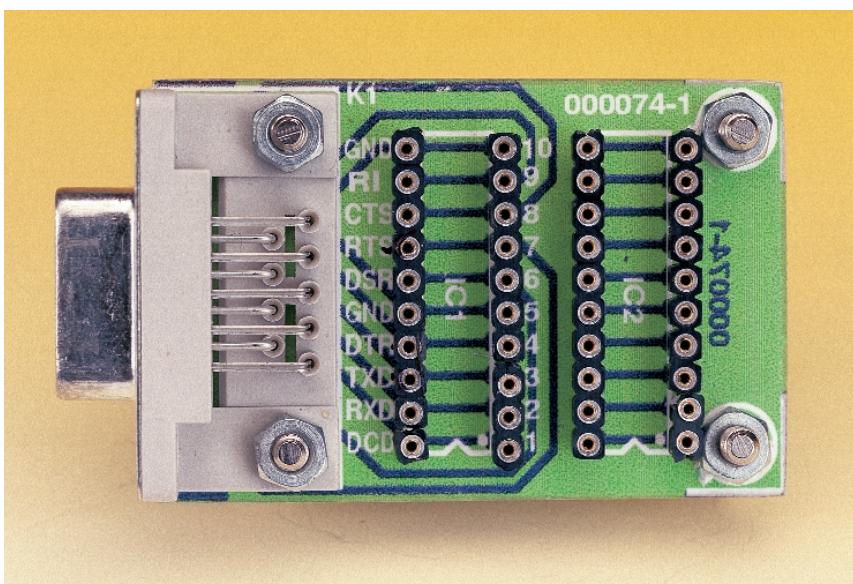


Figure 3. Modèle d'essai une fois monté.

Dependent Resistor). La platine de raccordement comporte des douilles pour toutes les lignes de l'interface série. Une petite carte d'essai servira au montage des circuits expérimentaux. Même le maniement du fer à souder est superflu : tous les composants sont simplement enfoncés.

Tous les essais sont programmés en Visual Basic 5. Les programmes (sur disquette EPS000074-11 et par téléchargement à partir de [www.elektor.presse.fr](http://www.elektor.presse.fr)) sont aussi à la portée des débutants, permettant à chacun de concrétiser ses propres idées sans difficulté. Les dernières parties exposeront aussi des thèmes plus complexes tels que les commandes à durée critique ou le couplage à une électronique externe. Un atout supplémentaire est l'initiation à la technique de mesure appliquée que l'on reçoit pour ainsi dire gratuitement.

## La platine

Une platine de conception très simple sert de base aux travaux. Comme le montre le circuit de la **figure 1**, il s'agit d'une carte de petites dimensions pour prototypage équipée de 4 réglettes à douilles et d'une prise D-Sub 9 broches (dite DB 9). Deux des 4 réglettes à douilles sont reliées à la prise D-Sub, les deux autres ne sont reliées qu'entre elles. Cette carte vous est présentée en **figure 2**. Il vous faudra la réaliser vous-même, mais sa complexité ne devrait pas avoir de quoi vous effrayer.

La **figure 3** représente la platine montée. Pour ne pas se perdre dans les connexions et autres raccordements des connecteurs 25 et 9 broches utilisés avec les interfaces sérielles, on consultera le **tableau 1** qui liste toutes les lignes de l'interface série et le brochage. Les connecteurs côté PC sont toujours équipés de broches (ils sont partant mâles); la connexion sera donc effectuée par connecteurs femelles. Dans le cas de la platine utilisée ici, on se sert d'un câble de prolongation à 9 lignes. Si le PC est muni d'un connecteur 25 broches, on fera appel à un adaptateur. L'échange de données proprement dit par interface série est effectué normalement par la ligne série d'émission TXD et la ligne série de réception RXD. Les lignes restantes ont des fonctions auxiliaires d'établissement et de contrôle de la transmission des données. On les désigne communément par lignes d'acquittement (*handshake*) à cause de leur fonction d'acquittement entre les appareils. Les lignes de « *handshake* » offrent en particulier l'avantage de permettre l'écriture ou la lecture directe de leur état.

L'ordre du brochage de la platine est le même

# COURS D'INITIATION

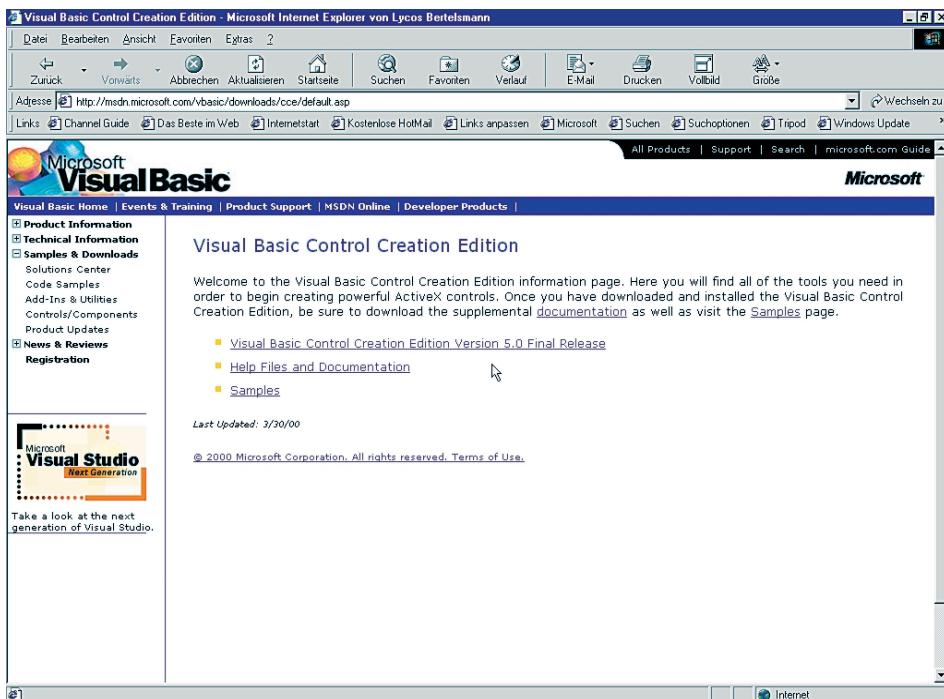


Figure 4. Emplacement du téléchargement de Visual Basic (VB5CCE).

que celui du connecteur D-Sub à 9 broches. Chaque ligne de ce connecteur est raccordé à 2 douilles. Seule GND possède 4 connexions. La petite carte d'essai sur la platine possède 5 champs à 4 douilles chacun. Les contacts décolletés des douilles sont placés sur 4 réglettes à 1 rangée. Il est aussi possible de se servir de deux boîtiers DIP à 20 connexions qu'il est plus facile de se procurer.

## Visual Basic

Le cours requiert la version 5 ou 6 de Visual Basic. Tous les codes-sources sont disponibles sur disquette et peuvent également être téléchargés depuis le site Internet d'Elektor. Elles peuvent aussi être chargées et compilées sans difficulté avec VB6. Pour qui ne possède pas encore Visual Basic et recule devant le prix, il est possible de se procurer une

## Tableau I. Brochage

Broche DB 25	Broche Db 9	Entrée/Sortie	Dénomination	Fonction
2	3	Sortie	TxD (Transmit Data)	Émission de données
3	2	Entrée	RxD (Receive Data)	Réception de données
4	7	Sortie	RTS (Request To Send)	Activation émission
5	8	EEntrée	CTS (Clear To Send)	Prêt à émettre
6	6	Entrée	DSR (Data Set Ready)	Prêt à émettre
7	5	GND	(Ground)	Masse
8	1	Entrée	DCD (Data Carrier Detect)	Niveau du signal de réception
20	4	Sortie	DTR (Data Terminal Ready)	Terminal paré
22	9	Entrée	RI (Ring Indicator)	Appel entrant

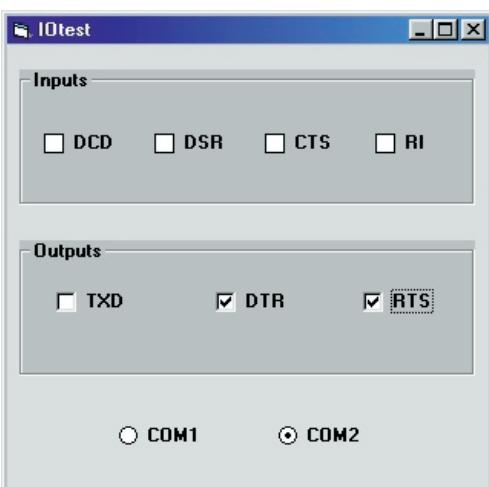
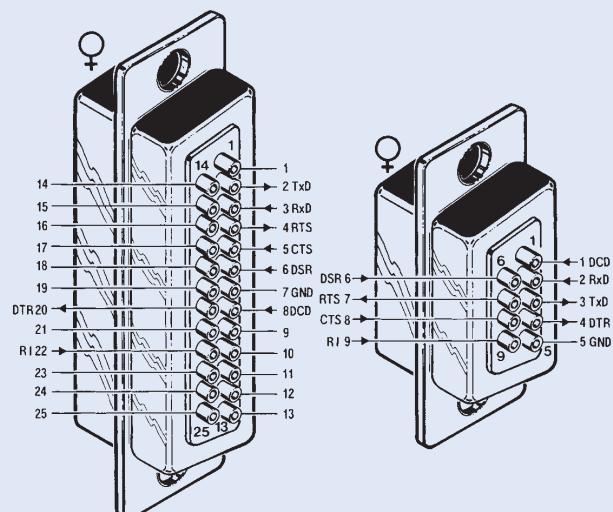


Figure 5. Fenêtre d'accès direct des lignes de l'interface sérielle.



version gratuite de VB5 auprès de Microsoft. « Visual Basic Control Creation » VB5CCE se trouve sur Internet sous :

<http://msdn.microsoft.com/vbasic/downloads/cce/default.asp>

## Le premier programme : I/O-Test

Passons enfin au premier programme. Il s'agit d'un programme de test peu compliqué donnant accès à toutes les lignes (hormis RXD). Trois sorties peuvent être commutées en cliquant, 4 entrées sont lues et affichées. Le programme est prévu pour l'interface COM2. Il est toutefois en mesure de s'apercevoir que COM2 est occupé et essaie alors automatiquement d'avoir accès à COM1. On a supposé que la souris de la plupart des PC est raccordée à COM1 et que COM2 est encore libre. Mais la désignation des interfaces sérielles d'un PC manque souvent de clarté. Ce programme peut alors s'avérer utile

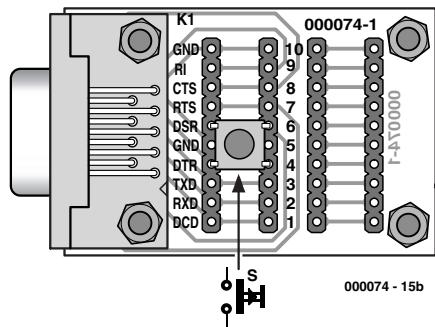
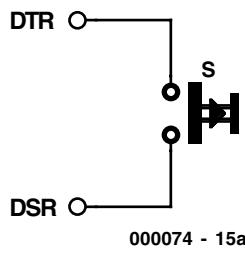


Figure 6. Connexion d'un simple interrupteur pour la première expérience.

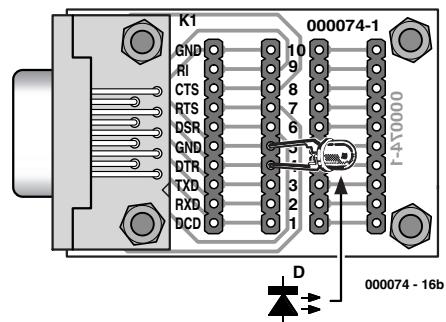
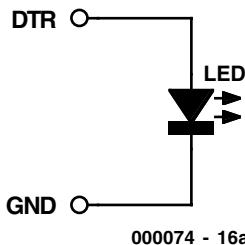


Figure 7. Il est tout aussi simple de raccorder une LED dans la seconde expérience.

## Contenu de la disquette

PORTS.BAS	845	03-28-00	5:31p
ANPEL.FRX	1,094	04-10-00	6:01p
BLINK.FRX	1,094	04-10-00	6:43p
COUNTER1.FRX	1,094	05-04-00	6:49p
COUNTER2.FRX	1,094	05-04-00	7:14p
CONTENTS.TXT	0	06-15-00	9:35a
IOTEST.FRX	1,094	05-04-00	12:21p
AMPEL.VBP	497	04-10-00	6:01p
BLINK.VBP	497	04-10-00	6:43p
COUNTER1.VBP	499	05-04-00	6:49p
COUNTER2.VBP	499	05-04-00	7:14p
IOTEST.VBP	497	05-04-00	1:13p
AMPEL.VBW	81	05-05-00	12:57p

BLINK.VBW	80	05-05-00	4:26p
COUNTER2.VBW	80	05-05-00	4:27p
IOTEST.BW	80	06-08-00	1:15p
PORT.DLL	46,080	02-07-99	1:15p
BLINK.FRM	3,488	04-10-00	6:43p
COUNTER1.FRM	3,422	05-04-00	6:49p
COUNTER2.FRM	6,683	05-04-00	7:14p
IOTEST.FRM	5,317	05-04-00	12:21p
ANPEL.FRM	3,669	04-10-00	6:01p
	22 file(s)	77,784 bytes	

Ces 22 fichiers se trouvent sur la disquette EPS000074-11 disponible auprès des adresses habituelles. Cette disquette comporte 4 sous-répertoires baptisés respectivement F, D, UK et NL pour chacune des langues concernées.

pour identifier une interface COM. Si les 2 interfaces COM sont libres, on peut se servir de COM1 ou de COM2 au choix.

Qu'est-ce qui est « EN CIRCUIT » et « HORS CIRCUIT » ? Si l'on relie simplement un voltmètre entre DTR et GND (masse), on constate que : « EN CIRCUIT » fournit une tension positive d'environ 10 V, « HORS CIRCUIT » fournit une tension d'environ -10 V par rapport à GND, la masse. Relions par exemple la sortie DTR à

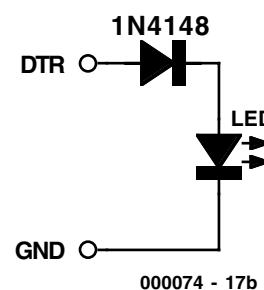
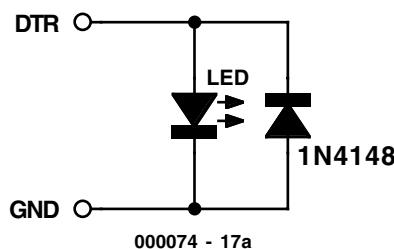


Figure 8. Deux façons de protéger la LED de tensions trop élevées en sens inverse.

l'entrée DSR par un fil. Dès que DTR est en circuit, l'entrée DSR devient aussi « EN CIRCUIT ». L'interrogation de l'état d'un interrupteur constitue la première application pratique (**figure 6**). L'entrée DTR doit être en circuit pour que l'état de l'interrupteur soit reconnu. Et voilà déjà une application sur PC ! L'interrupteur peut se trouver à un endroit éloigné et une information peut être transmise au PC. Il appartient à l'utilisateur de décider de la nature de cette information. Qu'en serait-il par exemple de la surveillance du piège à souris placé à la cave ? Encore une toute petite application pour compléter le premier épisode. Une diode luminescente est raccordée directement au PC qui

l'allume (**figure 7**). De nombreux lecteurs d'Elektor protesteront vraisemblablement qu'il ne faut jamais raccorder une LED sans résistance protectrice. Mais ici, la résistance de protection fait pour ainsi dire déjà partie du PC.

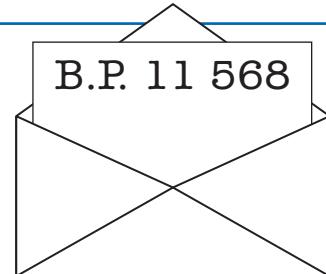
Autre objection : La tension inverse permise indiquée dans les fiches de données des LED est la plupart du temps de 3 V ou 5 V. Mais la tension du circuit présenté ici peut dépasser 10 V dans la direction inverse. L'expérience montre toutefois qu'il s'agit dans une large mesure d'un faux problème : le courant qui passe en

direction inverse dans la plupart des LED est négligeable, même lorsque la tension appliquée dans cette direction atteint 20 V. Mentionnons toutefois la solution « propre ». Une diode au silicium, une 1N4148 par exemple, suffit pour protéger la LED de tensions trop élevées en direction inverse. La diode peut être connectée anti-parallèlement ou en série (**figure 8**). Une règle empirique : limiter la tension inverse à 5 V pour les véritables projets de développement, abandonner cette belle rigueur pour les projets simples.

(000074)

## COIN DU LECTEUR

Nous ne pouvons malheureusement pas répondre in extenso à toutes les lettres relevant des questions techniques. Dans cette rubrique nous répondons à des lettres pouvant présenter un intérêt général et concernant des montages âgés de moins de 2 ans. Vu le nombre de lettres qui nous arrivent mensuellement, nous regrettions de ne pas pouvoir répondre séparément à chacune d'entre elles et sommes dans l'impossibilité de donner suite à des souhaits individualisés d'adaptation de montages publiés ou de réalisation de montages à publier ni même de répondre à des demandes d'information additionnelle concernant un montage décrit dans Elektor.



### Démagnétiseur pour CD

Bonjour,  
Je regrette d'avoir à vous embêter un petit peu, mais dans le numéro Juillet-Août 2000. Il y a un démagnétiseur de CD ROM alors, ceci n'est pas possible, car un CD est constitué d'une couche d'aluminium déposé sous forme de vapeur sous vide. L'aluminium n'est pas un métal magnétique (ne pas confondre avec les courants de Foucault). Il est fait allusion aussi que la lumière est déviée par le magnétisme, c'est encore impossible. L'exemple est donné avec la lumière déviée dans un tube cathodique, mais ce n'est pas de la lumière (photons), mais des électrons qui effectivement eux sont déviés par un champ magnétique. (C'est la couche de l'écran qui est phosphorescente qui reconvertis le faisceau d'électron en lumière visible). De ce fait il n'y aucune chance pour que des erreurs pendant la

lecture soit dues à cela. Généralement de telles erreures de lectures sont dues au manque de soin apporté au correcteur d'erreurs (électronique) de la platine CD.

Merci de votre attention, j'espère que vous ferez un petit récapitulatif.

Dans l'attente de vous lire, recevez mes sincères salutations.

**CpeziCola@aol.fr**

Dans l'Elektor des vacances, un montage amusant et simple pour démagnétiser les CD ....

Un petit moteur fait tourner une plaque sur laquelle sont attachés deux petits aimants dont les polarités sont perpendiculaires (je veux dire les directions des deux champs).

Bonjour,  
Seraît-ce le montage bidon de ce numéro juillet-août ???

Qu'en penser ???? Moi électricien de métier, qui ai appris le magnétisme à l'école. Je n'en reviens pas, et j'en suis presque tombé sur le c.. !

Un CD étant constitué d'une couche d'aluminium ou d'or, de polycarbonate plus une goutte de vernis, il n'y a rien de magnétique donc rien de démagnétisable. Les seuls traitements que l'on peut effectuer sur un CD concernent l'aspect optique, électrostatique et MDI. (Micro décharges d'interfaces, recherches de Mr Pierre Johannet, plus connu du monde Audioophile pour ses recherches, et ses articles dans la revue l'Audioophile). Il a dû rigoler cinq minutes, voir son commentaire.

**GROS Jean-François**

Et en plus, dans l'article, ils écrivent sans rire que les champs magnétiques font dévier les rayons lumineux/laser, en ajoutant

tant pour ceux qui n'auraient pas compris: "c'est comme ça que les rayons sont déviés dans les oscilloscopes!".

Légère différence: dans les oscillos, ce sont les rayons cathodiques constitués d'électrons qui sont déviés, jamais personne n'a réussi à dévier un rayon électromagnétique avec un champ quelconque... Ca se saurait!

Maintenant, au point où on en est, il n'est même pas sûr que cet article soit volontairement bidon, quand on voit que de l'eau distillée dûment magnétisée améliore la variété et pas le classique!

Gardons le moral!

**BIJOJO**

Chers Lecteurs,  
C'est en effet un montage simple à réaliser et amusant... Ce ne sont ici que quelques-uns des nombreux E-mails que nous avons valu la publication d'un article (dérivé au demeurant d'un appa-

reil vendu dans le commerce !!!) sur un démagnétiseur de CD. Certains d'entre vous ont quand même eu une impression de « déjà-vou » comme disent les Américains. Certains lecteurs comme Mr Gros, ont cependant pensé tout de suite à une possibilité de montage-bidon, d'autant plus que, pour forcer le portrait, nous avions ajouté que le faisceau de lumière subissait une déviation, alors que nous savons bien que c'est d'un faisceau d'électrons qu'il s'agit....

De quoi faire parler dans les chaumières de France, de Navarre et du reste de l'Europe d'ailleurs puisque ce montage-bidon a été publié dans nos 4 éditions, française, anglaise, allemande et néerlandaise.

À première vue, c'est peut-être la première fois que certains d'entre nos lecteurs rencontrent un montage-bidon dans un numéro double, tradition éléktorienne vieille depuis plus de 20 ans....

Cordialement, La rédaction

### Yeux sensibles

Je possède des yeux très sensibles qui m'empêchent de travailler avec des écrans d'ordinateur. Je viens de découvrir qu'il existe des agendas électroniques dotés d'écrans sombres qui ne rayonnent pas. Serait-il possible d'adapter un tel affichage pour pouvoir l'utiliser avec un ordinateur classique et si oui, comment faire ?

### M.R. Dane

Il n'existe pas de solution simple pour utiliser de tels affichages en tant que moniteur d'ordinateur. Avez-vous déjà essayé des moniteurs pour ordinateur de type LCD ou TFT. Il en existe peut-être des modèles qui tirent parti de la lumière réfléchie et n'en émettent pas eux-mêmes.

### OUT en QBASIC

On m'a dit qu'il était possible, en QBASIC, de faire commuter des relais par le biais du port COM d'un PC; c'est là l'idée que j'envisage d'utiliser pour un appareil que je dois mettre au point. Mais je n'arrive pas à trouver les instructions nécessaires, ni par le biais du menu d'Aide, ni dans les

ouvrages que je possède. Un collègue qui vous lit régulièrement m'a soufflé à l'oreille que vous pourriez peut-être avoir la solution. Pourriez-vous peut-être me donner l'instruction à utiliser pour résoudre ce problème ?

**Denis Russo**

Il est possible, en QBASIC, de piloter directement les ports COM par le biais de l'instruction OUT. Il suffit pour cela de faire suivre l'instruction OUT par l'adresse du port COM concerné. Vous trouverez la syntaxe de cette instruction dans le fichier D'Aide de QBASIC en vous référant à l'instruction OUT.

### Nouveau préamplificateur

Le Prélude (modulaire) est et reste, en ce qui me concerne, l'un des projets les plus intéressants qu'ait jamais publié votre magazine. Il est vrai que cela fait déjà bien longtemps. Il me semble que reprendre ce projet avec la technologie actuelle pourrait intéresser nombre de vos lecteurs. Un préamplificateur numérique modulaire avec DSP, conversion N/A pour CD, DVD-A, DVD-V, SACD et doté des entrées et sorties numériques requises, avec, éventuellement Dolby Surround, Dolby Digital, des décodeurs THX optionnels....

Tout cela sous la forme d'un projet bien évidemment « no-nonsense », sans compromis non plus, approche à laquelle nous sommes habitués en ce qui vous concerne. Vous pourriez prendre en exemple le nouveau préamplificateur DC 330 d'Accuphase présenté sur Internet à l'adresse : [www.accuphase.com/dc-330\\_3.htm](http://www.accuphase.com/dc-330_3.htm).

Devons-nous garder espoir ?

**Tony De Lobelle**

Il va sans dire que cela fait un moment déjà que nous cultivons l'idée d'un nouveau préamplificateur. Le problème rencontré dans le cas d'une approche numérique est le besoin d'écrire un logiciel de pilotage interne et de commande du système complexe, opération qui prend énormément de temps. Le problème additionnel qui se pose en ce qui

concerne le décodage Surround-Sound est que les circuits intégrés requis ne sont pas disponibles pour le grand public, ceci pour des raisons de droits de licence. Il n'en reste pas moins que cette idée continue de nous trotter dans la tête. Tôt ou tard nous aurons l'occasion de vous présenter un nouveau préamplificateur haut de gamme, le seul point d'interrogation est que nous ne savons pas encore sous quelle forme !

### Expériences avec le port imprimante

Cela fait un certain temps déjà que je procède à des expériences faisant appel à mon port imprimante. J'en suis arrivé au point où il m'est possible, par une simple action sur un bouton, d'allumer l'éclairage de ma chambre et de démarrer ma chaîne audio. Je programme en QBASIC et en Pascal. J'envoie ensuite les 8 bits de données vers le port imprimante. Je les amplifie par le biais d'un transistor associé à un relais. Il se veut maintenant que j'aimeraï pouvoir utiliser un appareil en entrée. Je dispose en effet d'un capteur (vous connaissez également ce genre de truc qui allume un éclairage lorsque vous passez à proximité) et je voudrais le connecter au port COM pour que, lorsque je rentre dans ma chambre mon ordinateur détecte ce passage de personne et le note dans un fichier log adéquat, ou pour qu'il mette automatiquement l'éclairage et la musique en route. Ma question est simple : savez-vous comment je dois m'y prendre ? (Quels sont les bits de donnée du port COM etc).

P.S. J'utilise mon 386, de sorte que ce n'est pas très grave si je démolis mon port imprimante, cet E-mail vous étant envoyé par le biais de mon Pentium III/600 MHz.

**Tjerk Kostelijk**

Nous avons publié, dans le numéro de janvier, un article intitulé « expérimentateur LPT ». Il donne un certain nombre d'informations intéressantes en

ce qui concerne les entrées et sorties au niveau de ce port.

### Adaptation d'un manche Nintendo au PC

Dans l'article publié dans le numéro de février 2000 vous affirmez que les connecteurs requis par cette réalisation ne sont pas disponibles dans le commerce. J'ai découvert par hasard que le FNAC de Bruxelles vendait des câbles de prolongation pour la console Nintendo-64. Il suffit de couper la fiche mâle et l'on dispose, pour 299 FB/49 FF, d'un câble doté d'une fiche femelle de toute beauté.

Merci de votre information. Nous la fournissons à nos lecteurs, l'Hexagone comptant un nombre important de magasins FNAC.

### Holtek HT12A/HT12D

Après avoir lu l'article publié dans votre numéro de février 2000 « télécommande IR pour le baladeur MiniDisc MZ-30 de Sony », j'ai commandé la platine chez vous. Il apparaît cependant qu'il soit pratiquement impossible de trouver le décodeur HT12D. Mon revendeur de composants m'affirme que le distributeur n'accepte que des commandes par quantités très importantes. En savez-vous plus ?

Nous avons pris contact avec Holtek et il apparaît effectivement que la plupart de leurs distributeurs n'acceptent pas de vendre à la pièce. Il existe cependant, pour ceux d'entre nos lecteurs qui « grenouillent » sur Internet et qui n'ont pas peur d'utiliser leur carte de crédit, une solution : Maplin dont l'adresse Internet est : [www.maplin.co.uk](http://www.maplin.co.uk) possède et le HT12A et le HT12D à leur catalogue. Le seul hic dans cette approche est le montant relativement élevé des frais de port auxquels on se trouve confronté lors d'une commande. Mais cela marche et rapidement, nous l'avons vérifié par nos propres moyens, n'est-ce pas Monsieur Corbalan, si tant est que leur stock de ce type de composant ne soit pas épuisé.

# Tube Preamp (II)

## Partie 2a : réalisation et résultats de mesure

Projet : Gerhard Haas

Une fois terminée l'opération d'implantation des composants sur les différentes platines qui le constituent, on pourra enfin procéder au câblage du Tube Preamp. Le résultat (tant visuel que mesuré) a de quoi impressionner.

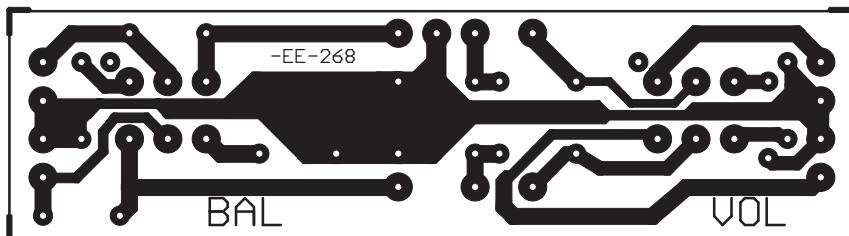
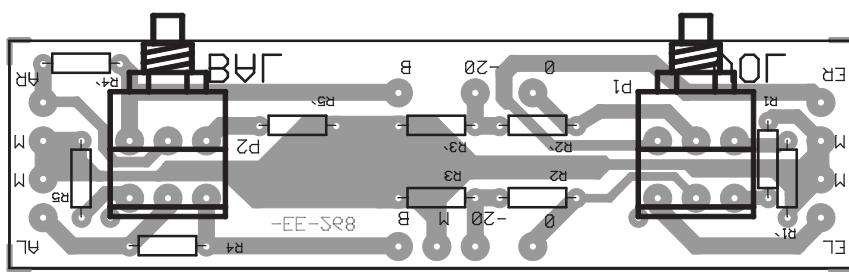
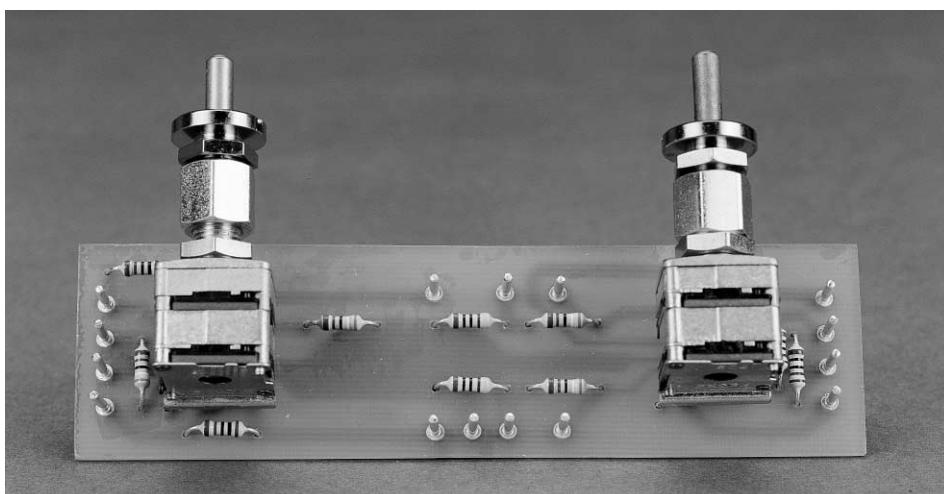


Figure 1. Platine de faible taille du volume et de la balance.



Cependant il reste, avant de pouvoir passer à cette étape si importante qu'est la réalisation, à planter les composants sur 4 plaques supplémentaires, à savoir la plaque de volume et de balance (**figure 1**), la petite plaque des relais de sortie (**figure 2**) ainsi que les plaques d'alimentation Haute et Basse Tension (**figures 3** et **4** respectivement).

La plaque de faible taille représentée dans la figure 1 comporte les relais de sortie et la douille de jack de commutation. Elle ne nécessite aucun élément de fixation car la douille assume ce rôle. Il faut à tout prix éviter de se tromper en attribuant l'entrée et la sortie du signal de l'amplificateur : le jack n'est protégé que si le câblage est correct. Veiller aussi à ce que la polarité des relais soit bien celle indiquée sur le

### Liste des composants du réglage de volume

- R1, R1' = 100  $\Omega$
- R2, R2', R4, R4' = 3k $\Omega$
- R3, R3' = 470  $\Omega$
- R5, R5' = 10 k $\Omega$
- P1, P2 = potentiomètre 10 k $\Omega$  linéaire à crémaillère (crans sensibles)
- 2 supports pour potentiomètre
- S1 = interrupteur à bascule 2 circuits/3 positions (de bonne qualité à contacts dorés)
- Picots

schéma de montage. Les contacts sont à l'abri des salissures dans des relais entièrement fermés. Ils sont en outre dorés et, selon la fiche de données, homologués à 250 V.

Il est inutile de s'attarder sur le montage de la platine du circuit de protection (figure 2) et sur celui des deux plaques d'alimentation des figures 3 et 4. On peut se servir de socles pour les circuits intégrés. Veiller (comme toujours) à la position correcte et à la polarité de tous les composants et tout ira bien. Nous n'avons pas représenté ici de plaque pour l'alimentation secteur symétrique facultative. Les informations livrées par les nombreux projets Elektor devraient suffire.

Les plaques dont tous les composants ont été posés sont prêtes à être montées dans le boîtier. Le boîtier prévu décrit ci-dessus comporte

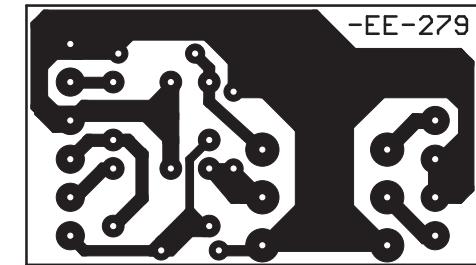
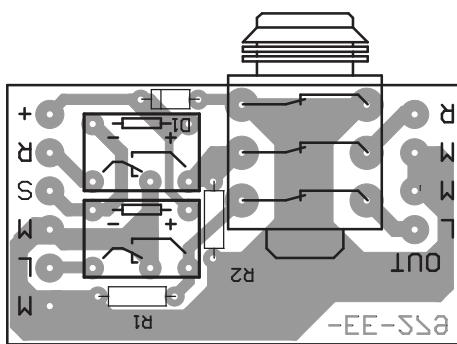


Figure 2. La petite plaque des relais de sortie et de la prise du casque.

#### Liste des composants du casque/commutateur Ligne

R1,R2 = 680 kΩ  
Re1,Re2 = relais unipolaire inverseur 12 V (Omron G6E)  
K1 = embase jack stéréo encartable (Source : Re-an)

#### Liste des composants de l'alimentation Haute Tension

##### Résistances :

R1 = 10 kΩ/2 W  
R2 = 100 Ω/4W5  
R3,R4 = 1 kΩ  
R5 = 6Ω8/2 W  
R6 = 150 kΩ/2 W

##### Condensateurs :

C1 = 100 µF/400 V, RM 10  
C2,C3 = 22 µF/400 V, RM 7,5

##### Semi-conducteurs :

T1 = BUZ92  
T2 = BC546  
D1 à D3 = diode zener 10 V/IW3  
D4 = diode zener 18 V/IW3  
G11 = pont B500C1500, modèle plat

##### Divers :

Radiateur pour T1 : SK 68/50 (Fischer) par exemple  
Petit matériel d'isolation et de montage pour T1 et le radiateur  
Picots

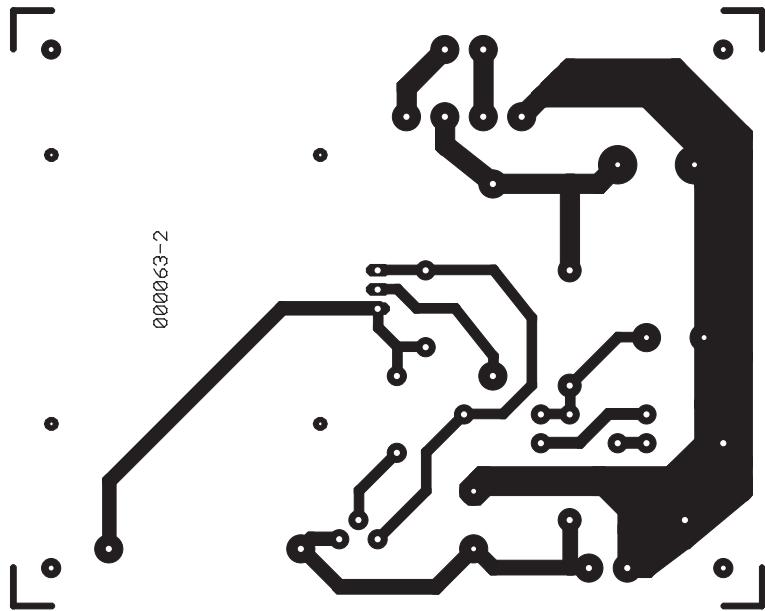
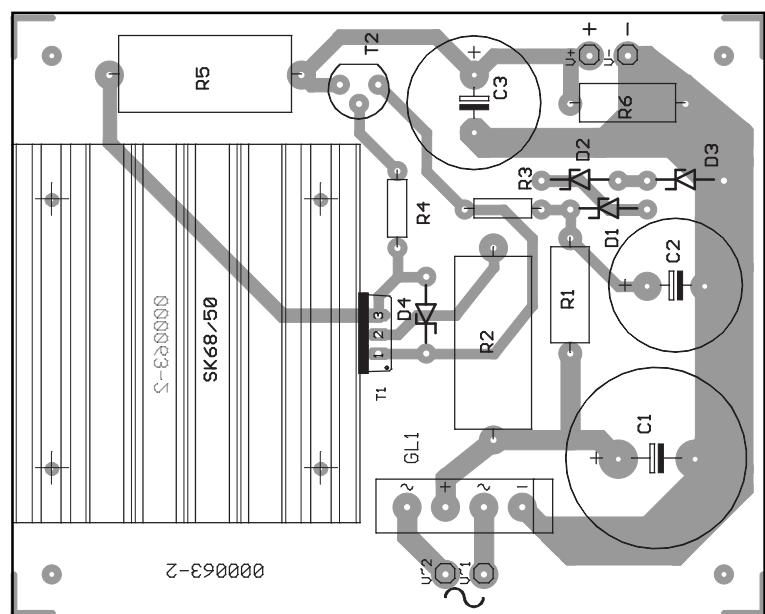


Figure 3. La plaque de l'alimentation Haute Tension (HT) et ...

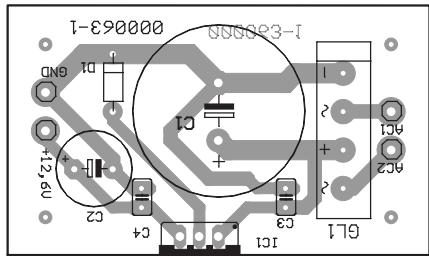
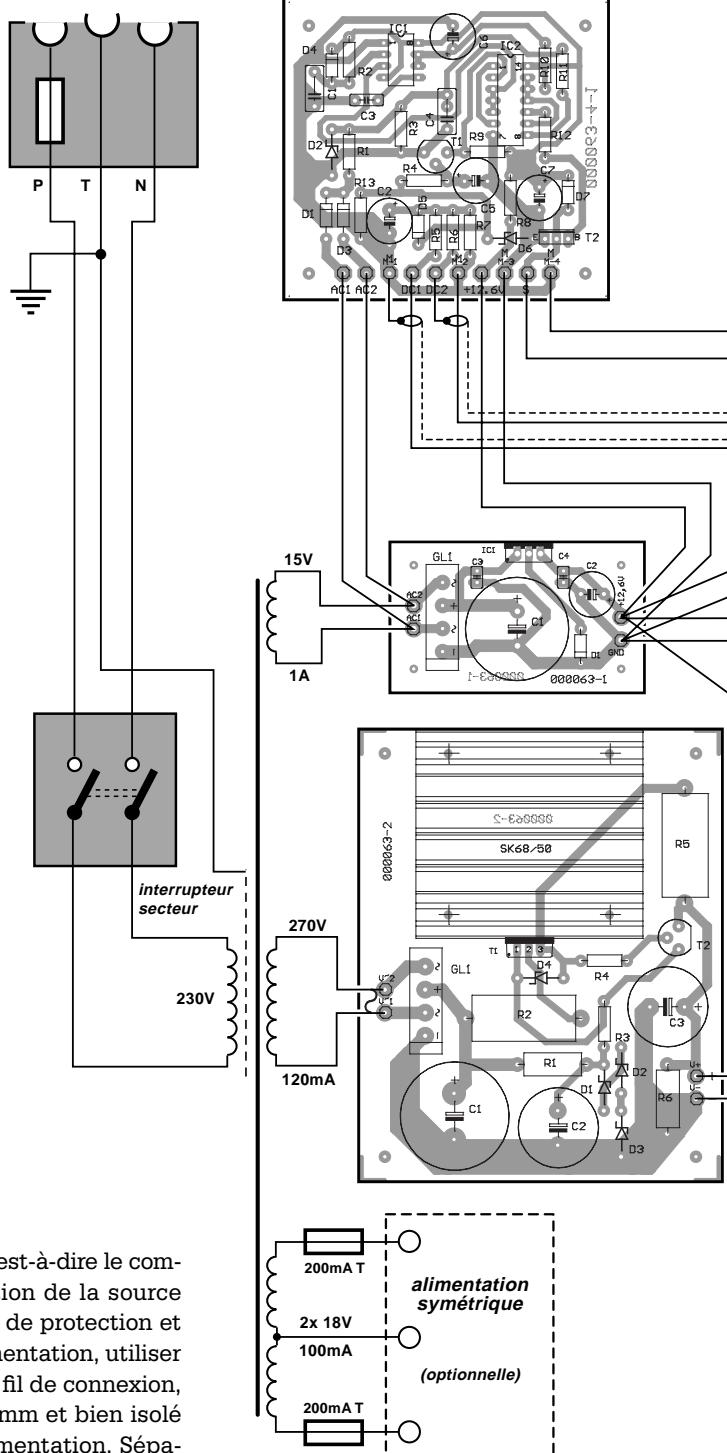


Figure 4. ... celle de l'alimentation Basse Tension.

### Liste des composants de l'alimentation Basse Tension

C1 = 2 200  $\mu$ F/40 V, RM 7,5  
 C2 = 100  $\mu$ F/40 V, RM 5  
 C3,C4 = 100 nF céramique RM 5  
 GL1 = pont B80C1500 (modèle plat)  
 D1 = IN4148  
 ICI = 7812 (TO 220)  
 Petit matériau d'isolation pour ICI  
 Si1,Si2 = fusible 0,2 A lent + porte-fusible  
 encartable  
 Picots



déjà tous les trous nécessaires ; il suffit donc de visser les éléments (dont également la prise secteur, les interrupteurs, les potentiomètres et les douilles de l'enregistreur). Si vous construisez votre boîtier vous-même, veillez à y disposer rationnellement les plaques et à bien les fixer. Il suffit par exemple de disposer le module d'entrée à proximité immédiate du transformateur d'alimentation – et bonjour les ronflements.

### Tiré au cordeau

Le câblage des plaques montées est la tâche qui devrait s'avérer la plus ardue. La méthode la plus sûre (oui, même au laboratoire d'*Elektor*) consiste à cocher chaque liaison effectuée sur une copie du plan de câblage de la **figure 5**.

Toutes les lignes qui transmettent le signal audio doivent être blindées et de bonne qua-

lité. Pour le reste, c'est-à-dire le commutateur de sélection de la source du signal, le circuit de protection et l'ensemble de l'alimentation, utiliser tout bonnement du fil de connexion, d'une section de 1 mm et bien isolé dans le cas de l'alimentation. Séparez aussi bien que possible les liaisons des signaux des fils de connexion et fixez les câbles au boîtier à l'aide d'attaches. Les douilles de la sortie enregistreur sont reliées à la plaque à partir de la barre de liaison par des cavaliers.

### Le grand moment

Une fois que le câblage est complètement au point et que tout a été vérifié 2 et même 3 fois, on peut se risquer à tenter un premier essai de

fonctionnement. Retirer tout d'abord la liaison entre le transformateur réseau et la plaque haute tension (et mettez-la à l'abri de tout contact). La tension de chauffage doit immédiatement monter à 12,6 V et être à l'épreuve des courts-circuits. Elle peut varier entre  $\pm 5\%$ . Les filaments des tubes doivent briller visiblement après 1 à 2 minutes. Cela constituera plus tard l'indication que le préam-

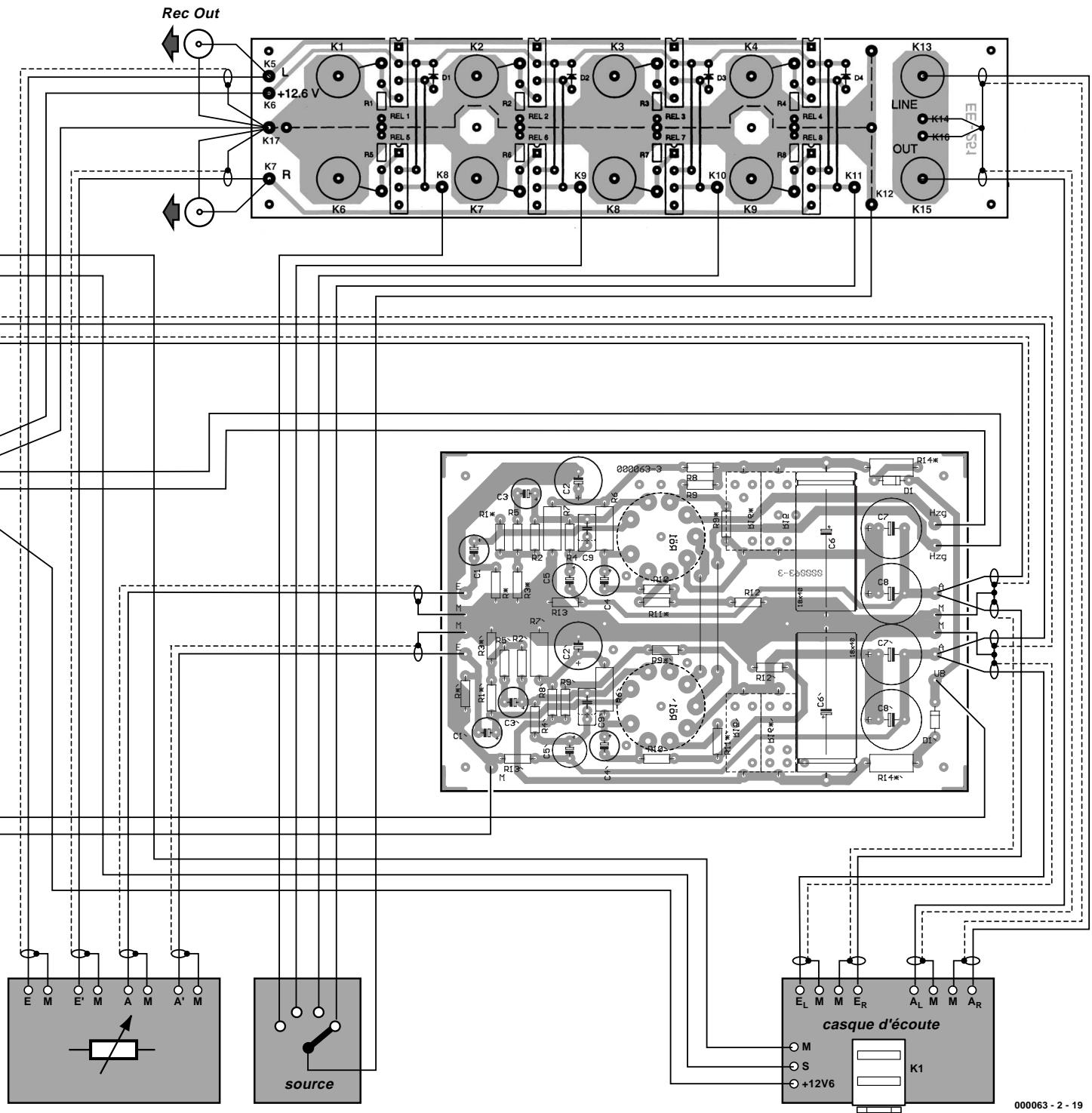


Figure 5. Le câblage du Tube Preamp dans toute sa complexité.

plificateur est en fonctionnement. Pour une indication plus précise, il suffit de raccorder une LED en série avec une résistance de protection et une diode 1N4007 à l'enroulement 15 V du transformateur réseau. On peut aussi tester à ce stade le bon fonctionnement des relais. Brancher la haute tension à la suite de ce premier essai. Elle doit atteindre sa valeur nominale peu

après l'enclenchement. Pas de fumée ? Contrôler les valeurs nominales du schéma de câblage, puis le parcours de la B.F. et ses modifications au moyen d'un générateur sinusoïdal et d'un oscilloscope. Essai réussi ? Visser le boîtier, raccorder le préamplificateur à tubes à une source de signaux et à un amplificateur final et tout mettre en marche. Les cathodes des tubes ont besoin

d'un certain temps de chauffe pour émettre un nombre suffisant d'électrons ; un appareil à tubes ne fonctionne donc vraiment qu'après environ 2 à 5 minutes. Mais il ne donne le meilleur de lui-même qu'au bout de 10 à 20 minutes, lorsque les tubes ont atteint leur équilibre thermique. Rien ne devrait alors plus venir troubler votre jouissance auditive.

## Résultats de mesure

Le laboratoire d'Elektor a soumis le préamplificateur à tubes à des essais très stricts dont nous ne voulons naturellement pas vous cacher les résultats. Les chiffres sont indiqués dans le tableau dans toute leur rigueur, tandis que les courbes de mesure méritent quelques commentaires. Toutes les mesures ont été effectuées après une période de chauffe de 4 heures. La tension d'entrée efficace était de 1 V et la tension efficace de sortie de 2 V. Les sorties ont été terminées par  $10\text{ k}\Omega$ , les entrées libres par  $600\text{ }\Omega$  (potentiomètre de symétrie en position médiane).

Rapport signal/bruit	Estimation acoustique	100 dBA
	Linéaire de 22 Hz à 22 kHz	87 dB
THD+bruit	1 kHz, B = 80 kHz	<0,05 %
	1 kHz, B = de 22 Hz à 80 kHz	<0,01 %
	1 kHz, B = de 400 Hz à 22 kHz	<0,05 %
IMD	(50 Hz : 7 kHz = 4:1)	<0,02 %
DIM	(rectangle 3,15 kHz + sinus 15 kHz)	0,003 %
Séparation des canaux	1 kHz	84 dB
	20 kHz	63 dB
Diaphonie	1 kHz	<-115 dB
	20 kHz	<-93 dB
Impédance d'entrée	Vol. min.	6,4 k $\Omega$
	Vol. max.	2,1 k $\Omega$
Impédance de sortie		<200 $\Omega$
Facteur d'amplification		2,54
Bande passante		de 3,5 Hz à 500 kHz
Plage de réglage de la symétrie		de +3 à -4,7 dB
Atténuateur		18 dB
THD	Tension de sortie 50 V <sub>eff</sub>	0,1 %

### Quelques remarques :

Les mesures ont été effectuées sans blindage des tubes, de sorte que des perturbations H.F. provenant de l'environnement de mesure (labo Elektor avec appareils de mesure de PC) pouvaient influencer les résultats de mesure. Il faut blinder les tubes si des sources de perturbations H.F. se trouvent à proximité de l'amplificateur. Voici ce qu'indiquent les 5 courbes de mesure :

#### A) Caractéristique de l'amplitude

L'échelle fortement agrandie jusqu'à 200 kHz (limite de puissance de notre générateur audio) indique une légère montée de la caractéristique. Mais la courbe est comme tirée au cordeau dans la plage audio qui nous intéresse !

#### B) Séparation des canaux

Les courbes se passent de commentaire, elles sont pratiquement identiques pour les 2 canaux. Elles ne commencent qu'à 200 Hz pour éliminer de ces mesures l'influence de l'ondulation de la tension d'alimentation.

#### C) Spectre de fréquence

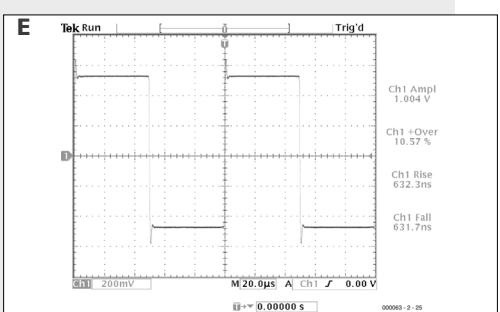
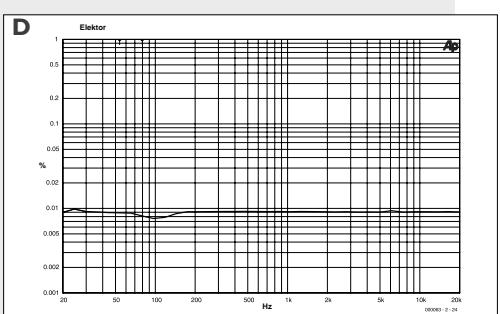
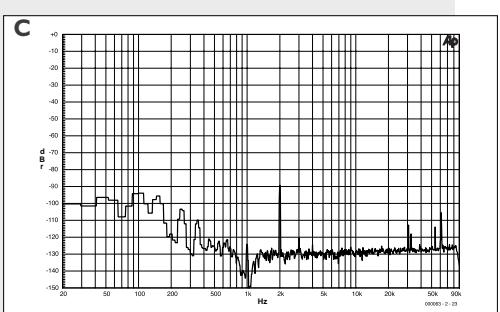
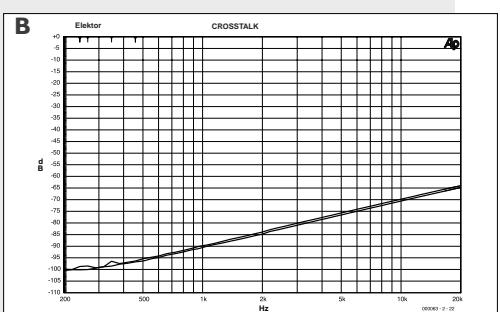
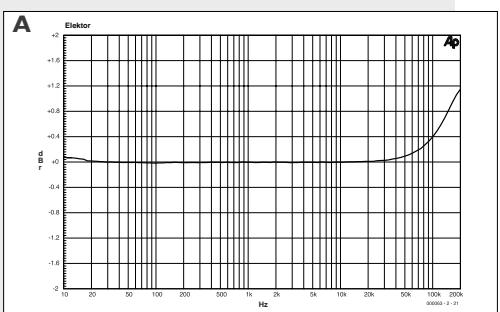
On remarque l'influence de l'ondulation de la tension d'alimentation sur le spectre de fréquence. Le spectre de l'ondulation atteint environ 800 Hz. Cela explique la différence relativement élevée entre les mesures basées sur une estimation acoustique et les mesures linéaires (100 dBA ou 87 dB). Les tons de 1 kHz (1 V<sub>eff</sub>) sont dominés par la deuxième harmonique qui atteint jusqu'à -90 dB. L'influence de champs parasites (en l'occurrence ceux d'un écran de type ancien se trouvant au voisinage de l'amplificateur) se reconnaît aux pics dans le haut du spectre de fréquence (30 kHz et 60 kHz) qui apparaissent lorsque les tubes ne sont pas blindés.

#### D) THD+bruit

La courbe a été déterminée pour une largeur de bande de 22 Hz à 80 kHz. La fraction causée par des signaux parasites est due à l'ondulation résiduelle de la tension d'alimentation et probablement aussi au rayonnement du transformateur.

#### E) Réponse aux transitoires

Un signal rectangulaire à l'entrée (10 kHz, 1 V) provoque une légère sursignalisation d'environ 10 % à la sortie.

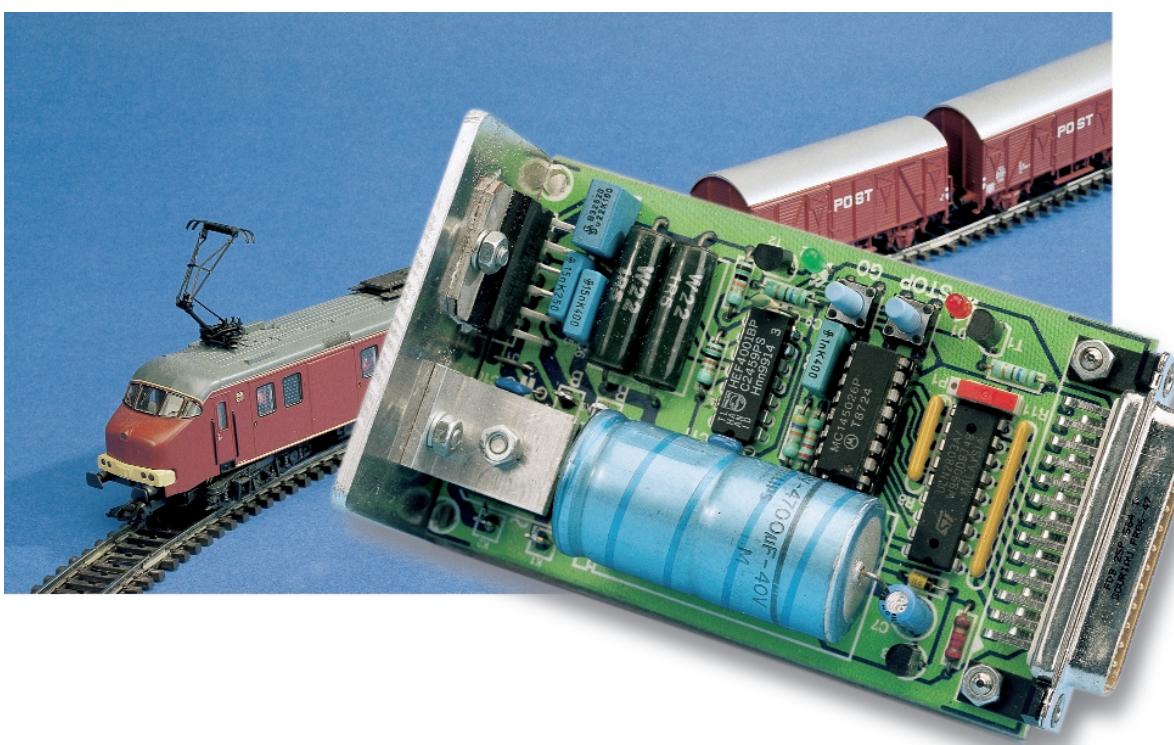


# Digital Railrunner-I

## Pilotage de réseaux ferroviaires par ordinateur

Joachim Schröder (idée et logiciel)

À l'image de son homologue de la SCNF néerlandaise, Railrunner, qui est une carte permettant de voyager à faible coût, est destiné aux amateurs de réseaux ferroviaires désirant piloter leur réseau miniature par ordinateur sans pour autant que cela ne leur coûte les yeux de la tête. Notre slogan : restons bon marché, le plaisir n'en sera que plus grand.



Il est loin d'être exclu que l'amateur (débutant) de réseaux ferroviaires miniatures ait déjà, sans qu'il ne le sache, un certain nombre de locomotives dotées d'un décodeur se promenant sur son circuit (Märklin) et auxquelles il peut, à l'aide d'un simple transformateur, insuffler la vie. Aujourd'hui, les locomotives dotées d'un décodeur coûtent à peine plus cher que leurs homologues traditionnelles (sans décodeur). Il n'y a rien d'étonnant à cela vu que l'électronique est

plus simple à produire en série et coûte moins cher que le traditionnel relais de commutation. Märklin a toujours conçu ses décodeurs de manière à ce qu'ils puissent s'accommoder d'un fonctionnement en alternatif (AC), c'est-à-dire qu'ils fonctionnent également avec leurs transformateurs traditionnels. L'utilisateur pourra toujours passer ultérieurement au numérique. Ce pas-

sage au numérique est donc techniquement possible, mais il implique un apport financier important qui peut fort bien, très vite, dépasser la valeur-même du matériel roulant. Il a été effectué diverses tentatives en vue d'abaisser ce seuil. Elektor a décrit dans ses pages, à partir de 1987, un système à réaliser soi-même, baptisé EDITS (*Elektor Digital Train System*) qui a connu un très

beau succès. De son côté, Märklin a lancé, voici quelque temps déjà, son système Delta, qui était en fait une variante améliorée de son systèmes Digital H0 à possibilités d'adressage de locomotive limitées (4 au lieu des 80 classiques). C'est très précisément de ce système Delta que ce projet tire sa source. De nombreuses locomotives de chez Märklin quittent, pour des raisons de standardisation, les chaînes montage dotées d'un décodeur Delta au lieu du relais de changement de sens de circulation classique.

Ces locomotives elles aussi sont commandées en mode AC standard (c'est-à-dire à l'aide d'un transformateur doté d'un rotacteur de commande) par la majorité des chefs de réseaux ferroviaires miniatures. Digital Railrunner permet à chacun d'entre eux, pour peu qu'il sache manier un fer à souder et qu'il dispose d'un (vieux) PC, de faire connaissance, à moindres frais, avec le concept de la numérisation d'un réseau ferroviaire. S'il devait se faire que vous ne disposiez pas d'une locomotive à décodeur Delta (ou standard), voici l'excuse rêvée pour aller, sans attendre le mois de décembre, faire un tour dans le magasin de modélisme ferroviaire le plus proche.

## Bis repetita placent

On ne peut pas demander aux novices dans le domaine du modélisme ferroviaire de connaître tous les tenants et aboutissants de leur futur violon d'Ingres. Ceci nous amène partant à nous lancer dans un résumé concis de la substantifique moelle de la série d'articles consacrée au système EDITS, sans oublier de parler des développements les plus récents.

L'essence d'un pilotage numérique (par ordinateur) de modèles réduits ferroviaires est qu'il permet de commander, sur un même réseau, plusieurs trains et cela indépendamment l'un de l'autre. Contrairement à ce qui se passe dans le cas d'un réseau à pilotage traditionnel, où une action sur le rotacteur du transformateur se traduit par une mise en mouvement de « tout-ce-qui-peut-bouger », sur un réseau numérisé, chaque locomotive possède son

## Caractéristiques techniques de Digital Railrunner

- Se branche directement sur le port parallèle du PC.
- Programme confortable (Windows 3.1x, 95, 98, NT) permettant le pilotage d'un maximum de 15 trains indépendamment l'un de l'autre.
- Commande tant les décodeurs Märklin Digital H0 avec format de données Motorola classique que les décodeurs Delta.
- Booster compact intégré, courant de sortie de 3,5 A maximum et protection contre une surcharge
- Fonctionne avec le transformateur Märklin d'origine ou une tension alternative unique de 15 V.
- Mise en ou hors-fonction manuelle d'une fonction additionnelle.

propre régulateur (de vitesse) ce qui en permet un pilotage différencié. Cette approche numérique offre en outre toutes sortes de possibilités telles que la commande des aiguillages et de signaux au travers des rails, mais il s'agit là d'un aspect qui sort du cadre de cet article. Comment cela fonctionne-t-il ? Dans le cas d'un réseau traditionnel la voie véhicule tout simplement une tension (alternative chez Märklin, continue avec les autres marques), sa caractéristique de surtension (chez Märklin) ou de polarité (pour les autres marques) fournissant l'information de sens de circulation. Dans le cas d'un réseau commandé

par ordinateur, la voie véhicule un signal qui évolue entre 2 niveaux de tension positif et négatif définis, tension qui varie en fonction de l'échelle du réseau et qui, partant, peut aller de  $\pm 12$  à  $\pm 18$  V. Le rythme de commutation de ce signal fournit les informations de pilotage de chacune des locomotives voire les instructions de commande des autres attributs pilotables.

Comme dans bien d'autres domaines, l'industrie du ferromodélisme n'est pas arrivée non plus à se mettre d'accord sur un standard. Seuls 2 des 4 systèmes au départ, Märklin Digital H0, Lenz, Fleischmann FMZ et Selectrics, à savoir les 2 premiers, peuvent se targuer d'être très répandus et d'avoir ainsi établi, de fait, un quasi-standard propre. Signalons, à titre d'information, que le sys-

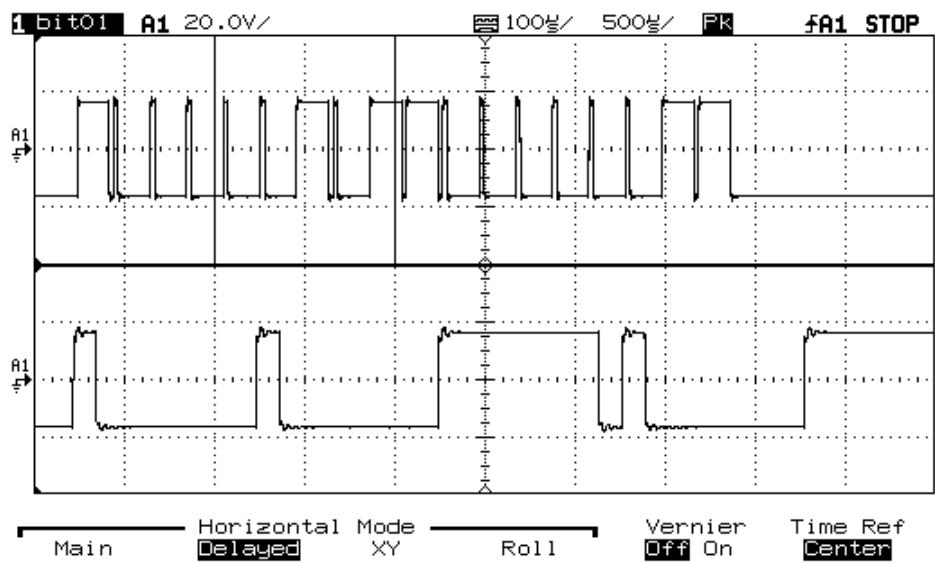


Figure 1. Exemple d'information numérique véhiculée par un train d'impulsions (trace du haut) telle qu'on le trouve sur la voie : adresse de locomotive 56 (X00X), bit de fonction actif (1), cran de vitesse 7 (0001). La trace du bas montre, sous une loupe, un zéro logique (00), un ouvert logique (10) et le début d'un un logique (1...).

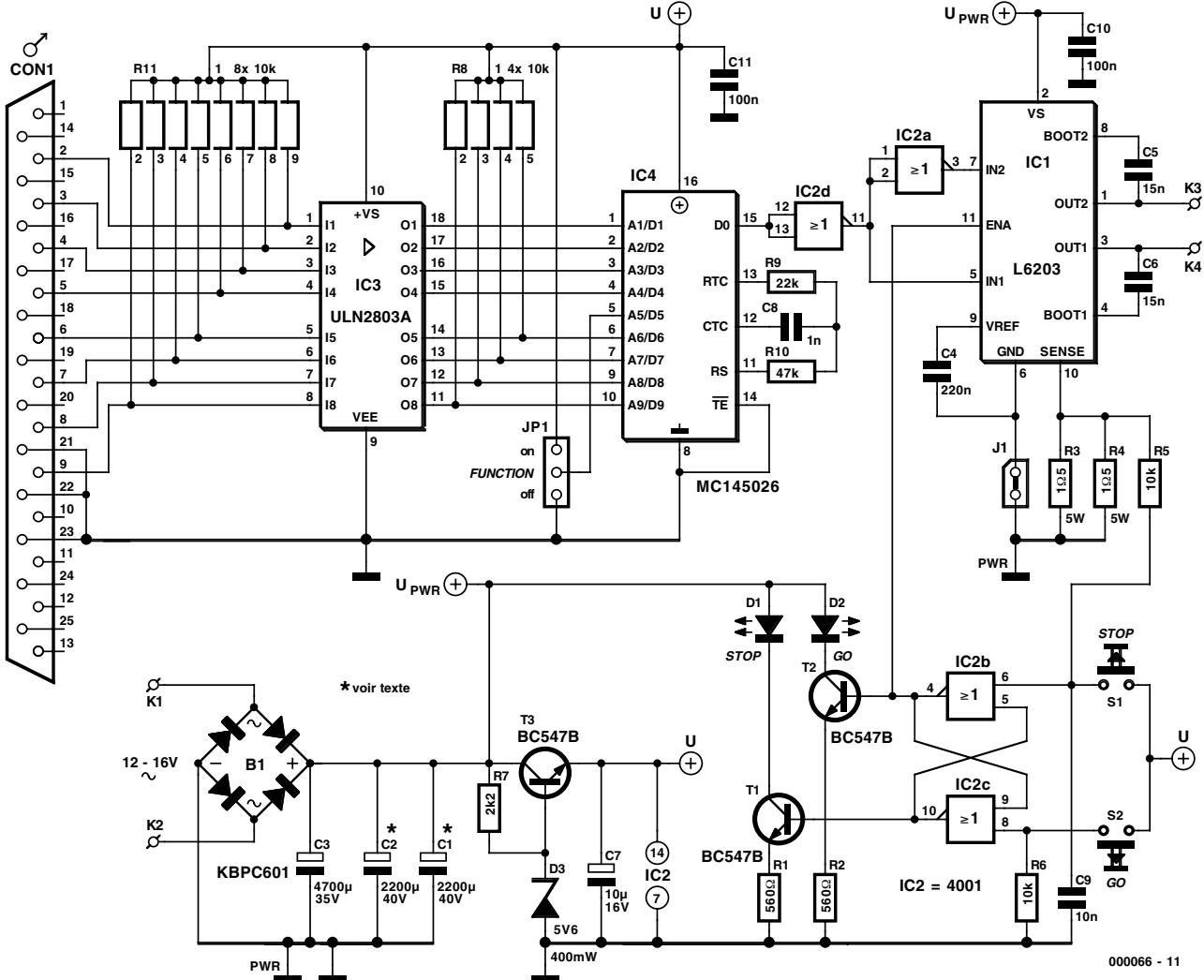


Figure 2. Schéma du circuit de pilotage de réseau ferroviaire.

tème Märklin Digital H0 connaît lui-même un certain nombre de « déclinaisons » (dont le format de donnée EDiTS et ce qu'ils ont appelé le « nouveau format de données Motorola »). Les autres marques elles aussi se déclinent en plusieurs variantes. Nous allons, dans cet article, nous en tenir au Märklin Digital H0, pour la simple et bonne raison que c'est le format de données utilisé par le système décrit ici.

Märklin Digital H0 repose sur une séquence de commutation définie à l'époque par Motorola qui pensait à ses télécommandes futures. L'information est empaquetée en trains sériels de 18 impulsions. À y regarder de plus près, ces impulsions se subdivisent en « wagons » constitués de 2 impulsions chacun. Chacune de ces paires d'impulsions peut ainsi prendre 1 des 3 possibilités de combinaison suivantes : 00 représente un zéro logique, 11 qui est un un logique et 10 qui correspond à un ouvert (open) logique. Le for-

mat de données Motorola d'origine ne connaissait pas la combinaison 01; notons que cela a cependant changé avec le nouveau format de données Motorola.

Un train d'impulsions comporte partant 9 paires d'impulsions. Les 4 premières servent à la définition des adresses (de locomotive) et offrent ainsi  $3^4$  combinaisons, soit 81 adresses différentes, Märklin n'en utilisant que 80. Les 5 paires restantes subissent un décodage purement binaire : 00 ou 11; le bit 5 indique l'état marche/arrêt de la fonction additionnelle, les bits 6 à 9 fournissent les instructions de vitesse et de basculement de sens de circulation.

Il y a quelque temps, Märklin a lancé ce qu'ils ont appelé le « nouveau format de données Motorola ».

Dans ce format, chacune des paires d'impulsions peut prendre 1 des 4 combinaisons logiques possibles (00, 01, 10 et 11). Les possibilités de combinaison additionnelles du bit de fonction et des 4 autres bits de données servent au transfert de l'information de sens de circulation non volatile et à des fonctions de commutation supplémentaires. Les encodeurs et décodeurs Motorola de la première heure ne savent que faire de ces combinaisons de paires d'impulsions.

Ajoutons, avant d'en avoir fini, que 2 trains de 18 impulsions sont séparés par une pause permettant une synchronisation de l'émetteur et du récepteur. Si chaque train d'impulsions dure quelque 3,8 ms, la longueur de chaque pause est de 2 ms environ. Motorola a prévu, comme

mesure de sécurité additionnelle, un mécanisme requérant du récepteur de recevoir 2 paquets identiques successifs avant qu'il ne puisse accepter l'information qu'ils véhiculent comme « vraie », algorithme qui s'avère, dans le cas d'un réseau fourmillent de convois, étonnamment efficace dans la pratique.

## L'électronique

Le schéma du circuit, représenté en **figure 2**, se distingue par son évidente simplicité. Le cœur en est IC4, un encodeur Motorola du type MC145026, circuit intégré chargé de la conversion des données fournies par le port parallèle du PC en un train série au format requis. L'utilisation de cet encodeur définit du même coup le cadre de travail de ce montage : il ne peut fournir que l'ancien format de données (le format classique). Il permet donc de piloter les décodeurs de locomotive standard et Delta de chez Märklin. Les décodeurs d'autres marques (ceux de Lenz par exemple), voire les décodeur disposant des 4 sorties de fonction additionnelles qui travaillent avec le nouveau format de données Motorola, sont inutilisables avec Digital Railrunner.

IC3, un réseau de transistors Darlington, constitue l'interface entre le port parallèle et l'encodeur. Les 4 premières sorties servent au paramétrage, au niveau de l'encodeur, de l'adresse de locomotive. Comme les sorties à collecteur ouvert ne sont pas dotées de résistances de forçage au niveau haut (*pull up*), l'état des lignes d'adresses est toujours, soit « bas » soit « à haute impédance ». On a besoin de cette possibilité de créer un état « à haute impédance » parce que les adresses Delta définies par Märklin possèdent toutes des bits « ouvert logique ».

Si nous avions connecté l'encodeur directement au port parallèle il ne nous aurait pas été possible de donner vie aux décodeurs Delta.

Le second quartet (*nibble*) du port parallèle sert au paramétrage des bits 6 à 9 de l'encodeur. Ces bits comportent l'information de vitesse et l'instruction de changement de sens de circulation : combinaison de bits 1000.

Le bit 5, le bit de fonction n'est ni

viande ni poisson. Ce bit est paramétrable par le biais d'un cavalier de court-circuit, JP1, étant partant soit à « 1 » soit à « 0 » pour la totalité des locomotives adressables. Ceci n'a pas de conséquence gênante sachant qu'il s'agit, la plupart du temps, de la fonction d'éclairage et que cette dernière est, de préférence, activée. Seul le premier décodeur de locomotive d'EDiTS utilisait le bit 5 (non-volatile) pour l'information de sens de circulation. Il n'est partant pas possible d'utiliser ce premier modèle de décodeur avec le présent montage sachant que toute modification au niveau de JP1 se traduirait immédiatement par un changement du sens de circulation pour la totalité des trains. On peut fort bien utiliser les versions plus récentes des décodeurs de locomotive pour EDiTS ainsi que celles destinées à EDiTS Pro.

R9, R10 et C8 servent à définir la chronologie de l'encodeur. R9 définit la durée d'un train de 18 impulsions (3,8 à 4 ms), R10 déterminant la durée des pauses de synchronisation séparant 2 trains d'impulsions.

À l'origine, ce montage a été conçu pour être connecté directement au booster d'EDiTS; cependant nombreux seront ceux qui hésiteront à effectuer cet accouplement vu le courant de sortie de 10 A de ce booster et son prix non négligeable. Ceci explique que nous ayons doté le présent montage de son propre mini-booster, IC1, doté de ses dispositifs de sécurité contre les courts-circuits et les surcharges, ainsi que d'une alimentation. Il est également possible, pour limiter les frais, d'y brancher le transfo Märklin standard. Le pont B1 associé aux condensateurs C1/C2 (ou C3, cf. paragraphe construction) produisent une unique tension continue lissée. Il nous faut faire appel, pour pouvoir disposer, à partir de cette unique tension, d'une tension de sortie commutant entre une valeur positive et une valeur négative, à un étage de sortie en plein pont. Le L6203 de STMicroelectronics intègre ces différents sous-ensembles et, de par sa technologie DMOS, n'a pas le moindre problème de vitesse de commutation ni de dissipation. Il reste, au courant de sortie maximal de 3,5 A choisi, relativement « cool ». Si le refroidissement

s'avérait insuffisant, un dispositif de protection anti-surchauffe interne se charge d'une coupure automatique lorsque ce composant atteint une température trop élevée. C5 et C6 sont ce que l'on appelle des condensateurs d'élévation artificielle de l'impédance (*bootstrap*); ils sont chargés d'établir une tension de grille suffisamment élevée pour les 2 FET-MOS de puissance présents dans la moitié supérieure du pont.

Le courant de sortie circule vers la masse en passant par la connexion de détection de courant (*sense*) et la paire de résistances R3/R4. La tâche de ces résistances est de surveiller le courant de sortie maximal, le L6203 n'étant pas réellement protégé contre les courts-circuits. La tension aux bornes de R3 et R4 est amenée, par le biais du réseau R5/C9 (filtre passe-bas destiné à éliminer les inévitables crêtes de commutation véhiculées par le courant), jusqu'à l'entrée de la porte NON-OU (NOR) IC2b. Associées à IC2c, cette porte constitue une bascule bistable (*flipflop*) en raison de l'utilisation détournée de IC2b en tant que comparateur analogique. Les circuits CMOS standard commutent gentiment aux alentours de la moitié de la tension d'alimentation. Lorsque la tension appliquée à la broche 6 de IC2b atteint 2,5 V (ce qui se passe à un courant de 2,5 V /  $0,75 \Omega = 3,5 \text{ A}$ ), IC2b et IC2c basculent. L'entrée de validation (*enable*) du L6203 passe au niveau bas, le pont déconnecte et, partant, la voie se trouve hors-courant. Simultanément, la LED verte « GO », D2, s'éteint et la LED rouge « STOP », D1, s'allume. Une action sur S2 permet de remettre le système sous tension. S1 permet pour sa part de forcer un arrêt du circuit.

Les 2 portes restantes de IC2 servent à appliquer au pont le signal numérique direct (non-inversé) et sa variante inversée.

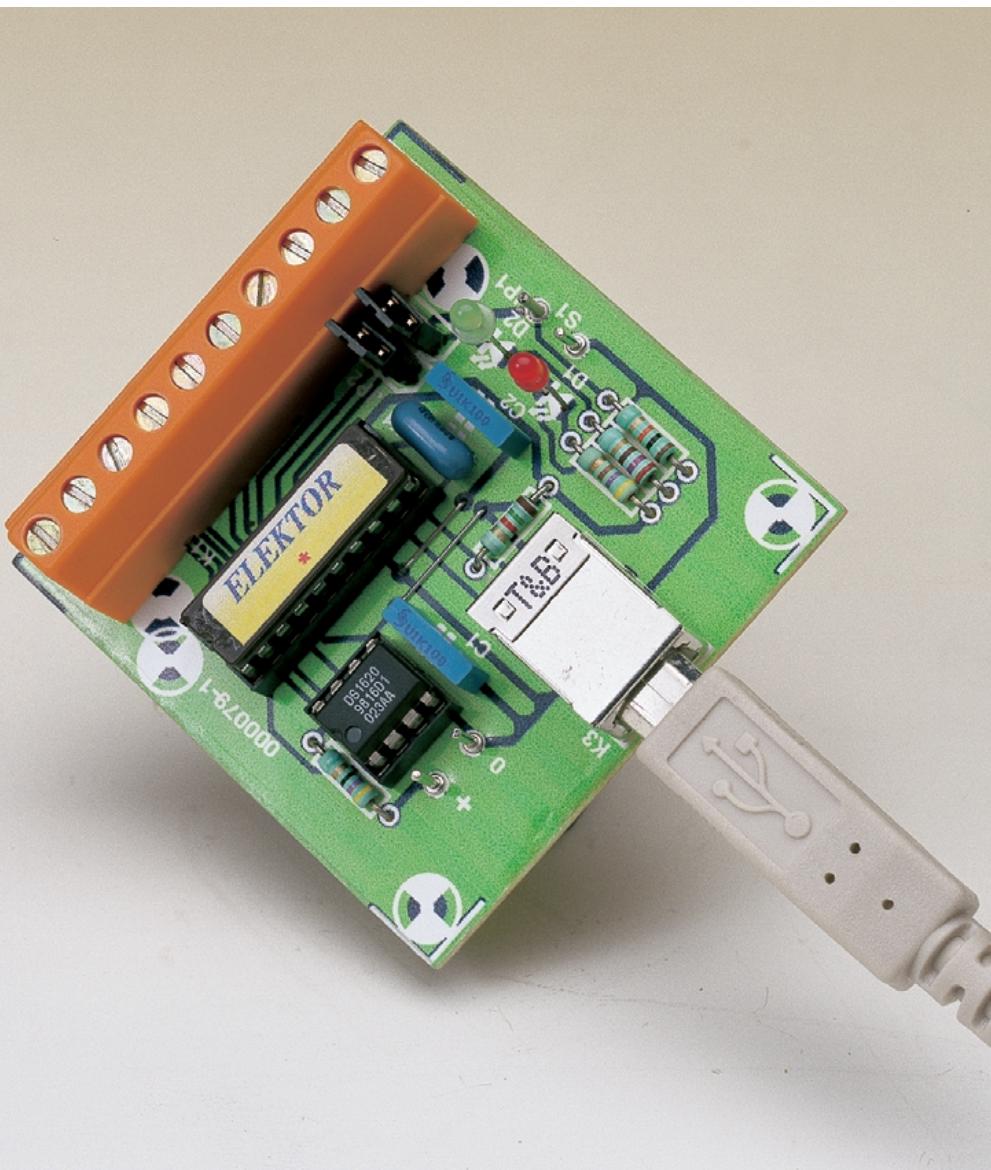
L'alimentation de 5 V requise par la logique est dérivée de l'alimentation de puissance. Nous faisons appel pour ce faire à une diode zener et à un tampon de sortie, T3, monté en émetteur-suiveur, sachant que les conditions de la tension d'entrée ne sont pas parfaitement définies. Si l'on utilise un transformateur Märklin par exemple et que l'on joue sur le rotacteur jusqu'à l'amener au point d'inversion du sens de circulation, on disposera d'une tension alternative comprise entre 24 et 30 V au point de connexion de l'alimentation. Après redressement, cela nous donne une tension d'entrée qu'un régulateur de tension du type 7805 est loin d'apprécier. De par la présence de D3 et T3, le circuit n'a pas de problème pour survivre à des conditions anormales aussi extrêmes. Le pont de sortie ne craint pas une tension d'un niveau aussi important (52 V maximum).

# Interface USB

## Essayez-vous à l'Universal Serial Bus

projet : Burkhard Kainka

L'acronyme USB se trouve, aujourd'hui, dans toutes les bouches. Cette nouvelle interface sérielle risque, à moyen terme, de se substituer à la plupart des interfaces actuelles d'un PC. Cela explique qu'Elektor ait décidé de mettre la main à la pâte.



Prenons le temps, avant de passer à la pratique, de donner quelques-unes des rares caractéristiques de base de l'USB (le *Universal Serial Bus*). Il existe, dans la première mouture de cette norme, la version USB-V1.1, les périphériques qui travaillent à faible vitesse (1,5 Mbit/s) et ceux qui l'utilisent à pleine vitesse (12 Mbit/s). La vitesse de transmission la plus faible dépasse déjà de beaucoup les taux de transmission usuels des interfaces sérieuses actuelles. L'un des problèmes majeurs que l'on rencontrait sur un PC tant soit peu « évolué » était une absence d'interface qui soit encore libre. L'USB présente l'indiscutable avantage de permettre, par la simple adjonction d'un répartiteur de bus (HUB), de disposer de 4 nouveaux ports. Il est, théoriquement, possible de connecter un maximum de 127 périphériques au total.

La connexion USB est en mesure de fournir la tension d'alimentation requise par les périphériques petits consommateurs. Elle peut fournir jusqu'à 100 mA sans le moindre problème. Il est même possible, à condition de mentionner spécifiquement ce besoin, de drainer 500 mA. Nombre d'applications de laboratoire typiques se contentent de beaucoup moins. L'absence de câble d'alimentation pour chacun des périphériques à alimenter est un excellent moyen de lutter contre les imbroglios

de câbles.

En principe, les périphériques USB peuvent être connectés au système (et également en être déconnectés) sans qu'il ne soit nécessaire de mettre le PC hors-tension. Le système d'exploitation charge automatiquement les pilotes requis. Cette capacité de « Plug-and-Play » avancée simplifie très sensiblement la mise en oeuvre simultanée de plusieurs périphériques. Un nouveau périphérique connecté au système se voit attribuer automatiquement un numéro de bus. Après interrogation du nouveau périphérique le système d'exploitation en reçoit les informations requises et charge automatiquement le pilote nécessaire.

De ce fait, tout périphérique USB possède, dès le départ, une certaine « intelligence » que n'a pas un périphérique connecté à l'interface RS-232. Il lui faut s'identifier auprès du système d'exploitation par le biais d'une série de tableaux descriptifs de manière à ce que ce soit bien le pilote adéquat qui soit choisi et chargé. Lors de la connexion initiale du périphérique le système d'exploitation demande la présentation (sur disquette ou CD-ROM) du pilote

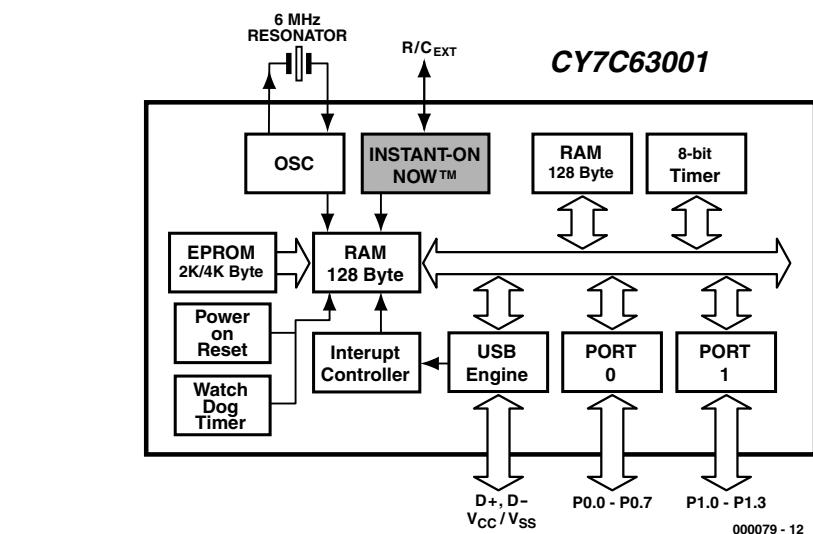


Figure 1. Structure interne du CY7C63001.

concerné. Après cette initialisation, toute nouvelle connexion du périphérique se traduira pour un chargement automatique du pilote sans que l'utilisateur n'ait plus à intervenir. Dès déconnexion du périphérique le pilote correspondant est automatiquement supprimé de la mémoire. La mise en oeuvre de l'interface USB

est, actuellement, malheureusement, limitée au système d'exploitation Windows 98. Si l'on dispose de Windows 95 version SR2, l'utilisation de périphériques USB est, en principe également possible, mais il ne faut pas s'attendre à un fonctionnement parfaitement fiable en toutes circonstances. Il est partant recommandé, si l'on envisage de travailler avec des périphériques USB, d'opter pour

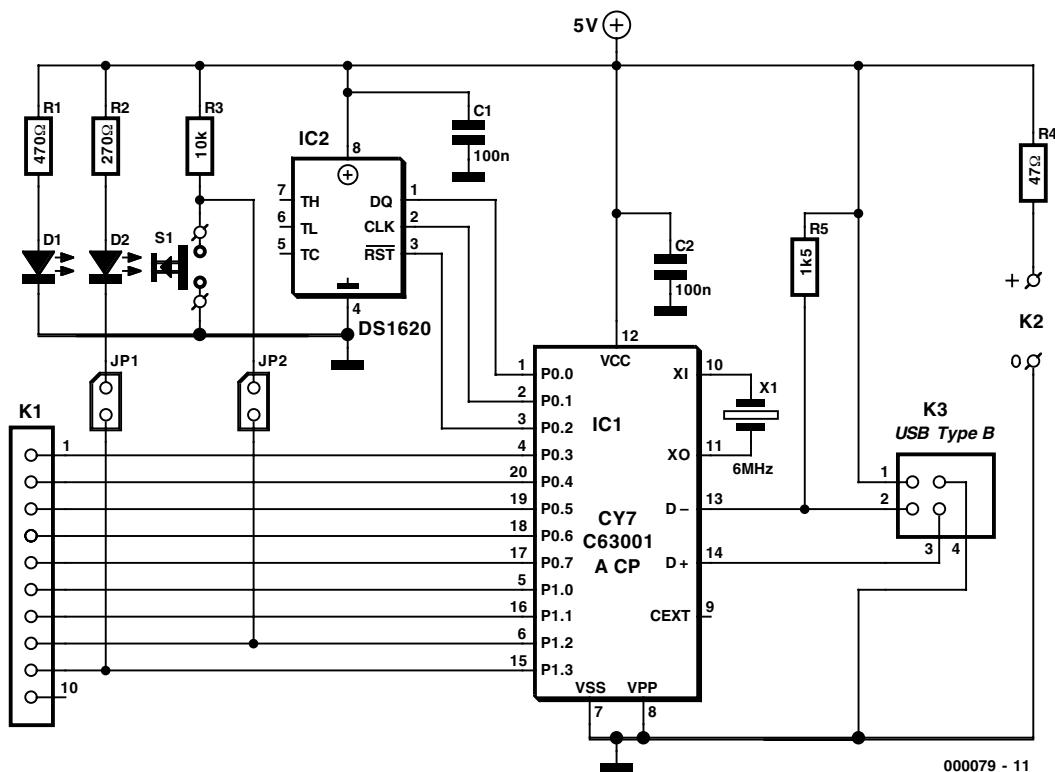


Figure 2. L'électronique de l'interface USB avec son capteur de température.

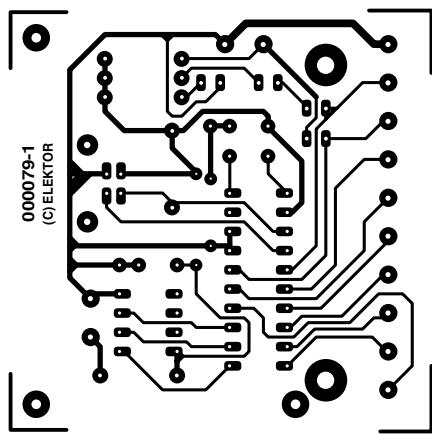
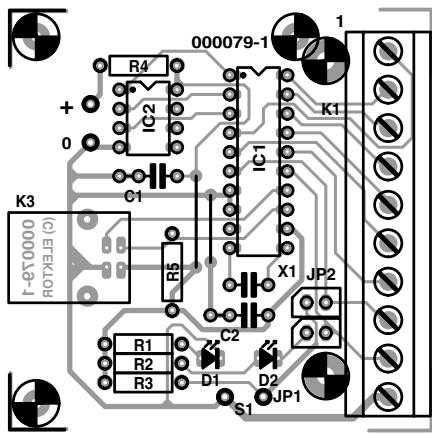


Figure 3. Cette platine servira de support à l'ensemble de l'électronique constituant notre interface USB.

Windows 98. Windows NT4 n'a pas la moindre idée de ce qu'est l'USB, Windows NT5 et Windows 2000, les systèmes d'exploitation les plus récents de Microsoft®, supportent eux l'interface USB. Signalons cependant que nous n'avons pas testé le fonctionnement des programmes accompagnant ce projet sous Windows 2000.

## L'électronique

L'interface USB fait appel à un microcontrôleur du type CY7C63000 de Cypress. Les fonctions premières dont fut chargé ce microcontrôleur à l'origine étaient les souris et autres ports de jeux; il intègre un moteur USB (*USB engine*) complet prévu pour le mode USB faible vitesse (*Lowspeed USB*). Les 2 lignes de données, D+ et D- attaquent le microcontrôleur directement.

La **figure 1** nous montre la structure interne de ce composant. Le microcontrôleur intègre,

## Liste des composants

### Résistances :

R1 = 470  $\Omega$   
R2 = 270  $\Omega$   
R3 = 10 k $\Omega$   
R4 = 47  $\Omega$   
R5 = 1k $\Omega$

### Condensateurs :

C1,C2 = 100 nF

### Semi-conducteurs :

D1 = LED rouge (haut rendement)  
D2 = LED verte (ordinaire)  
IC1 = CY7C63001ACP  
(programmé **EPS000079-41**)  
IC2 = DS1620 (Dallas Semiconductor)

### Divers :

JP1,JP2 = embase mâle à 2 contacts + cavalier  
K1 = bornier encartable à 1 rangée de 10 contacts (voire 2 borniers à 5 contacts)  
K2 = picot  
K3 = embase USB Type B (Farnell 153-503)  
S1 = bouton-poussoir unipolaire à contact travail  
X1 = résonateur céramique 6 MHz (Murata CSA6.00MG chez Farnell 295-292 ou Newport ZTA6.00MT)  
Boîtier : 61 · 22 · 80 mm (tel que Conrad 522848)

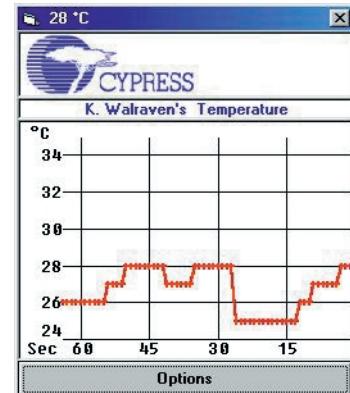


Figure 4. Recopie d'écran de la démonstration proposée à l'origine par Cypress.

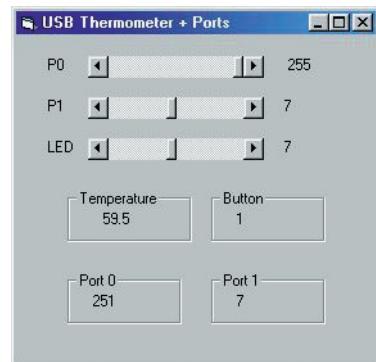


Figure 5. Le programme permet d'accéder à toutes les fonctions de port; il affiche en outre la température.

outre un processeur RISC, une ROM OTP (*One Time Programmable* = programmable une unique fois) pour le programme d'exploitation (*firmware*), 2 ports qui mettent à disposition, ensemble, 12 lignes, sans oublier de la RAM et un temporisateur (*timer*).

L'interface USB décrite dans le présent article repose sur une application de Cypress. Cette société propose, avec son Starter Kit CY3640, de réaliser un thermomètre à interface USB. La fonction première du kit est bien évidemment de faire connaissance avec le processeur et son environnement. Ce Starter Kit n'est malheureusement plus disponible actuellement sous sa forme originale, mais le présent projet publié dans Elektor lui donne une nouvelle vie. Il vous permettra de procéder à

vos premières expériences avec l'USB et ensuite d'utiliser cette interface USB pour nombre d'applications plus sérieuses.

L'interface USB possède les fonctions suivantes :

- Mesure de la température
- Interrogation d'une touche
- Réglage de la luminosité d'une LED
- Mise à disposition d'un total de 9 lignes de port d'E/S.

L'interface USB est compatible à 100% avec l'application de thermomètre présentée à l'origine par Cypress. On pourra partant utiliser tous les programmes d'exemple et implémenter le pilote de thermomètre. Dans l'application originale, il n'était cependant pas possible d'accéder à toutes les lignes de port. Pour cette raison le progiciel du

## Informations sur Internet

Microsoft VB5

[www.microsoft.com/msdownload/vbcce.htm](http://www.microsoft.com/msdownload/vbcce.htm)

Microcontrôleur CY7C63001ACP de Cypress

<http://www.cypress.com/cypress/prodgate/usb/cy7c63001.html>

Capteur de température de Dallas Semiconductor

<http://www.dalsemi.com/datasheets/pdfs/1620.pdf>

microcontrôleur a été modifié et doté de fonctionnalités additionnelles. L'électronique sur laquelle repose cette réalisation prend la forme du schéma représenté en figure 1. La mesure de la température est confiée à un circuit intégré spécialisé de Dallas Semiconductor, un DS1620; ce

composant requiert au total 3 lignes de port, P0.0 à P0.2. Les 9 autres lignes de port disponibles, P0.3 à P0.7 et P1.0 à P1.3, ont été amenées sur le bornier à griffe K1 pour une éventuelle utilisation. On pourra également, par le biais du cavalier JP1, connecter une LED verte ordinaire

(D2 ne sera pas partant du type faible courant) à la ligne de port P13. Cette LED visualise une énumération se faisant bien, sa luminosité pouvant prendre l'un des 16 niveaux disponibles. Par le biais de l'embase à cavalier JP2, le bouton-poussoir S1 se trouve relié à la ligne de port P1.2. Le bouton-poussoir fait basculer un registre interne du programme, registre que l'on peut interroger par le biais de l'interface USB.

Les 2 lignes de bus D+ et D- sont amenées à une embase USB du type B. S'il vous est impossible de mettre la main sur un connecteur de ce type, rien ne vous interdit de souder l'extrémité du câble USB directement aux points correspondants de la platine. L'interface USB dérive sa tension d'alimentation (+5 V et Masse) du bus. La seconde LED, D1, de couleur rouge sera elle du type à haut rendement (ou faible courant comme vous voulez), signalise la présence de la tension d'alimentation.

## Listage I. Le module USBI.BAS avec déclarations

```

Attribute VB_Name = "Module1"
Type SECURITY_ATTRIBUTES
nLength As Long
lpSecurityDescriptor As Long
bInheritHandle As Long
End Type

Type OVERLAPPED
Internal As Long
InternalHigh As Long
offset As Long
OffsetHigh As Long
hEvent As Long
End Type

Declare Function CreateFile Lib "kernel32" Alias "CreateFileA"
(ByVal lpFileName As String, ByVal dwDesiredAccess As Long,
ByVal dwShareMode As Long, lpSecurityAttributes As SECURITY_ATTRIBUTES,
ByVal dwCreationDisposition As Long, ByVal dwFlagsAndAttributes As Long,
ByVal hTemplateFile As Long) As Long
Declare Function DeviceIoControl Lib "kernel32" (ByVal hDevice As Long,
ByVal dwIoControlCode As Long, lpInBuffer As Any, ByVal nInBufferSize
As Long, lpOutBuffer As Any, ByVal nOutBufferSize As Long,
lpBytesReturned As Long, lpOverlapped As OVERLAPPED) As Long
Declare Function CloseHandle Lib "kernel32" (ByVal hObject As Long) As Long

Public Security As SECURITY_ATTRIBUTES
Public gOverlapped As OVERLAPPED
Public hgDrvrHnd As Long
Public Const GENERIC_READ = &H80000000
Public Const GENERIC_WRITE = &H40000000
Public Const FILE_SHARE_WRITE = &H2
Public Const FILE_SHARE_READ = &H1
Public Const OPEN_EXISTING = &H3

```

## Listage 2. Accès à tous les ports et fonctions

```

Dim sFileName As String
Dim htemp As Long
Dim lIn As Long, lInSize As Long, lOut As Long, lOut-
Size As Long, lSize As Long
Dim lTemp As Long
Sub USB_IO()
    sFileName = "\.\Thermometer_0"
    hgDrvrHnd = CreateFile(sFileName, GENERIC_WRITE Or
        GENERIC_READ, FILE_SHARE_WRITE Or FILE_SHARE_READ,
        Security, OPEN_EXISTING, 0, 0)
    lTemp = DeviceIoControl(hgDrvrHnd, 4&, lIn, lInSize,
        lOut, lOutSize, lSize, gOverlapped)
    htemp = CloseHandle(hgDrvrHnd)
End Sub

Sub Brightness(Level)
    lIn = Level * 256 + 14
    lInSize = 2
    lOutSize = 1
    USB_IO
End Sub

Function RdPort(Port) As Integer
    lIn = Port * 256 + 20
    lInSize = 2
    lOutSize = 2
    USB_IO
    RdPort = (lOut / 256) And 255
End Function

Sub WrRAM(Adresse, Wert)
    lIn = 65536 * Wert + Adresse * 256 + 23
    lInSize = 3
    lOutSize = 1
    USB_IO
End Sub

```

```

    USB_IO
End Sub

Private Sub HScroll1_Change()
    Wert = HScroll1.Value * 8 + 7
    WrRAM 46, Wert
    Label4.Caption = Str$(Wert)
End Sub

Private Sub HScroll2_Change()
    WrRAM 47, HScroll2.Value
    Label5.Caption = HScroll2.Value
End Sub

Private Sub HScroll3_Change()
    Brightness HScroll3.Value
    Label6 = HScroll3.Value
End Sub

Private Sub Timer1_Timer()
    lIn = 11
    lInSize = 1
    lOutSize = 3
    USB_IO
    Temp = ((lOut \ 256) And 255) / 2
    Minus = (lOut \ 65536) And 255
    If Minus > 0 Then Temp = Temp * -1
    Button = (lOut \ 16777216) And 255
    Label7.Caption = Str$(Temp)
    Label8.Caption = Str$(Button)
    Label9.Caption = Str$(RdPort(0))
    Label10.Caption = Str$(RdPort(1)) And 15
End Sub

```

mentation. Cette tension est également disponible aux picots + et 0 pour d'éventuelles applications. En tout état de cause, une résistance de  $47\ \Omega$ , R4, limite, même en cas de court-circuit, à 100 mA l'intensité maximale

pouvant circuler au niveau de l'embase USB.  
Le microcontrôleur dérive sa fréquence d'horloge d'un résonateur céramique travaillant à 6 MHz pris

entre 2 (et non pas 3) de ses broches. Les condensateurs requis au fonctionnement de ce composant y sont déjà intégrés.

## Fonctions

0Bh: Read Thermometer

  lIn: 0Bh (longueur : 1 octet)  
  iOut: Button, Signe, Température, État (longueur : 4 octets)

0Eh: Set LED Brightness

  lIn: luminosité (0 à 15), 0Eh (longueur : 2 octets)  
  iOut: Status (longueur : 1 octet)

14h: Read Port

  lIn: Port (0,1), 14h (longueur : 2 octets)  
  iOut: Valeur, État (longueur : 2 octets)

15h: Write Port

  lIn: Valeur (0 à 255), Port (0,1), 15h (longueur : 3 octets)  
  iOut: État (longueur : 1 octet)

16h: Read RAM

  lIn: Adresse (0 à 255), 16h (longueur : 2 octets)  
  iOut: valeur, État (longueur : 2 octets)

17h: Write RAM

  lIn: Valeur (0 à 255), Adresse (0 à 255), 17h (longueur : 3 octets)  
  iOut: État (longueur : 1 octet)

18h: Read ROM

  lIn: Index, Adresse (0 à 255), 18h (longueur : 3 octets)  
  iOut: Valeur, État (longueur : 2 octets)

## Contenu de la disquette EPS 000079-II

CypressSemiconductorsCYPRESS.INF	Infos d'installation des pilotes USB
Thermometer.exe	Programme de démonstration de Cypress
USB 20e.ASM	Listing en assembleur pour le microcontrôleur
Usb 20e.hex	Fichier HEX pour le microcontrôleur
Usb 20e.lst	Code-source pour le microcontrôleur
USBTherm.sys	Pilote USB
USBelektor.vbp	Projet Visual Basic
USBelektor.frm	Fichier Format Visual Basic
usb1.bas	Module Visual Basic
copyright.txt	Texte de copyright
contents.txt	Contenu de la disquette

### Réalisation et essais

La mise en place des composants sur la platine dont nous vous présentons les 2 faces en **figure 3**, ne devrait pas poser le moindre problème. Il n'y a pas la moindre contre-indication à utiliser des supports (de bonne qualité) pour les 2 circuits intégrés. Une fois l'étape de réalisation terminée il reste à s'assurer de l'absence d'erreur d'implantation grossière, c'est-à-dire visible au premier coup d'oeil avant de pouvoir passer au moment crucial. Connectez l'interface USB au PC par le biais d'un câble USB de type A-B. Quelques instants plus tard Windows devrait se manifester et signaler la détection de la connexion au système d'un nouveau matériel. Un message demande alors la présentation d'une disquette comportant le pilote correspondant. Répondez-y en introduisant dans le lecteur correspondant (A: en règle générale) la disquette (**EPS000079-11**) disponible auprès des adresses habituelles requise. Le système y trouvera le fichier *CypressSemiconductorsCYPRESS.INF* donnant la description du périphérique et le pilote requis. Il procédera ensuite au chargement du pilote proprement dit, *USBTherm.sys*. Dès que ce processus sera terminé on aura énumération du périphérique, c'est-à-dire son annonce au système. On verra alors s'allumer non seulement la LED rouge, si tant est que le cavalier JP1 soit mis en place, mais également la LED verte.

Le programme de démonstration *Thermometer.exe* de Cypress égale-

ment présent sur la disquette permet de réaliser un thermomètre à visualisation graphique de l'évolution de la température. L'utilisateur peut, par action sur le bouton-poussoir présent sur la platine, passer, à condition que le cavalier JP2 soit implanté, d'un affichage en degrés Celsius à un affichage en degrés Fahrenheit. Il est en outre possible de jouer sur la luminosité de la LED verte. Le microcontrôleur comporte à cet effet un convertisseur numérique/analogique à 4 bits permettant d'ajuster la valeur du courant drainé au niveau des sorties de port P1. Le logiciel permet uniquement de jouer sur le port P1.3. On voit en **figure 4** une recopie d'écran du thermomètre.

### Programmation en Visual BASIC

Une interface ne présente un intérêt réel que lorsqu'il est également possible de la piloter par le biais de programmes que l'on aura écrit soi-même. L'accès au périphérique se fait à l'aide d'un petit programme en Visual BASIC. Nous avons utilisé VB5CCE, un programme mis gratuitement au téléchargement par Microsoft®, logiciel parfaitement fonctionnel, mais n'étant cependant pas en état de produire des applications autonomes (*stand-alone*). Après installation du set de programmes, on ouvre le projet présent sur le lecteur de disquette A; on sélecte et ouvre ensuite le fichier USBelektor.VSP. Le projet comprend le module USB1.BAS (cf. la liste du contenu de la disquette dans l'enca-

dré) qui comporte un certain nombre de déclarations et le fichier de format USBElektor.VPB, responsable non seulement du pilotage graphique du moniteur mais aussi du déroulement correct de l'application (communication avec le contrôleur de Cypress, traitement de données). le programme est lancé par une action sur l'option de menu « RUN ».

Il vous faudra, pour pouvoir accéder au pilote SYS, des fonctions de Windows suivantes : CreateFile, CloseHandle et DeviceIoControl. Elles sont déclarées dans le cas présent dans le module USB1.BAS.

Le fichier de format USBElektor.FRM (Listage 2) permet l'accès à toutes les fonctions de port du microcontrôleur. Il faudrait en outre conserver toutes les fonctions de thermomètre d'origine. Nous avons besoin, pour la commande du pilote, d'un certain nombre d'informations que nous passerons en revue un peu plus loin. Ce programme génère l'écran représenté en **figure 5**.

Lors de l'appel du pilote par le biais de l'instruction DeviceIoControl il faut indiquer un numéro de fonction de pilote, un tampon d'entrée IIn et un tampon de sortie IOuT. Si le numéro de fonction de pilote reste imperturbablement à 4, le choix de l'action proprement dite se fait par le biais d'un octet de commande de IIn. Le pilote répond, en fonction de la fonction concernée, en fournissant entre 2 et 4 octets. L'octet de poids faible comporte à chaque fois une information d'état du résultat de l'accès. Il existe, outre la lecture du thermomètre, d'autres fonctions spécifiques servant au réglage de la luminosité de la LED et à l'accès à la mémoire et aux ports du microcontrôleur. L'encadré « Fonctions » vous en donne un aperçu.

La fonction Write Port ne travaille, avec le pilote disponible, que partiellement. L'auteur a partant essayé, tout en ne modifiant pas le pilote, de trouver une solution qui tout en se limitant uniquement à une adaptation du progiciel du microcontrôleur permettrait toutes les tâches ayant trait au port. La solution à ce problème a pris la forme d'une utilisation de l'instruction WriteRAM. Il fut défini 2 variables RAM additionnelles :

Fonctions du port 0 : 2Eh

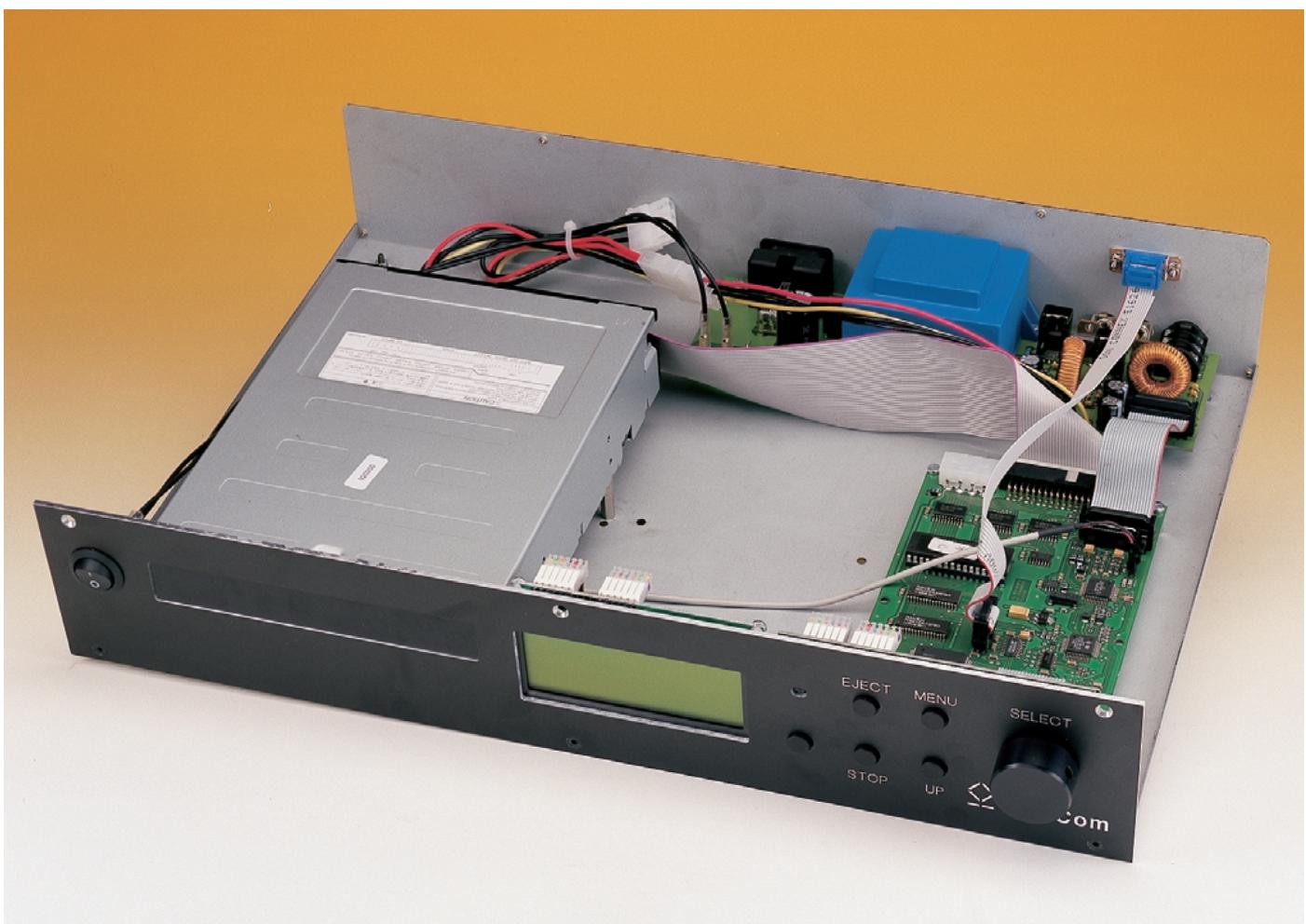
Fonctions du port 1 : 2Fh

Il faut à partir de là écrire directement les données de port dans certaines des adresses de mémoire de RAM du microcontrôleur. Le progiciel actualise ensuite les sorties de port. Le programme d'exploitation se trouve d'ailleurs dans le fichier *USB 20e.asm* présent sur la disquette évoquée plus haut.

# OSCAR

## Lecteur MP3 autonome

Bureau d'ingénierie A. Kurpiers & V. Pantelic



Notre plan de bataille était établi dans le moindre détail : le projet, « préparé » par un article de base consacré au thème MP3 et publié dans le numéro de mai, à savoir un lecteur MP3 autonome, verrait sa première partie décrite dans le numéro de juin pour être fin prêt dans la présente édition de notre magazine. Mais les choses ne se passent pas toujours comme prévu. Nous nous vîmes confrontés, avec le MP3-Cube, à des problèmes techniques imprévus que le laboratoire d'Elektor n'était malheureusement pas en mesure de résoudre dans les délais requis.

Nous avons heureusement trouvé une (meilleure) alternative de lecteur MP3 sous la forme d'un projet mis au point par des étudiants de l'Université de Technologie de Darmstadt (RFA) baptisé OSCAR (*Optical Storage Compressed Audio Replay*). L'encadré « Historique » récapitule les repères intéressants du développement de cet appareil. OSCAR fait appel au même set de puces que celui que devait utiliser MP3-Cube, mais son intelligence repose sur un microcontrôleur notablement plus puissant. Cela se justifie sachant que les caractéristiques techniques d'OSCAR sont de bien loin plus intéressantes que celles du MP3-Cube. Fin des lamentations !

Les différences les plus notables sont des possibilités, d'une part, de connexion d'un disque dur IDE, de sélection de morceaux spécifiques par le biais d'un encodeur rotatif d'autre part sans oublier une compatibilité aux normes MP2 et S/PDIF. L'encadré récapitule les caractéristiques techniques les plus marquantes de ce lecteur MP3.

## Démonstration de force

OSCAR est un lecteur autonome (ne requérant partant pas de PC) de fichiers de musique codés en un format on ne peut plus populaire de nos jours, à savoir MP3, fichiers stockés soit sur un CD-ROM acheté tel quel ou gravé soi-même à partir de CD Audio que l'on possède, CD-ROM dont le format de fichiers respecte la norme ISO9660, soit encore sur un disque dur formaté selon le système d'exploitation Windows FAT16/FAT32.

Cet appareil est également en mesure de lire des CD Audio classiques respectant le standard S/PDIF. Comparé à un lecteur de CD courant, OSCAR peut se targuer de nombre de possibilités additionnelles. Il est ainsi en mesure de

## Caractéristiques techniques :

- Lecteur MPEG Audio complet et autonome (pouvant partant se passer de PC)
- Lit les fichiers MPEG1/2 Layer 2/3 Audio (mp2, mp3) ainsi que les CD Audio placés dans le lecteur de CD-ROM ou transférés sur le disque dur
- Reproduit tous les mp2 et mp3 à tous les taux d'échantillonnage, de 8 à 320 Kbits/s ainsi que les fichiers VBR
- Comporte une sortie Ligne (Line-out) et une sortie pour casque d'écoute
- Supporte ISO9660 l'extension Joliet y comprise et peut lire et écrire sur FAT16 ou FAT32
- Un affichage LCD à 4 lignes visualise les noms de répertoires et de fichiers ainsi que les repères ID3.
- Un encodeur rotatif permet un accès rapide à l'ensemble des fichiers de musique
- Utilise un processeur RISC 32 bits puissant de Hitachi, le SH7032
- Système d'exploitation sophistiqué disposant de nombreuses fonctions centrées sur MP3
- La présence d'un BIOS en Flash permet une réactualisation aisée du progiciel
- Interface sérielle pour la communication avec un PC voir la télécommande de l'appareil par de dernier
- Platine de format Europe, tension d'alimentation de 5 et 12 V.

visualiser, sous forme de texte, le nom de l'interprète et le titre d'un morceau donné voire encore, si tant est qu'il existe, le repère ID3. Il est ainsi possible de reproduire un morceau donné, un sous-répertoire

sélectionné, voire l'ensemble du CD-ROM ou du disque dur, en mode séquentiel, bouclé, ou encore aléatoire. La succession des morceaux pourra prendre la forme d'un fichier liste du type m3u (Winamp) présent sur le CD ou le disque dur, mais on pourra également créer

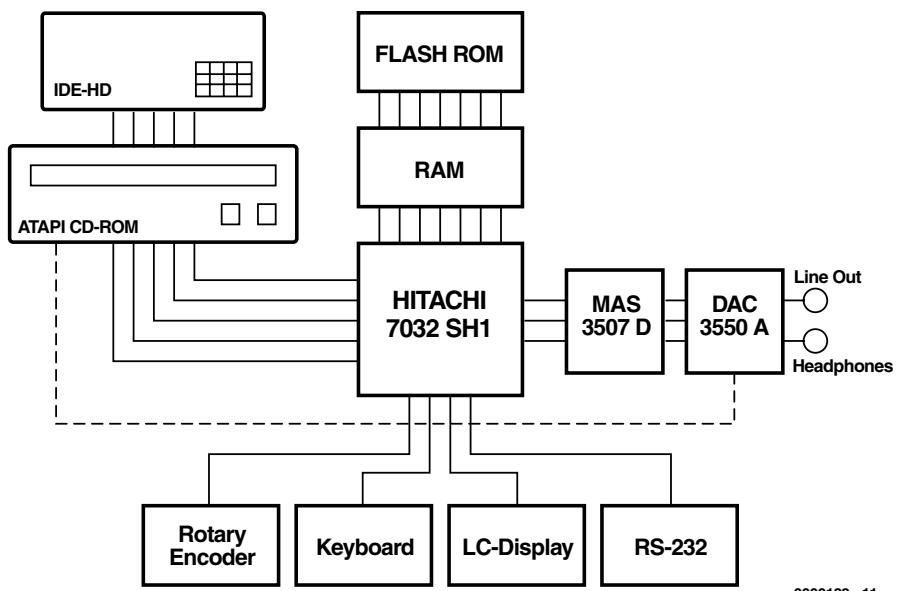
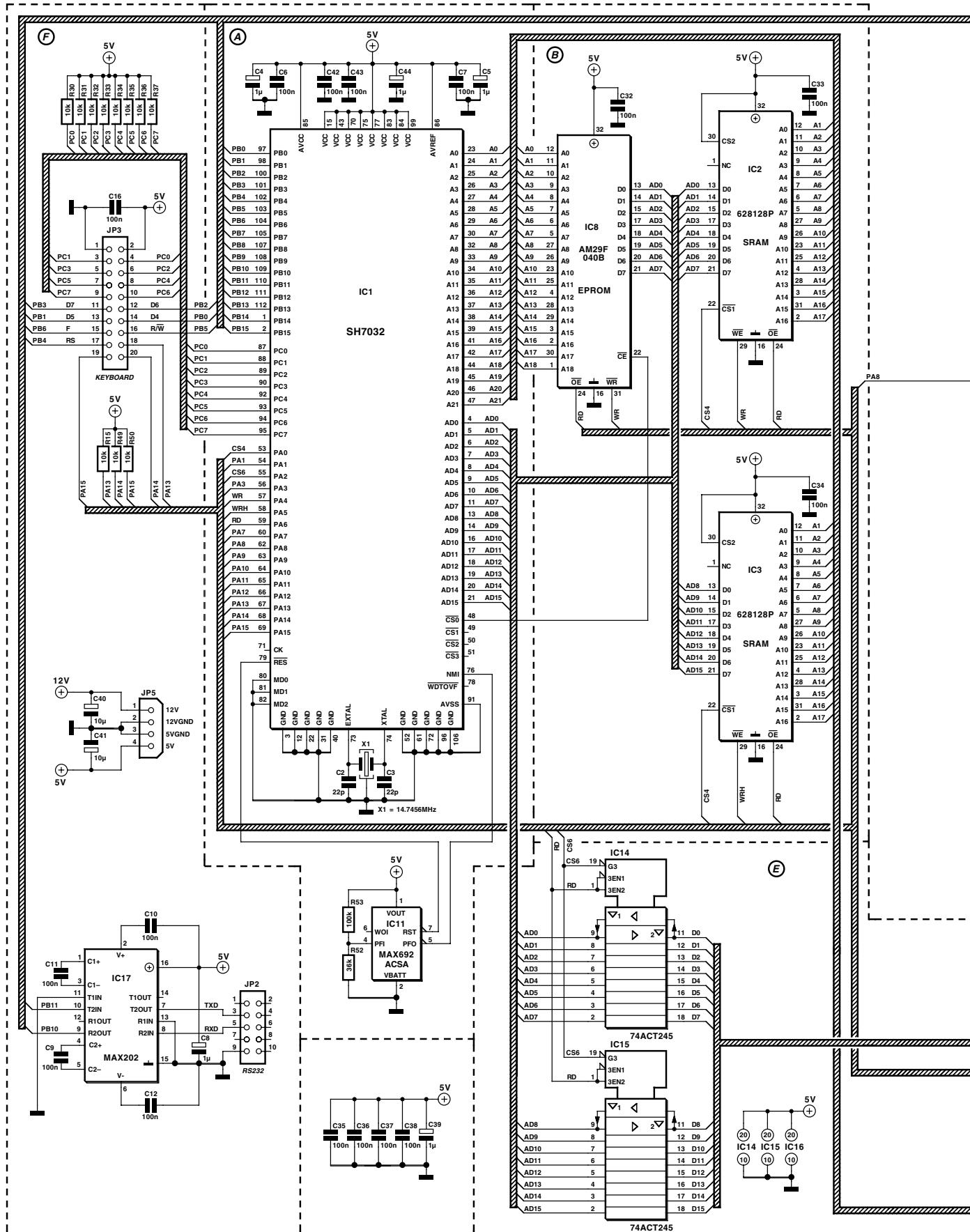


Figure 1. Synoptique du lecteur MP3.



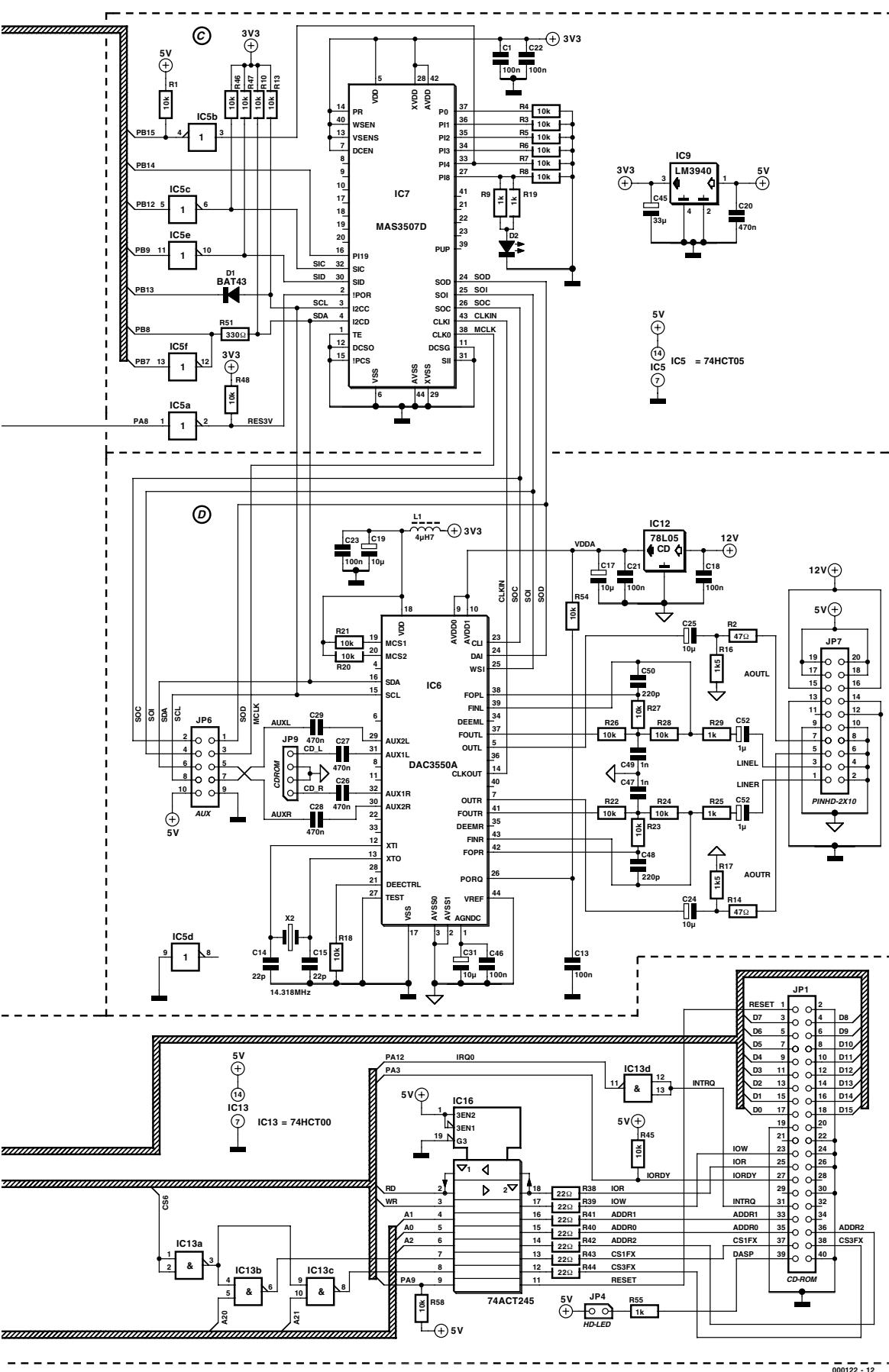


Figure 2.

- Le pilote aux commandes : un microcontrôleur 16 bits puissant de Hitachi, le SH7032.
- Extension de mémoire à base de ROM Flash et de SRAM.
- Le décodeur MP3 avec bus I<sup>2</sup>C vers le microcontrôleur et bus I<sup>2</sup>S vers le CNA.
- Le CNA de qualité professionnelle met à disposition une sortie Ligne et une sortie pour casque d'écoute.
- Matériel de l'interface IDE reliant le contrôleur au lecteur de CD-ROM et au disque dur.
- Convertisseur d'interface RS-232 et câble de liaison vers la face avant.

cette liste soi-même sur l'appareil, quitte à la sauvegarder sur le disque dur si tant est que l'appareil en soit doté.

On pourra, selon le cas, charger les fichiers sur le disque dur en connectant ce dernier à un PC pour ensuite le monter dans OSCAR, soit recopier à demeure morceau par morceau, voire des sous-répertoires complets du lecteur de CD-ROM vers le disque dur. On a, une fois que le disque dur est monté dans OSCAR, la possibilité de le formater, de le doter de répertoires ou encore d'en effacer certains, voire tous.

Une fonction de « mise en sommeil » (*sleep-timer*) permet une mise hors-fonction de l'appareil après une durée de fonctionnement prédéterminée, une fonction d'automate (*autoplay*) démarrant automatiquement la reproduction des morceaux dès mise sous tension d'OSCAR.

Un encodeur rotatif doté d'une fonction de bouton-poussoir épaulé par 5 touches additionnelles permettent à eux tous une commande très confortable de l'appareil. Le choix du numéro se fait par rotation de l'encodeur, une action sur la touche en lançant la lecture. Une structure de menu arborescent bien claire permet de choisir chacun des modes de fonctionnement disponibles et de les mémo-riser. Le progiciel au coeur d'OSCAR est en permanente extension et amélioration; on pourra, le moment venu, le réactualiser à l'aide d'un CD prévu à cet effet (*update*) ou encore utiliser pour cela l'interface sérielle dont dispose cet appareil.

L'article théorique consacré à MP3 et aux différents sets de puces prévus pour le décodage des fichiers MP3 publié dans le numéro de mai dernier aura au moins eu un avantage, celui de nous permettre de nous limiter ici à l'aspect électronique de cette réalisation. Un coup d'œil au synoptique de la **figure 1** nous permet de constater que le set de puces battant au coeur d'OSCAR est la fameuse paire MAS3507D/DAC3550 de Micronas Interme-

## Tableau I. Fonctions des ports.

Port	Fonction	Fonction alternative
PA0	Chip select SRAM	CS4
PA1	–	CS5
PA2	Chip select Interface IDE	CS6
PA3	WAIT to generate wait states	CS7
PA4	Memory write	WR
PA5	Lower byte select for 16bit wide memory LBS	
PA6	Memory read	RD
PA8	Reset MPEG decoder	
PA9	Reset IDE devices	
PA10	RTC DS1307 data	
PA11	RTC DS1307 clock	
PA12	IRQ interface IDE	IRQ0
PA13	IR detection	IRQ1
PA14	réservé (câblé à JP3)	IRQ2
PA15	réservé (câblé à JP3)	IRQ3
PB0	LCD	D4
PB1	LCD	D5
PB2	LCD	D6
PB3	LCD	D7
PB4	LCD	RS
PB5	LCD	R/W
PB6	LCD	E
PB7	I <sup>2</sup> C data out	
PB8	I <sup>2</sup> C data in	
PB9	Serial data to MPEG decoder	TxD0
PB10	RxD RS232	RxD1
PB11	TxD RS232	TxD1
PB12	Serial clock to MPEG decoder	SCK0
PB13	I <sup>2</sup> C clock	
PB14	IRQ demand from MPEG decoder	IRQ6
PB15	IRQ frame from MPRG decoder	IRQ7
PC0 à 7	Clavier	Entrées AC

tall. Le contrôleur adresse, outre de la RAM externe, également une ROM Flash; il pilote toute une série de fonctions périphériques telles que commande, affichage et interfaces. Il est en mesure de piloter tout aussi

bien le lecteur de CD-ROM que le disque dur.

### Au coeur : un contrôleur d'Hitachi

## Tableau 2. Set de registres IDE

/CS1FX	/CS3FX	A2	A1	A0	Accès lecture	Accès écriture
0	1	1	1	0	Alternate Status	Device Control
1	0	0	0	0	Data (16bit)	
1	0	0	0	1	Error	Features
1	0	0	1	0	Sector Count	
1	0	0	1	1	Sector Number / LBA Bits 0-7	
1	0	1	0	0	Cylinder Low / LBA Bits 8-15	
1	0	1	0	1	Cylinder High / LBA Bits 16-23	
1	0	1	1	0	Device/Head / LBA Bits 24-72	
1	0	1	1	0	Status	Command

## Historique

Le premier prototype d'OSCAR a vu le jour dans le cadre d'un projet d'étude réalisé à l'Institut des technologies de l'information au département électrotechnique de l'Université Technique de Darmstadt. Étant à la recherche de projets d'étude intéressants, 2 étudiants s'attaquèrent à la réalisation d'un lecteur de CD MP3. Après 6 mois de travail acharné naquit le premier OSCAR (qui est, nous le disions dans l'article, l'accronyme de **O**ptical **S**torage **C**ompressed **A**udio **R**eplay), appareil dont on peut voir une photographie à l'adresse Internet suivante : <http://www.oscar-mp3.com/pic.html>.

Ce premier prototype était lui aussi déjà en mesure d'afficher le titre des morceaux sous la forme de texte, mais ressemblait encore beaucoup à un lecteur de CD courant. OSCAR subit ensuite une cure de rajeunissement complète et vit sa fonction disque dur très améliorée. C'est ainsi que le nouvel OSCAR vit le jour, ce nouveau modèle étant doté, outre d'un environnement utilisateur rénové bien mieux adapté à l'environnement MP3, de nombre de nouvelles fonctions et d'améliorations. Depuis lors, OSCAR a fait son petit bonhomme de chemin commercial puisqu'il est disponible sous la forme d'un appareil tout monté ou sous celle d'un kit comprenant les platines dotées de leurs composants respectifs.

De par les fonctions multiples à assurer, le lecteur MP3 est piloté par un processeur très puissant tournant à la fréquence élevée de 20 MHz et de prix très abordable; il s'agit du contrôleur RISC 32 bits SH7032 de la famille SH1 de Hitachi (**figure 2A**). Ce processeur dispose d'un nombre très important de broches d'entrées/sorties utilisables très librement et de toutes sortes de blocs périphériques intégrés tels qu'interfaces sérieuses (SCI), temporisateur (timer), contrôleur d'interruption (INTC = *INTerrupt Controller*), contrôleur de DMA (DMAC), sans oublier un convertisseur analogique/numérique (CAN = ADC pour *Analogic/Digital Converter* en anglais). Il dispose en outre d'un domaine d'adressage très étendu (plus qu'il n'en faut à OSCAR) et d'un set d'instructions puissant.

Le contrôleur dispose d'un port à 8 bits, PC0 à PC7, et de 2 ports d'une largeur de 16 bits, PA0 à PA15 et PB0 à PB15. Les lignes des ports de forte largeur peuvent servir tant d'entrée que de sortie, le port PC0 à PC7 ne pouvant servir lui que d'entrée seu-

lement.

Les lignes PC0 à PC7 attaquent le convertisseur CAN, de sorte que l'on dispose de 8 entrées analogiques. Certaines de broches remplissent des fonctions spéciales; elles peuvent, par exemple, servir de sources d'interruption ou encore permettre la sélection de groupes de fonctions périphériques telles que clavier, affichage LCD, décodeur MP3, horloge en temps réel et bien d'autres. Ces fonctions récapitulées dans le **tableau 1** se voient attribuer des lignes de port lors de la mise sous tension du lecteur MP3.

Le processeur dispose de 8 domaines de mémoire indépendants où il peut adresser jusqu'à un maximum de 16 Moctets, domaines qui peuvent également prendre la forme de RAM dynamique (DRAM). Le contrôleur proprement dit dispose de 4 Koctets (7034) voire 8 Koctets (7032) de RAM interne. Le bus de données d'une largeur de 16 bits du contrôleur attaque 256 Koctets de RAM statique (une paire de circuits de SRAM de 128Kx8 du type 62812P à la vitesse d'accès de 15 ns) occu-

pant le domaine d'adresses allant de 0400000H à 05FFFFH (ce H en suffixe indiquant une adresse hexadécimale) ainsi qu'une ROM Flash de 512 Koctets (AM29F040B d'AMD ou AT29C040 d'Atmel), cette ROM occupant les adresses allant de 0000000H à 0FFFFFFH (cf. **figure 2B**). Le système d'exploitation du lecteur se trouve en ROM Flash, certains de ses composants étant recopier, lors de la mise en fonction de l'appareil, dans la SRAM plus rapide. La ROM Flash comporte en outre un petit programme moniteur qui permet une réactualisation in situ (dans l'appareil même) du logiciel principal. On pourra également utiliser cette mémoire pour la sauvegarde de données que l'on ne veut pas perdre lors de la coupure de l'alimentation, telles que, par exemple, les listes de morceaux voire des bandes annonces (*jingle*) de courte durée.

L'horloge-système est fournie par la combinaison oscillateur interne + quartz externe de 14,745 6 MHz. Cette valeur permet un taux de transmission série pouvant aller jusqu'à 230,4 kbits/s.

## Décodeur MP3

Le MAS3507D, dont on retrouve la structure interne en **figure 2C**, a été développé, à l'origine, pour le projet « *Worldspace Satellite Radio* ». Ceci explique qu'il soit en mesure de se synchroniser sur le taux de données d'un flux de bits MP3 prédéterminé.

Au cœur d'OSCAR il est mis à contribution dans un mode de fonctionnement différent, à savoir le mode multimédia. Le décodeur génère de façon autonome l'horloge de bits MP3 et demande, par le biais du signal DEMAND PI19, les données requises à la source de données. Ce signal devient inactif lorsque le tampon du décodeur est plein. Une fois que le traitement des données stockées est suffisamment avancé pour permettre la prise en compte de nouvelles données, la ligne DEMAND passe au niveau logique haut, signalant ce changement d'état au contrôleur par le biais de la ligne d'interruption IRQ6.

Le transfert des données MP3 (SID) en provenance de l'interface sérielle 0 du contrôleur en direction du décodeur se fait de façon asynchrone, le processeur générant la fré-

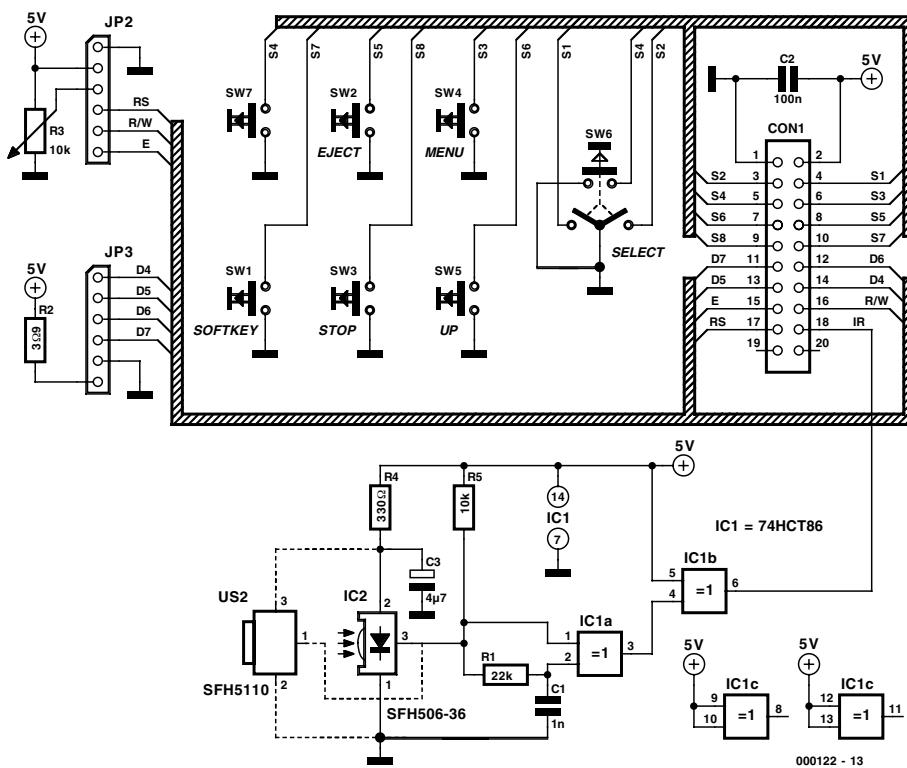


Figure 3. La commande d'OSCAR ne requiert que 4 touches et un encodeur rotatif. L'affichage LCD donne toutes les informations que peut souhaiter un utilisateur.

quence d'horloge (SIC) de 1 Mbit/s requise. Comme le taux de transmission est en tout état de cause supérieur au taux de bits du signal MP3, la transmission des données peut se faire par blocs, ce qui soulage le processeur et évite tout risque de décrochage au niveau du tampon en raison de l'absence d'une quantité de données suffisante (*buffer underrun*). Chaque trame MPEG traitée se traduit par la génération, au niveau du MAS3507D, d'une IRQ7, que l'on pourra mettre à profit pour déterminer les durées des morceaux. La transmission se fait par le biais de l'interface série sous la houlette d'une DMA (accès direct à la mémoire = *Direct Memory Access*), sans que le déroulement du programme n'ait à quelque moment que ce soit une influence sur cette transmission.

Le décodeur et le convertisseur numérique/analogique (CNA) monté en aval communiquent avec le processeur par le biais d'un bus I<sup>2</sup>C. Bien qu'il s'agisse d'un bus bidirectionnel, le microcontrôleur ne dispose pas de sorties en collecteur ouvert, ce qui a impliqué la prise en série dans la ligne SCL (*Serial CLock* = horloge série) d'une diode Schottky, D1, qui simule quasiment un collecteur ouvert.

2 lignes de port du microcontrôleur ont été

délégues au signal de données I<sup>2</sup>C bidirectionnel. On trouve, dans le sens allant du contrôleur à l'esclave I<sup>2</sup>C, un inverseur à sortie en collecteur ouvert, IC5f, dans le sens inverse, le signal de données entre par la ligne de port PB8. Tant qu'IC5f est inactif, les autres composants peuvent forcer le bus à la masse sans que cela ne pose le moindre problème.

L'utilisation d'un inverseur à collecteur ouvert résout un second problème. Le décodeur MP3, de même que le CNA requiert, contrairement à ce qui est le cas du contrôleur compatible TTL, une tension d'alimentation de 3,3 V; ces 2 composants ne supportent pas non plus l'application de tensions de niveau plus élevé à leurs entrées. La prise d'un inverseur à collecteur ouvert du type 74ACT05 au point nodal entre le contrôleur et les circuits intégrés à alimenter en 3,3 V, résout du même coup ce problème. Dans le sens inverse il n'y a pas d'inconvénient à connecter directement les lignes 3,3 V au microcontrôleur. La généra-

tion de l'alimentation de 3,3 V est l'affaire d'un régulateur de tension intégré à faibles pertes (*low drop*) du type LM3940, IC9, le convertisseur de tension continue intégré dans le MAS3507D n'étant lui pas mis à contribution.

## Interface analogique

Le CNA DAC3550 de la figure 2D a été conçu spécialement pour travailler en association avec le MAS3507; il lui fournit d'ailleurs la même fréquence d'horloge que celle qu'il utilise. On utilise à cet effet un quartz courant, X2, dont la fréquence d'oscillation est de 14,318 MHz. La valeur exacte de fréquence du quartz n'a pas une grande importance sachant que le DAC3550 se base sur la fréquence d'horloge fournie par le décodeur MPEG.

Le DAC3550 est un convertisseur Sigma-Delta haut de gamme doté d'un filtre antirepliement (*anti-aliasing*) intégré; il comporte un sélecteur d'entrée auquel est également connectée la sortie analogique du lecteur de CD-ROM, un amplificateur de sortie destiné aux signaux de niveau Ligne ainsi qu'un petit amplificateur de puissance avec réglage de volume destiné à se voir connecter un casque d'écoute. La liaison vers le décodeur se fait par le biais de 3 lignes I<sup>2</sup>S, à savoir SOI, SOD et SOC. La programmation des registres internes du convertisseur se fait elle par le biais du bus I<sup>2</sup>C.

Le convertisseur requiert une tension d'alimentation de 3,3 V (qui arrive par le biais du filtre passe-bas réduction de bruit que constitue la triplette L1, C19 et C23) pour sa section numérique et une alimentation de 5 V pour sa partie analogique. Ceci explique la présence d'un régulateur de tension linéaire réservé à ce composant. Cette mesure associée à une séparation physique des circuits de masse pour les électroniques numériques et analogiques et à un dessin de platine élaboré permet d'utiliser au mieux les capacités du CNA. La qualité du son ne peut qu'en être meilleure.

Le filtre antirepliement prend la forme physique d'une paire de filtres passe-bas à 2 pôles (R22 à R29 associées à C47 à C50). On dispose, outre de l'entrée CD-ROM analogique,

AUX1R et AUXL1 présente sur l'embase JP9, de l'entrée AUX2L/R, à laquelle on pourra, par exemple, connecter un récepteur radio. Cette embase met également à disposition les 2 bus-système (I<sup>2</sup>C et I<sup>2</sup>S). Il est possible, par leur biais, de réaliser, par exemple, une sortie S/PDIF numérique.

Un détail encore : lors de sa mise hors-fonction, le DAC3550A produit une quantité de bruit non négligeable. Nous avons prévu, en vue de l'éliminer, un générateur d'impulsion de remise à zéro du type MAX692, IC11, ce circuit intégré générant une impulsion NMI (*Non Masquable Interrupt* = Interruption non masquable) destiné au contrôleur dès que la tension d'alimentation tombe en-dessous de 4,7 V. Lorsque cela est le cas, le contrôleur force le CNA au silence par le biais du bus I<sup>2</sup>C.

## Interface IDE/ATAPI

Le pilotage du lecteur de CD-ROM et du disque dur se fait par le biais d'une interface IDE/ATAPI standard. En gros, l'interface IDE est constituée d'un bus de données d'une largeur de 16 bits et d'un bus d'adresses de 5 bits de large (A0 à A2, CS1FX et CS3FX) ainsi que de quelques signaux de commande additionnels. Un périphérique IDE met à disposition une série de 9 registres sélectables par le biais des lignes d'adresses. Tous ces registres, exception faite du registre de données à 16 bits, ont une largeur de 8 bits. Le **tableau 2** en donne une description globale.

Au niveau du PC, ces registres sont considérés par le contrôleur IDE comme étant autant de cellules de mémoire cartographiées dans le domaine des Entrées/Sorties du processeur; le hôte IDE peut partant les lire et y écrire. En ce qui concerne OSCAR, ces registres IDE se retrouvent dans l'un de ses 8 domaines

d'adresses du processeur.

L'hôte IDE pilote le périphérique IDE en respectant le protocole ATA. Avec cette norme, on commence par écrire les paramètres des instructions dans différents registres du lecteur avant d'écrire enfin le mot d'instruction proprement dit dans le registre COMMAND. Le lecteur exécute ensuite l'instruction. La transmission des données se fait ensuite par une série de lectures ou d'écritures au niveau du registre DATA. Le début de la transmission des données est soit indiqué par le biais d'une interruption générée par le lecteur soit détectée par l'hôte par une interrogation répétitive du registre STATUS. La communication avec le lecteur de CD-ROM en respect du protocole ATAPI repose sur le protocole ATA que nous venons de décrire. À l'inverse de ce qui est le cas en ATA, on commence par envoyer, par le biais du registre DATA, un groupe d'instruction complet (sous la forme d'un bloc de 12 octets) vers le lecteur. Le lecteur exécute ensuite cette instruction, la transmission de données proprement dite au travers du registre DATA débutant après.

La **figure 2E** reproduit l'électronique de l'interface IDE/ATAPI qu'utilise OSCAR. Les 9 registres du l'interface IDE sont accessibles dans le domaine d'adresse 6 du SH7032. Leur adressage se fait par le biais des lignes d'adresses A0 à A2 ainsi que A20 et A21. L'interface IDE pouvant utiliser des câbles en nappe d'une certaine longueur, toutes les lignes allant au périphérique IDE passent par un circuit de commande (*driver*) de bus. Les lignes de données passent elles par un driver de bus bidirectionnel du type 74ACT245, les lignes d'adresses et de commande (RD, WR) utilisent elles aussi un 74ACT245, câblé en sortie uniquement. Ce même driver pilote la ligne de remise à zéro (RESET). Le signal d'interruption

## Software et infos :

Pilotes ATA/ATAPI : [www.ata-atapi.com](http://www.ata-atapi.com)

Informations générales sur MP3 :  
[www.come.to/mp3chips](http://www.come.to/mp3chips)

actif au niveau haut de l'interface IDE passe par un inverseur sachant que le processeur ne peut traiter une interruption que si elle présente un flanc descendant (négatif).

Le choix d'un disque dur et/ou d'un lecteur de CD-ROM utilisable pour cette réalisation ne devrait pas être trop critique. On pourra utiliser n'importe quel disque dur reconnaissant l'adressage LBA, c'est-à-dire en fait la quasi-totalité des modèles de disque dur récents d'une capacité minimale de l'ordre de 1 Goctet. Bien que le système puisse travailler avec un formatage FAT16, il est recommandé d'utiliser pour le disque dur un format FAT32 pour le système de fichiers.

Pour autant que nous le sachions, on pourra utiliser tout lecteur de CD-ROM au standard ATAPI. On ne risque de problème qu'avec des lecteurs de première et seconde génération (2x à 8x) et encore dans certaines conditions avec des CD-ROM que l'on aura gravés soi-même.

## La périphérie

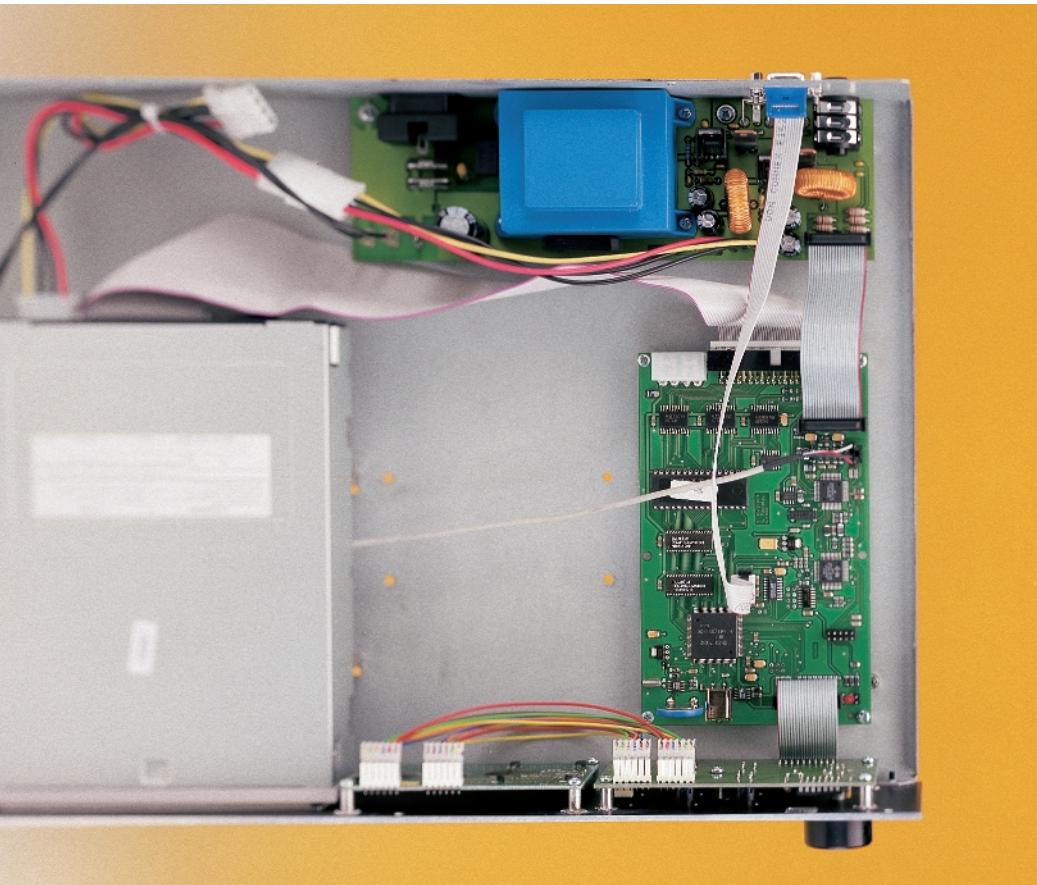
Le synoptique comportait une série de groupes de fonctions périphériques dont on retrouve les « interfaces » sur la platine principale représenté en **figure 2F** :

### L'affichage LCD

Nous utilisons ici un affichage LCD à matrice de points rétroéclairé par LED de l'industrie, le pilotage dudit affichage se faisant en mode 4 bits. Il va sans dire que l'affichage ne se trouve pas sur la platine principale mais sur la platine frontale d'où il est connecté à l'embase JP3.

### Le clavier

Les 4 touches EJECT, MENU, STOP et UP de même que l'encodeur rotatif SELECT attaquent le port C analogique d'une largeur de



8 bits du processeur, leur interrogation se faisant à intervalle régulier par l'intermédiaire d'une interruption générée par temporisateur (*timer*). L'encodeur travaille comme une paire d'interrupteurs ouverts et fermés l'un après l'autre. L'évolution chronologique de ces changements d'état permet au contrôleur de déterminer le sens de rotation. Le logiciel se charge également d'éliminer tout rebond, ce qui permet de limiter le matériel au strict minimum.

#### L'interface RS-232

Le contrôleur dispose de 2 interfaces sérielles, l'une d'entre elles étant compatible RS-232 asynchrone. Le contrôleur génère un taux de transmission maximum de 230,400 kbits/s, un MAX202, une variante de l'archi-connu MAX232, convertit les signaux en niveaux RS-232.

L'interface serielle servira au téléchargement, ultérieurement, de nouvelles versions du logiciel ainsi qu'à l'émission de messages d'états en cours de fonctionnement, étant ainsi un auxiliaire précieux lors du développement du programme. Si le système est doté d'un modem on pourra, par le biais de cette interface, télécharger de la musique depuis Internet (*Music On Demand*). Cette interface permet également une télécommande de l'ensemble de l'appareil. Un débogueur ins-

tallé sur PC pourra communiquer avec le processeur par le biais de l'interface RS-232 (pas d'acquitemment matériel).

#### Le récepteur IR

On trouve, sur la platine de la face avant, un récepteur IR (SFH506-36 ou SFH5110). Comme tant le code RC5 de Philips que le code SIRC de Sony reposent sur la MLI (**M**odulation en **L**argeur d'**I**mpulsion, **P**WM = *Pulse Width Modulation* en anglais) il faudra traiter tant les flancs montants que les flancs descendants. Ceci explique que le récepteur place, à chaque flanc, une courte impulsion sur la ligne de port PA13. La source de cette impulsion sont un réseau RC (R1/C1) et une paire de portes OU EXclusif (EXOR), IC1a et IC1b de la **figure 3**. Le contrôleur génère des interruptions IRQ1 lors de chaque flanc descendant de cette impulsion, ce qui lui permet le traitement des informations entrantes qu'elle véhicule.

#### La tension d'alimentation

L'appareil tout monté comporte une alimentation à découpage, mais rien n'interdit non plus, en principe, d'en-

visager l'utilisation d'une alimentation linéaire à condition qu'elle soit en mesure de fournir des tensions de 5 et 12 V propres à, pour chacune de ces 2 tensions, un courant de 2 000 mA. La consommation de courant réelle dépend du type de lecteur de CD-ROM (et/ou disque dur) utilisé. Il est recommandé, pour une utilisation mobile (par connexion à une batterie de voiture par exemple), d'utiliser un convertisseur CC/CC pouvant travailler en mode tant rehausseur qu'abaisseur de tension. Le fabricant de semi-conducteurs Linear Technology est dans ce cas-là l'un des partenaires privilégiés.

Horloge en temps réel (RTC) et code d'identification sur silicium

La platine comporte une horloge en temps réel et un numéro de série individualisé sous la forme d'un circuit à code d'identification (ID) sur silicium de Dallas Semiconductor; il ne remplit pas, pour le moment, de fonction réelle et est réservé à des applications futures.

### Des perspectives intéressantes

La réalisation chez soi d'un projet tel que celui que nous venons de décrire est réservé à une poignée de lecteurs disposant d'un appareillage quasi-industriel. En effet, les pistes de la carte au format Europe utilisée sont très serrées et souder manuellement des circuits intégrés CMS à 100 broches et plus n'est pas une sinécure. Nous avons voulu vous laisser une partie de réalisation personnelle sachant qu'OSCAR existe non seulement sous la forme d'un appareil tout monté, mais également sous celle d'un kit complet dont les plaques sont déjà dotées des composants requis. Vous pourrez également, s'il est dans vos intentions de réaliser votre propre application, un lecteur MP3 embarqué en voiture par exemple, acquérir séparément la plaque principale et le clavier ainsi que l'affichage LCD.

Nous verrons, dans le second article consacré à ce projet, comment faire, à partir du kit, un appareil complet. Le second aspect de ce second et dernier article sera l'examen de la structure du logiciel et son mode d'emploi.

**PIC17C75x**

**Set d'instructions**  
**Microcontrôleurs, logiciels**

Mnémonique, Opérandes	Description	Cycles	Code op 16-bits		État affecté	Notes
			MSb	LSb		
<b>COMF</b> f,d	Complement f	I	0001	001d ffff ffff	Z	
<b>CPFSEQ</b> f	Compare f with WREG, skip if f = WREG	I (2)	0011	0001 ffff ffff	None	6,8
<b>CPFSGT</b> f	Compare f with WREG, skip if f > WREG	I (2)	0011	0010 ffff ffff	None	2,6,8
<b>CPFSLT</b> f	Compare f with WREG, skip if f < WREG	I (2)	0011	0000 ffff ffff	None	2,6,8
<b>DAW</b> f,s	Decimal Adjust WREG Register	I	0010	111s ffff ffff	C	3
<b>DECf</b> f,d	Decrement f	I	0000	011d ffff ffff	O,V,C,DC,Z	
<b>DECFSZ</b> f,d	Decrement f, skip if 0	I (2)	0001	011d ffff ffff	None	6,8
<b>DCFSNZ</b> f,d	Decrement f, skip if not 0	I (2)	0010	011d ffff ffff	None	6,8
<b>INCF</b> f,d	Increment f	I	0001	010d ffff ffff	O,V,C,DC,Z	
<b>INCFSZ</b> f,d	Increment f, skip if 0	I (2)	0001	111d ffff ffff	None	6,8
<b>INFSNZ</b> f,d	Increment f, skip if not 0	I (2)	0010	010d ffff ffff	None	6,8
<b>IORWF</b> f,d	Inclusive OR WREG with f	I	0000	100d ffff ffff	Z	
<b>MOVFP</b> f,p	Move f to p	I	011p	pppp ffff ffff	None	
<b>MOVPF</b> p,f	Move p to f	I	010p	pppp ffff ffff	Z	
<b>MOVWF</b> f	Move WREG to f	I	0000	0001 ffff ffff	None	
<b>MULWF</b> f	Multiply WREG with f	I	0011	0100 ffff ffff	None	
<b>NEGW</b> f,s	Negate WREG	I	0010	110s ffff ffff	O,V,C,DC,Z	I,3
<b>NOP</b>	— No Operation	I	0000	0000 0000 0000	None	
<b>RLCF</b> f,d	Rotate left f through Carry	I	0001	101d ffff ffff	C	
<b>RLNCF</b> f,d	Rotate left f (no carry)	I	0010	001d ffff ffff	None	
<b>RRCF</b> f,d	Rotate right f through Carry	I	0001	100d ffff ffff	C	
<b>RRNCF</b> f,d	Rotate right f (no carry)	I	0010	000d ffff ffff	None	
<b>SETF</b> f,s	Set f	I	0010	101s ffff ffff	None	3
<b>SUBWF</b> f,d	Subtract WREG from f	I	0000	010d ffff ffff	O,V,C,DC,Z	I
<b>SUBWFB</b> f,d	Subtract WREG from f with Borrow	I	0000	001d ffff ffff	O,V,C,DC,Z	I
<b>SWAPF</b> f,d	Swap f	I	0001	110d ffff ffff	None	
<b>TABL RD</b> t,i,f	Table Read	2 (3)	1010	10ti ffff ffff	None	7
<b>TABL WT</b> t,i,f	Table Write	2	1010	11ti ffff ffff	None	5
<b>TLRD</b> t,f	Table Latch Read	I	1010	00tx ffff ffff	None	
<b>TLWT</b> t,f	Table Latch Write	I	1010	01tx ffff ffff	None	
<b>TSTFSZ</b> f	Test f, skip if 0	I (2)	0011	0011 ffff ffff	None	6,8
<b>XORWF</b> f,d	Exclusive OR WREG with f	I	0000	110d ffff ffff	Z	

**OPÉRATIONS ORIENTÉES BIT SUR LE REGISTRE DE FICHIER**

<b>BCF</b> f,b	Bit Clear f	I	1000	1bbb ffff ffff	None	
<b>BSF</b> f,b	Bit Set f	I	1000	0bbb ffff ffff	None	
<b>BTFSC</b> f,b	Bit test, skip if clear	I (2)	1001	1bbb ffff ffff	None	6,8
<b>BTFS</b> f,b	Bit test, skip if set	I (2)	1001	0bbb ffff ffff	None	6,8
<b>BTG</b> f,b	Bit Toggle f	I	0011	1bbb ffff ffff	None	

**OPÉRATIONS LITTÉRALES ET DE COMMANDE**

<b>ADDLW</b> k	ADD literal to WREG	I	1011	0001 kkkk kkkk	O,V,C,DC,Z	
----------------	---------------------	---	------	----------------	------------	--

**PIC17C75x**

**Set d'instructions**  
**Microcontrôleurs, logiciels**

**Source :**

Fiche de caractéristiques PIC17C75X, Microchip Technology Inc.

**Exemple d'application**

Mono-carte à PIC17, Elektor n° 267, septembre 2000

**Introduction**

Le set d'instructions du PIC17CXXX comporte 58 instructions. Chaque instruction prend la forme d'un mot à 16 bits subdivisé en un code opératoire (OPCODE) et un ou plusieurs opérande(s). Le code opératoire indique le type d'instruction, le (ou les) opérande(s) spécifiant le mode opératoire de l'instruction. Le set d'instructions du PIC17CXXX est subdivisé en 3 groupes :

- orienté octet (byte-oriented)
- orienté bit (bit-oriented)
- opérations littérales et de commande.

Les descriptions de champ pour les codes opératoires sont données dans le tableau I. Ces descriptions sont utiles pour la compréhension des codes opératoires donnés dans le tableau du set d'instructions et de la description spécifique de chacune des instructions.

**Instructions orientées octet,**

'f' représente un désignateur de registre de fichier et 'd' désigne un désignateur de destination. Le désignateur de registre de fichier spécifie que registre de fichier sera utilisé par l'instruction.

Le désignateur de destination indique où devra être mis le résultat de l'opération. Si 'd' = '0', le résultat sera placé dans le registre WREG. Si 'd' = '1', le résultat sera placé dans le registre de fichier spécifié par l'instruction.

**Instructions orientées bit,**

'b' représente un désignateur de champ de bit qui détermine le nombre du bit concerné par l'opération, 'f' représentant le nombre du fichier dans lequel se trouve le bit.

**Opérations littérales et de commande**

'k' représente une constante ou une valeur littérale de 8 ou 13 bits.

Le set d'instructions est fortement orthogonal et subdivisé en groupes d'opérations :

- orientées octet
- orientées bit
- littérales et de commande.

L'exécution de toutes les instructions se fait au cours d'un unique cycle d'instruction à moins que :

- Un test conditionnel ne s'avère « vrai »
- Le compteur ordinal (de programme) n'aït changé à la suite de l'exécution d'une instruction
- Il n'y ait eu exécution d'une instruction de lecture ou d'écriture de tableau (table); dans ce cas-là l'exécution requiert 2 cycles d'instruction, le second cycle étant exécuté comme une NOP = No OPeration).

Un cycle d'instruction se compose de 4 périodes d'oscillateur. Ainsi, si l'oscillateur tourne à une fréquence de 25 MHz, la durée normale d'exécution d'une instruction est de 160 ns. Cette durée d'exécution d'instruction passera partant à 320 ns dans le cas d'un test conditionnel vrai ou d'un changement du compteur ordinal dû à une instruction.

## Description des champs de code opérateur

Champ	Description
F	Register file address (00h to FFh)
P	Peripheral register file address (00h to 1Fh)
I	Table pointer control i = '0' (do not change) i = '1' (increment after instruction execution)
t	Table byte select t = '0' (perform operation on lower byte) t = '1' (perform operation on upper byte literal field, constant data)
WREG	Working register (accumulator)
B	Bit address within an 8-bit file register
K	Literal field, constant data or label
X	Don't care location (= '0' or '1') The assembler will generate code with x = '0'. It is the recommended form of use for compatibility with all Microchip software tools.
D	Destination select 0 = store result in WREG I = store result in file register f Default is d = '1'
U	Unused, encoded as '0'
S	Destination select

## Set d'instructions du PIC17Cxx

Mnémonique, Opérandes	Description	Cycles	Code op 16-bits	État affecté	Notes
<b>OPÉRATIONS ORIENTÉES OCTET SUR LE REGISTRE DE FICHIER</b>					
ADDWF f,d	ADD WREG to f	I	0000 111d ffff ffff	OV,C,DC,Z	
ADDWFC f,d	ADD WREG and Carry bit to f	I	0001 000d ffff ffff	OV,C,DC,Z	
ANDWF f,d	AND WREG with f	I	0000 101d ffff ffff	Z	
CLRF f,s	Clear f, or Clear f and Clear WREG	I	0010 100s ffff ffff	None	3

Mnémonique, Opérandes	Description	Cycles	Code op 16-bits				État affecté	Notes
			MSb	LSb				
ANDLW k	AND literal with WREG	I	1011 0101	kkkk	kkkk	Z		
CALL k	Subroutine Call	2	111k	kkkk	kkkk	kkkk	None	7
CLRWD T	Clear Watchdog Timer	I	0000 0000	0000	0100		TO,PD	
GOTO k	Unconditional Branch	2	110k	kkkk	kkkk	kkkk	None	7
IORLW k	Inclusive OR literal with WREG	I	1011 0011	kkkk	kkkk	Z		
LCALL k	Long Call	2	1011 0111	kkkk	kkkk	kkkk	None	4,7
MOVLB k	Move literal to low nibble in BSR	I	1011 1000	uuuu	kkkk	kkkk	None	
MOVLR k	Move literal to high nibble in BSR	I	1011 101x	kkkk	uuuu	uuuu	None	
MOVlw k	Move literal to WREG	I	1011 0000	kkkk	kkkk	kkkk	None	
MULLW k	Multiply literal with WREG	I	1011 1100	kkkk	kkkk	kkkk	None	
RETFIE —	Return from interrupt (and enable interrupts)	2	0000 0000	0000	0101		GLINTD	7
RETLW k	Return literal to WREG	2	1011 0110	kkkk	kkkk	kkkk	None	7
RETURN —	Return from subroutine	2	0000 0000	0000	0010		None	7
SLEEP —	Enter SLEEP Mode	I	0000 0000	0000	0011		TO,PD	
SUBLW k	Subtract WREG from literal	I	1011 0010	kkkk	kkkk	kkkk	O,V,C,DC,Z	
XORLW k	Exclusive OR literal with WREG	I	1011 0100	kkkk	kkkk	kkkk	Z	

## Notes

- 1 : Méthode par complément à 2.
- 2 : Arithmétique non signée.
- 3 : Si s = '1', seul le fichier est concerné; si s = '0' tant le registre WREG que le fichier sont concernés. Si l'on ne veut affecter que le registre de travail (WREG = Working REGister) il faudra spécifier f = WREG.
- 4 : Au cours d'une instruction LCALL, les contenu de PCLATH sont chargés dans le MSB (Most Significant Byte = octet de poids fort) du compteur ordinal (PC = Program Counter), kkkk kkkk étant chargé dans le LSB (Least Significant Byte = octet de poids faible) du PC (PCL).
- 5 : Instruction à cycle multiple pour la programmation d'EPROM lorsque le pointeur de tableau désigne l'EPROM interne. L'instruction est terminée par un événement d'interruption. En cas d'écriture vers de la mémoire de programme externe il s'agit d'une instruction à 2 cycles.
- 6 : Instruction à 2 cycles lorsque la condition est vraie, sinon instruction à 1 cycle.
- 7 : Instruction à 2 cycles exception faite pour TABLRD to PCL (octet de poids faible du compteur ordinal) auquel cas elle dure 3 cycles.
- 8 : Un 'skip' (outrepasser) signifie que l'instruction saisie au cours de l'exécution de l'instruction en cours n'est pas exécutée, elle sera remplacée par une exécution d'une instruction NOP.

# Les systèmes Surround modernes

## Dolby Digital contre DTS

Harry Baggen

Avec l'avènement du DVD, il est enfin possible de graver plus de 2 canaux sonores de bonne qualité sur un support maniable. Maintenant, le son Surround a de grandes chances de débarquer vraiment dans notre salon. Examinons ici le fonctionnement et les caractéristiques de 2 des systèmes Surround les plus importants du moment, le *Dolby Digital* et le *DTS*.



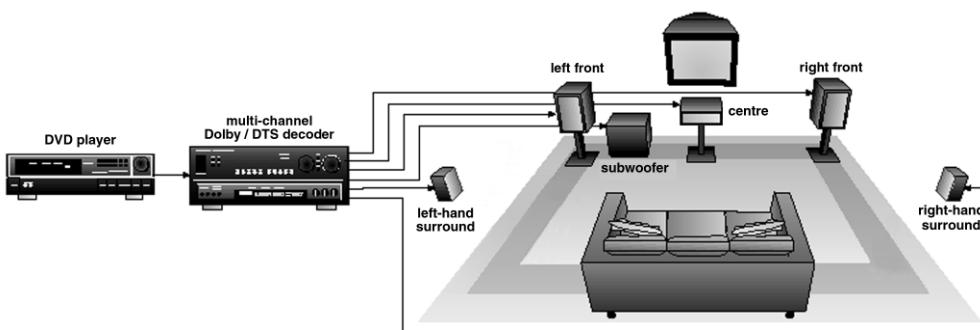


Figure 1. Une installation Surround domestique se compose de 5 enceintes et d'un caisson de grave séparé.

Le son Surround n'est pas une toquade de la mode de ces dernières années. Au cinéma, on travaillait déjà dans les années 70 avec plusieurs pistes de reproduction du son pour créer davantage de réalisme grâce à l'ambiance acoustique. Mais à la maison, le son à canaux multiples n'a jamais, jusqu'il y a peu, connu de réelle vulgarisation. L'industrie musicale a bien tenté, au cours des années 70 également, de lancer la quadriphonie, mais le projet n'a pas rencontré de succès, il y fallait des aiguilles spéciales pour la lecture des microsillons, d'ailleurs bien rares, et le public n'y a consacré qu'un intérêt très limité. Une reproduction stéréophonique convenable coûtait déjà suffisamment cher et la plupart des audiophiles voyaient mal où installer chez eux 2 baffles supplémentaires, un décodeur et un second amplificateur stéréophonique.

Pourtant, la reproduction Surround apporte, par une dimension supplémentaire, une présence sonore bien tangible. Ce n'est pas seulement vrai pour les films dans lesquels des avions en rase-mottes survolent vos têtes, mais aussi pour les concerts au cours desquels vous pouvez réellement vous croire au milieu du public.

De nouvelles techniques d'enregistrement et des médias à grande capacité de mémoire (CD/DVD), mais aussi des prix à la baisse pour les chaînes stéréo et les enceintes acoustiques, il n'en faut pas davantage pour inverser la tendance et attiser l'envie du grand public pour le son Surround à domicile. La percée décisive, on la doit au DVD, c'est le support qui a été la source idéale

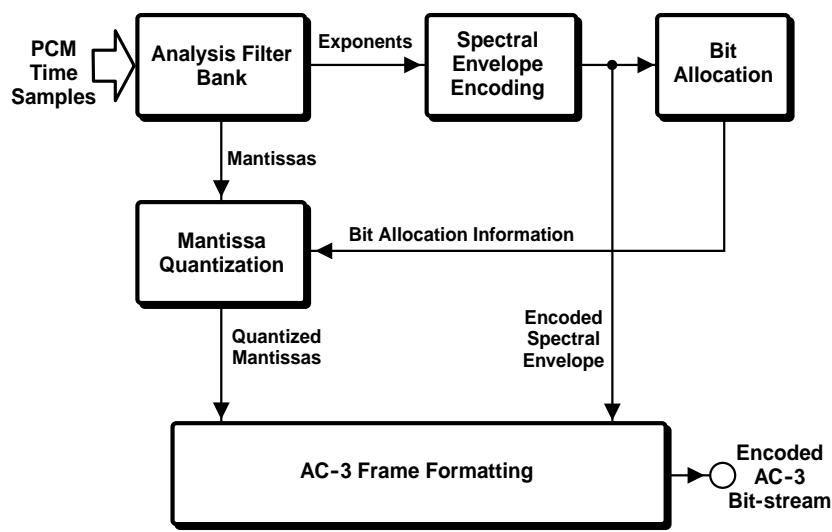
de reproduction à canaux multiples. Sur un DVD, à côté de la vidéo, vous pouvez trouver de nombreux canaux audio et même différents systèmes de codage.

Il n'y a, à l'heure actuelle, que 2 systèmes qui ont pu s'accrocher dans le secteur et tous deux émanent du monde du cinéma. Le plus célèbre, c'est Dolby, avec le *Dolby Digital*, suivi de son *sparring-partner*, moins bien soutenu, mais qualitativement supérieur, le DTS. Tous les autres systèmes, comme MPEG, ont été pratiquement éradiqués du marché, faute de parrainage.

## Matériel

Un système Surround moderne fait usage de 5 canaux plus un ultra-grave séparé, le 5.1-audio, comme à

la **figure 1**. Deux canaux alimentent les enceintes frontales, un autre, le canal médian, est chargé d'améliorer l'audibilité de la parole et la localisation dans l'espace et deux voies arrières assurent les effets et la création d'une acoustique spatiale. L'infra grave vient en supplément, non directif, il sert principalement, dans les films, à la reproduction de bruitages qui s'apparentent aux infrasons. De quoi devons-nous disposer chez nous pour reproduire le son Surround ? D'abord, un d'amplificateur Surround ou d'un récepteur équivalent, avec le décodeur pour le Dolby Digital, au minimum. Pareil récepteur, avec ses cinq amplificateurs de puissance, coûte aujourd'hui quelque 300 euros. Il faut y ajouter un lecteur de DVD, qui vaut à peu près autant et l'ensemble des enceintes acoustiques. Pour limiter les frais, on peut utiliser les baffles de la chaîne stéréo, il n'y a plus que trois haut-parleurs à acheter, pour le canal central et l'arrière, plus éventuellement un *subwoofer*.



000075-12

Figure 2. Le principe du codeur Dolby Digital (DD).

Provisoirement, on peut se passer du canal médian et ne mettre que de petits haut-parleurs à l'arrière, les systèmes modernes sont très souples à cet égard.

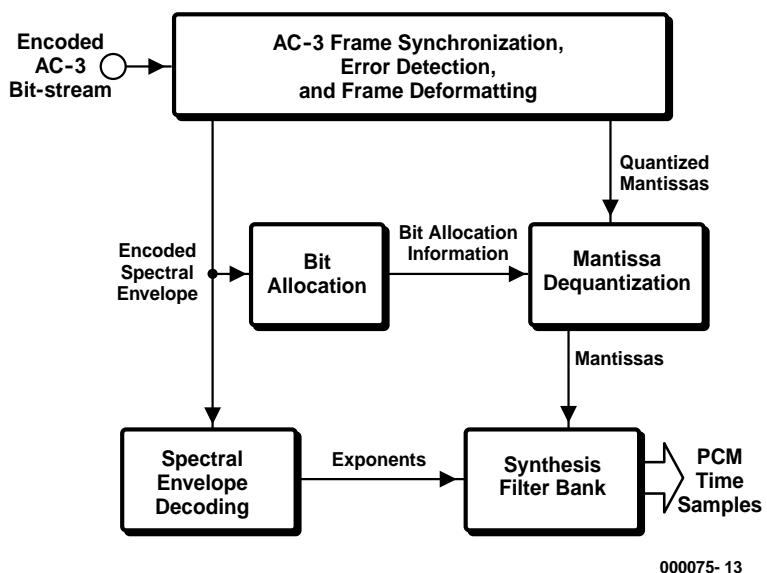
Au milieu de toute cette sonorisation, nous aurions presque oublié le poste de télévision, qui joue pourtant un rôle important et devrait occuper une position centrale dans le dispositif, entre les enceintes frontales, de préférence.

## Compression

Malgré l'énorme capacité du DVD, y caser cinq canaux sonores complets, voire davantage, en plus des images vidéo, représenterait un volume trop important. Raison pour laquelle tant le son que l'image subissent au préalable une compression. Voilà un terme encore passablement ambigu et qui donne à penser que la qualité d'ensemble aura à en souffrir. Mais grâce à des techniques de compression futées, ce n'est vraiment pas le cas. En outre, lors de la compression de canaux multiples synchrones, on trouve souvent une grande redondance dans les signaux, des similitudes qui rendent inutile l'inscription séparée de chaque canal. Sinon, un enregistrement complet de six canaux sous une résolution de 16 bits et une fréquence d'échantillonnage de 48 kHz nécessite un débit de données de 5,184 Mbit/s.

## Dolby Digital

Le Dolby numérique met en œuvre un algorithme qui a été depuis standardisé par l'ATSC (*United States Advanced Television Systems Committee*) sous l'appellation AC-3. Cet algorithme est capable de réduire le volume d'un enregistrement à 5+1 canaux (5 canaux complets et un infra grave) pour atteindre un débit entre 32 et 640 Kbit/s, selon la qualité demandée.



000075- 13

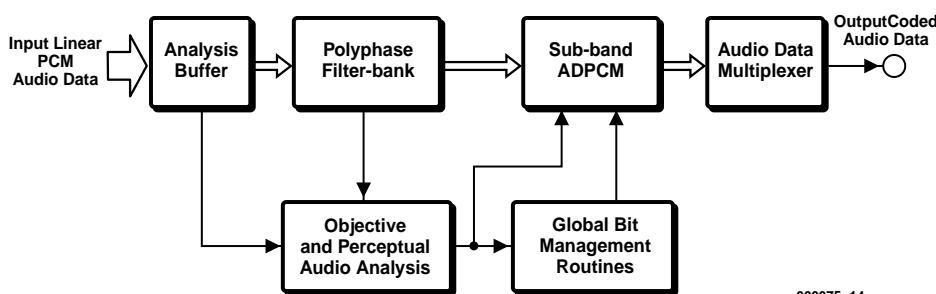
Figure 3. Le décodeur DD contient la logique nécessaire à la traduction inverse de tous les signaux.

Le diagramme fonctionnel d'un codeur AC-3 est représenté à la **figure 2**. Le signal d'entrée du codeur est constitué de données PCM. Elles sont d'abord converties dans le domaine de fréquences. C'est le rôle d'une batterie de filtres d'analyse. Les blocs juxtaposés avec chevauchement des 512 échantillons temporels sont multipliés avec une fenêtre de l'axe du temps et rangés dans le domaine de fréquence. Le chevauchement d'origine des blocs temporels fait en sorte que chaque échantillon d'entrée se retrouve dans deux bandes de fréquence successives. On simplifie, pour n'en garder qu'un sur deux, ce qui en ramène le nombre à 256 coefficients. Chacun

d'eux est enregistré en notation exponentielle, un exposant binaire et une mantisse. L'ensemble des exposants est décodé pour donner la forme générale du spectre du signal. C'est ce qu'on nomme l'enveloppe spectrale. Elle est ensuite utilisée par une routine d'attribution de bit (*bit allocation*) de manière à déterminer le nombre de bits nécessaires au codage de chacune des mantisses. L'enveloppe spectrale et les mantisses quantifiées approximativement pour six blocs (1 536 échantillons audio) sont alors consignées dans une trame AC-3. Le débit binaire AC-3 total est ainsi constitué de trames AC-3 consécutives.

Il y a plusieurs choses qui ne sont pas visibles dans le diagramme. On y ajoute par exemple un préambule qui regroupe toutes les informations générales nécessaires (débit binaire, fréquence d'échantillonnage, nombre de canaux, etc). S'y associent également les codes de détection d'erreur.

Mais il existe en outre différentes possibilités supplémentaires de codage. On peut ainsi modifier dynamiquement la résolution spectrale des batteries de filtres d'analyse, coder l'enveloppe spectrale sous une résolution temporelle variable et coupler des canaux pour relever la bande passante ou atteindre une plus forte compression.



000075- 14

Figure 4. Diagramme fonctionnel du codeur DTS, qui utilise le *Coherent Acoustic Algorithme*.

à débit binaire réduit.

La procédure de décodage est l'inverse de celle de codage. Le décodeur (à la **figure 3**) doit d'abord se synchroniser sur le débit binaire codé, vérifier s'il y a des erreurs et décortiquer les différentes sortes de données, comme l'enveloppe spectrale codée et les mantisses quantifiées. La routine d'attribution de bits fournit les renseignements nécessaires au déballage des mantisses et à leur déquantification. Il faut alors au décodeur extraire de l'enveloppe spectrale les exposants et finalement reconvertis les mantisses dans le domaine temporel.

Ce schéma-ci également est un peu lapidaire, il y manque entre autres la correction d'erreur, le découplage éventuel des canaux et l'adaptation dynamique des batteries de filtres.

## DTS Digital Surround

Bien que le Dolby Digital jouisse d'une plus large diffusion, la plupart des professionnels s'accordent pour dire que le DTS Digital Surround offre une meilleure qualité sonore. Quand on compare les diagrammes fonctionnels, on remarque effectivement des différences sensibles, mais il est bien délicat de définir précisément à quoi tient la variation à l'audition. Ce qui est indéniable, c'est que le DTS engendre un flux de données plus volumineux que le Dolby Digital.

Le DTS se sert de l'algorithme connu sous le nom de *Coherent Acoustic Compression*, appellation qu'il ne semble pas nécessaire de traduire. La **figure 4** en donne le principe sous forme de blocs fonctionnels. Tout d'abord, le signal entrant est dissocié en une série de sous bandes par l'intermédiaire d'une batterie de filtres polyphasée. Cela donne une information sur la répartition spectrale en temps réel et permet aussi d'éviter certaines redondances acoustiques. À ce stade, chaque partition, chaque sous bande recèle encore des données audio sous forme P.M. linéaire, mais à bande passante étroite. Cette largeur de spectre et le nombre de sous bandes dépendent de la largeur de spectre du signal entrant, mais en général, la gamme audio est subdivisée en 32 bandes équivalentes.

Chaque partition subit alors un

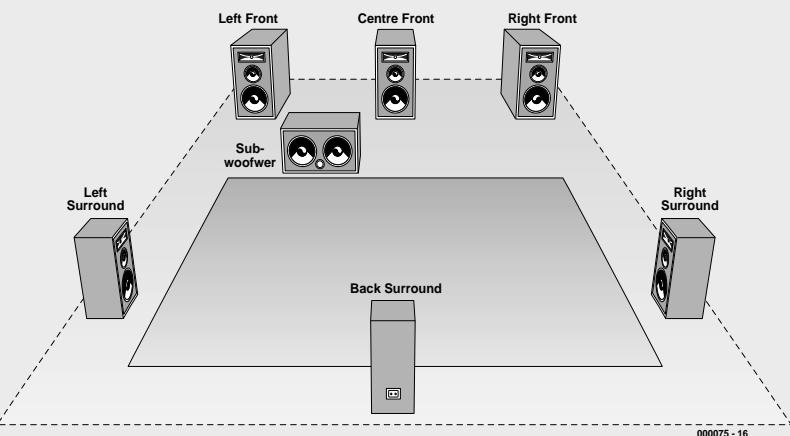
codage différentiel (*subband ADPCM*). Mais simultanément, on réalise des analyses psychoacoustiques et transitoires sur les signaux présents dans les sous bandes afin d'y détecter les informations que l'oreille ne peut percevoir et qui sont donc inutiles. Suivant le débit binaire, ces résultats sont employés pour adapter la routine principale chargée du codage différentiel. La

combinaison du codage différentiel et de l'analyse psychoacoustique se révèle particulièrement efficace.

Un élément intéressant à remarquer, c'est que lorsque le débit de données augmente, le taux d'adaptation psychoacoustique diminue et donc la qualité du signal s'améliore.

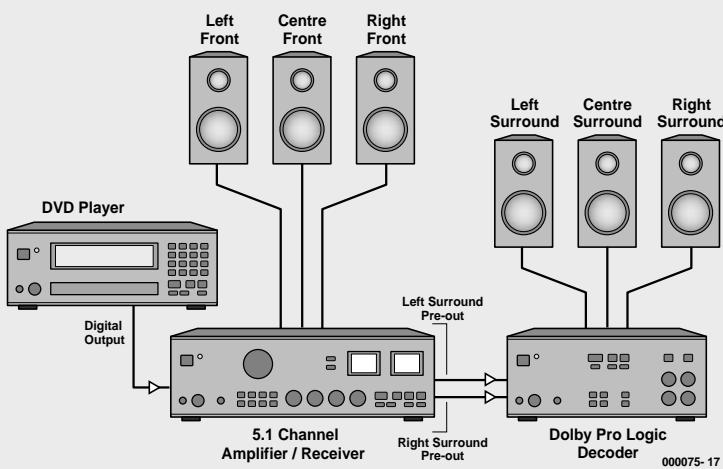
Les routines d'attribution des bits s'occupent de la répartition et du repérage des données dans toutes les sous bandes des canaux audio. Le dernier étage du codeur

## Dolby Digital Surround EX



Entre-temps, un nouveau standard est apparu sur les appareils domestiques récents les plus chers, le Dolby EX (que certains appellent Dolby 6.1). Un canal supplémentaire vient s'ajouter aux cinq principaux existants. Il vient compléter et préciser le panorama de la poupe, il se situe exactement entre les deux canaux arrière, c'est donc l'arrière central. Un certain nombre de salles de cinéma en sont déjà équipées. En pratique, inclure dans notre installation domestique ce sixième canal est aisément réalisable, sans devoir changer de décodeur. Il suffit d'un décodeur (analogique) Dolby Pro Logic et d'un amplificateur supplémentaire. Les sorties des deux canaux Surround sont envoyées au décodeur Pro Logic qui fournit alors deux nouveaux canaux arrière et l'arrière central supplémentaire. Différents périodiques et de nombreux amateurs fanatiques ont essayé le truc et déclarent qu'il fonctionne parfaitement.

Bien évidemment, la concurrence n'est pas restée inactive. DTS propose le DTS-ES (pour *Extended Surround*) qui dispose également d'un arrière central.



## THX et THX Ultra

THX de Lucasfilm (eh oui, celui-là même à qui l'on doit La guerre des étoiles !) n'est pas un système Surround à part (ouf !), mais une extension de systèmes existants combinée à une norme de qualité qui prescrit les exigences auxquelles une installation Surround et son logiciel doivent répondre pour mériter le label



« THX ». On distingue deux groupes, THX Select et THX Ultra. La principale différence entre les deux réside dans le volume de la salle. THX Select s'adresse à des pièces de moins de 70 m<sup>2</sup>, THX Ultra aux plus grandes salles, plus de 100 m<sup>2</sup>, de préférence. Le matériel doit satisfaire aux exigences formulées, et elles sont sévères. Citons par exemple la puissance et le taux de distorsion des amplificateurs, ainsi que le diagramme de rayonnement des haut-parleurs, leur caractéristique spectrale et la pression acoustique obtenue. En outre, THX oblige d'utiliser pour les enceintes arrière des dipôles. Le processeur THX accepte les signaux Dolby Pro Logic et Dolby Digital et les adapte à l'environnement acoustique. Une des particularités du système réside dans la décorrélation des canaux arrière, de manière à empêcher une localisation précise de la source sonore. Il y a en outre une fonction qui vise à l'équivalence de rendu des timbres entre l'avant et l'arrière, de sorte que lors d'un déplacement, le bruitage reste identique et passe harmonieusement de l'arrière à l'avant et vice versa.

Les fabricants doivent soumettre leur matériel au contrôle de THX pour obtenir l'agrément. Toutes les caractéristiques sont passées en revue, y compris la précision des signaux de test internes.

est constitué d'un multiplexeur de données qui associe en un même flux binaire les données des sous bandes de tous les canaux, des informations complémentaires et des bits de synchronisation.

Dans le décodeur, à la **figure 5**, après synchronisation, le flux est partagé en ses constitutantes et les erreurs détectées sont corrigées. On peut ensuite démultiplexer les données et les répartir entre canaux et sous bandes. C'est dans le deuxième étage que s'opère l'inverse de la quantification sur les signaux différentiels pour recréer les bandes PCM, suivant les instructions jointes dans chaque sous bande. Il n'y a plus alors qu'à replacer les signaux dans le dans le domaine

temporel, aux fréquences d'origine, pour reconstituer le spectre sonore complet, par un filtrage inverse de celui du codeur.

Un avantage considérable du système DTS, c'est que toute « l'intelligence » se situe dans le codeur, si bien que le décodeur ne joue presque plus qu'un rôle passif. Toutes les instructions de décodage sont incluses dans la transmission binaire, même l'algorithme de codage peut être modifié sans que le décodeur ne doive être adapté de l'extérieur.

## NOMBREUSES POSSIBILITÉS

Les deux systèmes autorisent de multiples latitudes pour adapter le signal aux convenances particulières.

D'abord, on peut indiquer le type et le nombre de haut-parleurs raccordés et le processeur du décodeur en tiendra compte dans l'aiguillage des signaux. On peut aussi lui fournir les mesures de distance entre les enceintes et l'auditeur et la position des canaux à l'arrière, question d'adapter les temps de retard. La répartition des fréquences est également susceptible d'adaptation aux circonstances. Par exemple, si les baffles arrière sont de petit format, les fréquences les plus graves seront redirigées vers les haut-parleurs frontaux. Il en va de même, naturellement pour le canal particulier de grave : en l'absence d'un récepteur spécifique, l'information est répartie sur les autres voies.

## ENCORE PLUS DE CANAUX ?

Les systèmes Surround modernes ne sont pas seulement des joyaux de la technique, ils offrent réellement une nouvelle dimension au plaisir des oreilles, même à la maison. Le consommateur leur a réservé un accueil relativement large, mais, une fois de plus, l'industrie a malheureusement tendance à l'exagération. Elle veut encore y ajouter d'autres canaux. Or la plupart des utilisateurs sont déjà bien contents d'avoir pu caser dans leur salon deux enceintes arrière. Où s'arrêtera-t-on ? Enfin, nous pouvons toujours nous consoler en pensant que ces systèmes sont généralement dotés d'un *down-matrix-mode*, un réseau de commutation qui permet de ramener le nombre de canaux à celui des voies disponibles.

(000075)

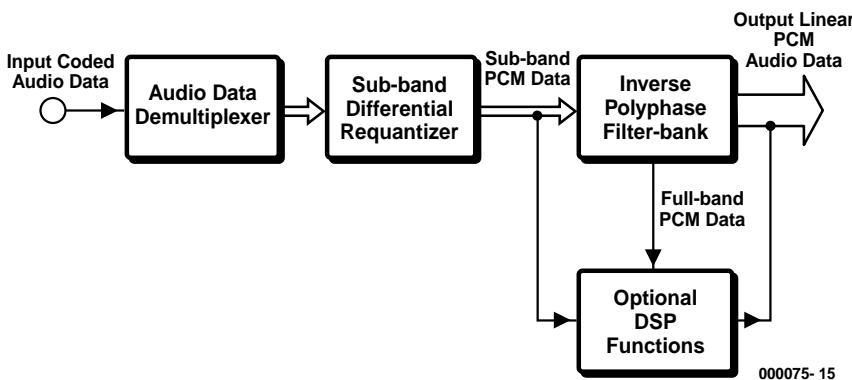


Figure 5. Le décodeur DTS est dit passif, parce que toutes les instructions de décodage sont incluses dans les données transmises.

*Vous trouverez davantage d'information sur ces systèmes aux adresses Internet suivantes :*

*Dolby:* <http://www.dolby.com>

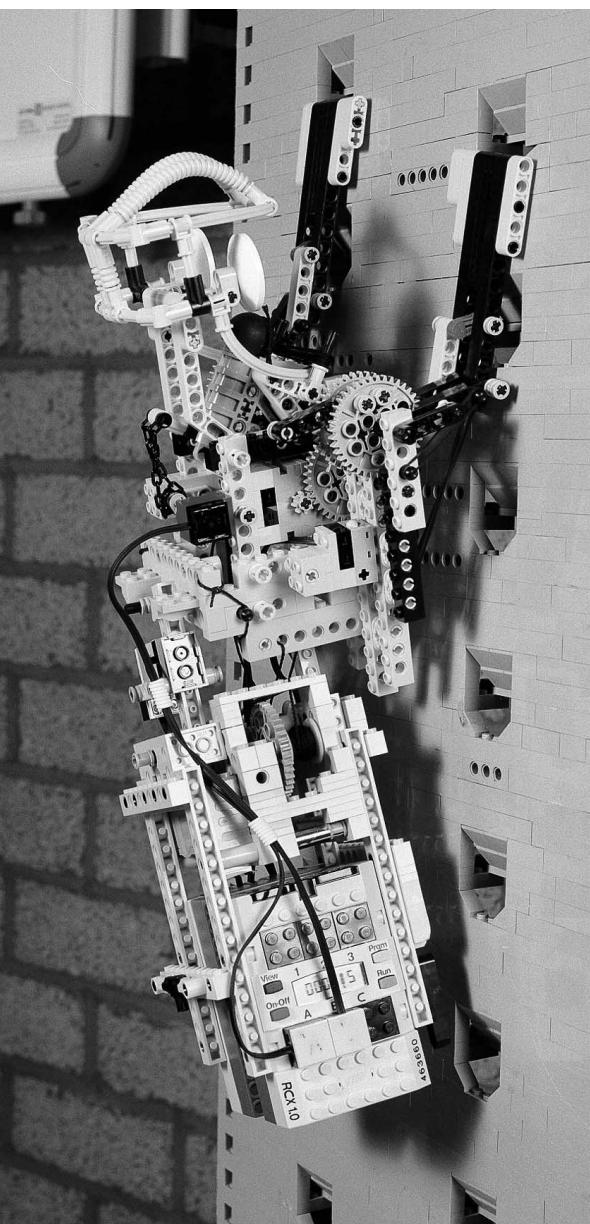
*DTS:* <http://www.dtstech.com>

*THX:* <http://www.thx.com>

# Lego Robotics Invention System

## 4<sup>e</sup> partie : des robots qui causent

Hans Steeman / Luc Lemmens



Le Robotics Invention System est par excellence destiné à la construction de robots évolués. Grâce au module RCX, on peut interroger différents capteurs et, à l'aide des trois sorties modulées en largeur d'impulsion, commander trois moteurs ou autres organes d'exécution. Parfois, on voudrait implanter davantage de potentialités dans le robot. En application de la devise « l'union fait la force », il n'y a plus qu'à mettre au travail une équipe de modules RCX.

Quelques expérimentations avec le Robotics Invention System vous convaincront de sa puissance et de la diversité de ses atouts. N'empêche, il possède aussi ses limites. Les automates complexes se targuent généralement de plus de fonctions que celles dont peut disposer un bloc RCX, que ce soit en entrées ou en sorties. Mais la souplesse du système est telle qu'il peut conjuguer ses forces pour s'adjointre de nouvelles possibilités. Le prototype de démonstration de Lego, illustré par la photo en tête de cet article, en

fournit un exemple parlant. Une seule maquette regroupe trois blocs RCX qui collaborent de manière à aboutir au résultat souhaité. Un RCX s'emploie au système de transport, deux autres modules s'affairent, à l'intérieur du robot, sur base d'un véritable partage des tâches, tout en communiquant à trois à propos de l'avancement du processus. On peut ainsi mettre en service un RCX propre à chaque fonction, qui, à tour de rôle, passe le témoin au suivant, une fois sa tâche accomplie. Mais on peut tout aussi bien faire travailler

deux blocs RCX indépendamment l'un de l'autre et se servir de messages pour superviser le déroulement du processus. Le constructeur de robots débutant réussira sans trop de peine à maîtriser les fonctions de communication du système dans l'environnement de développement standard.

## Communication

Pour communiquer entre eux, les robots utilisent l'émetteur-récepteur infrarouge présent dans chaque RIS, une interface capable d'échanger des codes à huit bits, donc des nombres compris entre 0 et 255. Chaque robot peut transmettre un message et tous les blocs RCX présents dans le domaine de couverture le reçoivent. Ce que le module qui le capte en fera réellement dépend exclusivement du logiciel dont il est porteur.

Le rayon d'action dans lequel les modules sont en mesure de communiquer est à régler à l'aide de l'écran de configuration. Si vous choisissez « Long », l'émetteur travaillera à pleine puissance et sa portée sera maximale. Cette latitude de configuration s'apparente à l'interrupteur présent sur l'émetteur infrarouge relié au PC. Il détermine également la puissance d'émission de cet émetteur et rien d'autre, tandis que le réglage de l'émetteur du RCX en service n'est possible qu'à l'intervention du logiciel. Le revers de la médaille, quand on travaille à puissance maximale, c'est le risque de brouiller d'autres RCX. En outre, on décharge plus vite les batteries, surtout si les robots se montrent bavards. Mieux vaut donc s'adapter aux circonstances, trouver pour l'ensemble un juste milieu.

Attaquons-nous à présent à la mise en chantier de cet échange de messages.

### Détecteur de message RCX

Le RCX-message *sensor watcher* constitue une option que l'on trouve dans les commandes standard. Ce bloc fonctionnel bleu opère à la manière d'un capteur ordinaire, mais il n'est nullement nécessaire de le connecter séparément.

Il guette en permanence l'arrivée d'un message et déclenche l'exécu-

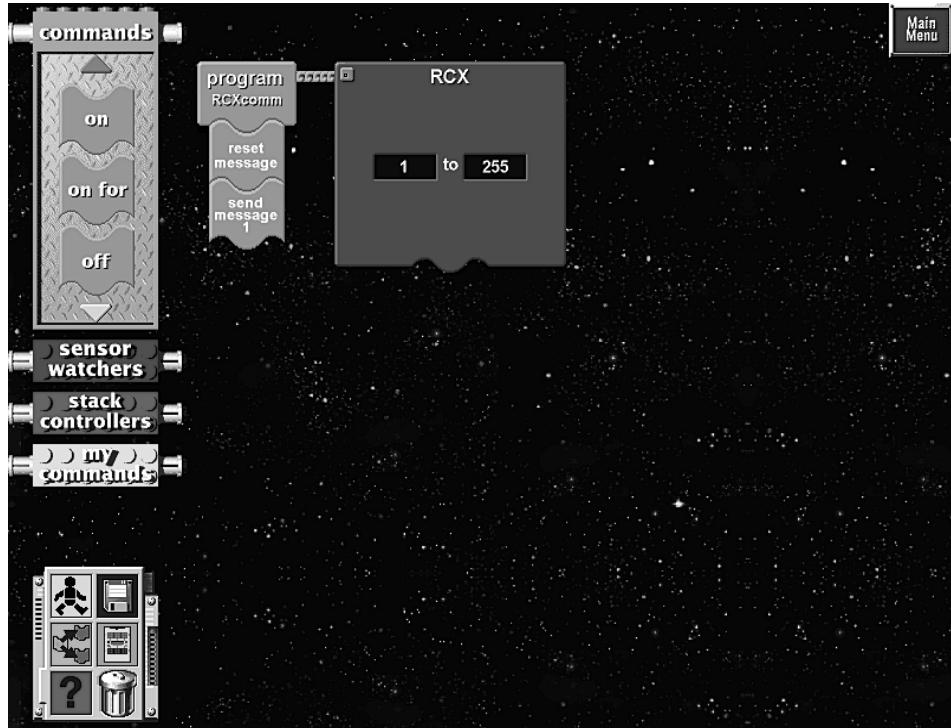


Figure 1. Les icônes des fonctions relatives aux messages proviennent du logiciel Lego.

tion d'une routine dès réception de données en provenance d'un autre RCX. Il est possible, à l'aide des instructions qui constituent ce bloc

fonctionnel, de déterminer à quel contenu de message le RCX doit réagir. Il suffit de cliquer sur les nombres présentés pour introduire les limites supérieure et inférieure des valeurs

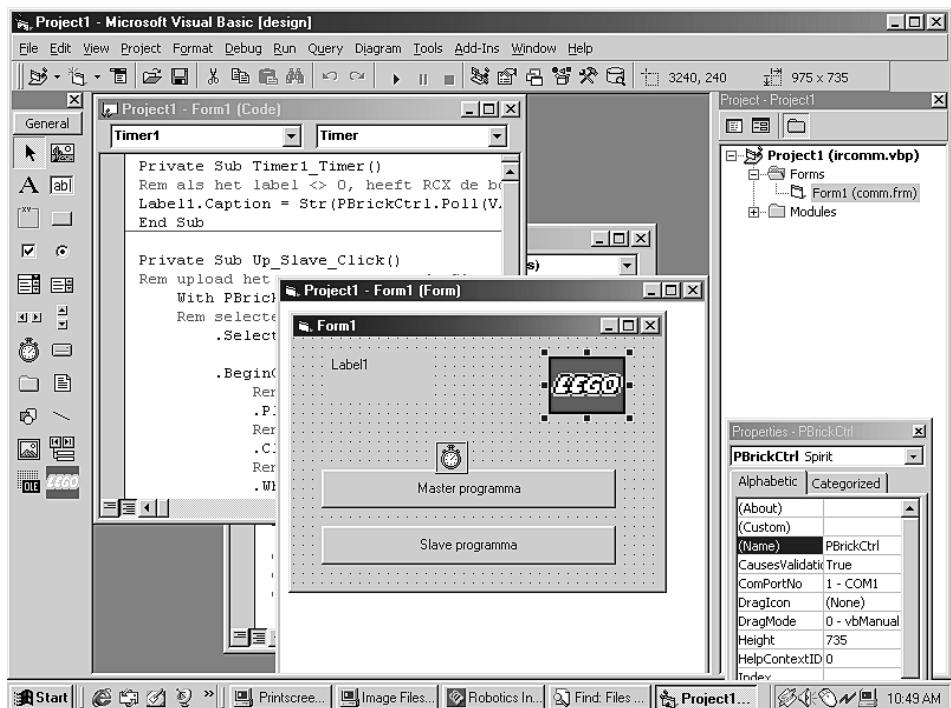


Figure 2. Ce formulaire peut se combiner au logiciel en Visual Basic. Les deux boutons conduisent au lancement des programmes dans le maître et l'esclave. Le résultat de l'exécution de Label1 apparaît au-dessus, à droite. On remarque que la relation avec le module Lego ActiveX est effective.

## Tableau I. La fonction Poll (interrogation).

### Poll(Source, Number)

Source et Number sont donnés pour indiquer où la fonction Poll doit aller voir dans le RCX.

#### Source Number Fonction

0	0 à 31	Variables 0 à 31
1	0 à 3	Timer 0 à 3
2	-	-
3	0,1,2	Motor status Bit 7: On/Off I/O Bit 6: Brake/Float I/O Bit 5: Output no. HiBit Bit 4: Output no. LoBit Bit 3: Direction CW/CCW I/O Bit 2: Power Level most significant bit Bit 1: Power Level Bit 0: Powel Level least significant bit
4	-	-
5	0,1	No function with RCX
6	0,1	No function with RCX
7	2	No function with RCX
8	-	RCX program number. Number actual program in use
9	0,1,2	SensorValue, value measured at input, depends on actual mode of operation
10	0,1,2	SensorType, tells what type of sensor the input is set-up for
11	0,1,2	SensorMode, tells what type of sensor the input is set-up for
12	0,1,2	SensorRAW i.e. the analogue value measured at the input
13	0,1,2	Sensor Boolean, returns the Boolean state of the input
14	0	RCX Watch. Integer where MSB=hours and LSB=minutes
15	0	Returns the PBMMessage stored internally in the RCX
16	-	No function with RCX

Le résultat d'une tâche Poll, dans laquelle on a demandé des données ou un état, est un nombre entier de 16 bits avec signe.

#### Exemples d'utilisation de Poll :

**Label1.Caption = PBrickCtrl.Poll 0, 7**

Label1 recueille la valeur de la variable 7 dans le RCX.

**Label2.Caption = PBrickCtrl.Poll 8**

Label2 recueille la valeur du programme effectué par le RCX.

transmises auxquelles le RCX devra réagir. La pile (c'est le terme utilisé par Lego pour désigner un sous-programme) correspondante s'exécutera si un autre RCX expédie dans son message un nombre compris dans la gamme réservée. Il peut aller de 0 à 255. Lorsque plusieurs modules RCX sont utilisés simultanément, chacun peut ainsi disposer de son domaine propre. On peut également prévoir des nombres communs pour réaliser des appels généraux, le *broadcasting*.

#### Message d'initialisation

Pour préparer une routine à recevoir des messages, il est nécessaire d'activer au préalable le bloc « reset message », sinon le RCX ne réagira pas à la commande.

#### Émission vers le RCX

Le bloc « send to RCX » provoque l'émission de données en infrarouge à destination d'autres RCX. Ce qui

va sans dire mérite parfois d'être rappelé. Pour travailler avec les options de communication, il faut naturellement compter deux blocs RCX au moins dans le projet. Oui, mais au moment d'envoyer le logiciel du PC vers un RCX donné, celui-ci doit impérativement être le seul allumé !

## Davantage d'autonomie en Visual Basic

Même si les possibilités de communication entre blocs RCX, que nous venons d'évoquer, sont disponibles dans Spirit.OCX, elles sont accessibles également en langages de programmation évolués (comme Visual Basic). Avec ces langues, il devient possible d'intégrer le PC comme acteur à part entière dans le réseau d'échange de données, non plus comme simple générateur de codes à télécharger en blocs de programmation. Nous avons vu dans les articles précédents qu'il est en mesure d'aller lire certains registres dans les RCX, de demander l'état des batteries, par exemple, ou même d'appeler les données d'une session d'enregistrement d'un RCX. Le PC ne peut malheureusement pas, à l'instar d'un pseudo-RCX, envoyer et recevoir des messages. Spirit.OCX ne dispose d'aucun équivalent du registre de messages qui équipe le RCX. En définitive, un PC ne peut aucunement faire la différence entre les blocs RCX et l'on doit donc limiter ses communications à un seul module à la fois, tout comme lorsqu'il faut télécharger du logiciel.

Mais voyons d'abord comment concevoir, à l'aide d'un langage de programmation plus évolué, un logiciel capable de faire communiquer des robots entre eux, puis nous verrons comment un PC peut parler avec des RCX.

## De RCX à RCX

La communication par messages entre blocs RCX se fonde sur trois instructions du module Spirit.OCX. Deux commandes de Spirit.OCX trouvent leur pendant dans l'environnement de programmation de Lego, il s'agit de ClearPBMMessage et SendPBMMessage, et pour le contrôle de la bonne réception, on fait appel

à la commande universelle Poll. Poll est une routine extrêmement puissante à usage multiple, elle se répète en de nombreuses circonstances dans le suivi du bloc RCX. On peut découvrir dans le **tableau 1** la manière de se servir de Poll. Elle permet de consulter les registres, de contrôler les temporiseurs, ainsi que de surveiller le fonctionnement des moteurs. Une des fonctions qu'elle peut aussi remplir, c'est de vérifier si des messages ont été reçus. Le **listage 1** contient un programme simple en Visual Basic destiné à faire des expériences.

En réalité, ce programme regroupe deux exemples qui, dans la pratique ont bien peu de chance de devoir travailler ensemble.

Le premier, « Private Sub Up\_master\_Click() », est une tâche (task) installée dans la fenêtre de programmation 0 (programme 1 sur le RCX). Il faut encore un autre RCX pour recevoir le message. On se sert alors de la routine en Visual Basic « Private Sub Up\_Slave\_Click() ». Elle place le code pour la réception d'un message dans le second RCX et utilise la fenêtre 3 (SelectPrgm 3). Cela signifie qu'il faut activer le programme au moyen du bouton logiciel gris sur le RCX comme programme 4. Cette routine est directe : après son lancement, le RCX émet un son (PlaySystemSound SWEEP\_DOWN\_SOUND), efface ensuite le registre dans lequel un message était éventuellement placé (ClearPBMMessage) et attend l'arrivée d'un message et la même rengaine se répète.

Il ne s'agit là bien sûr que d'un exemple élémentaire, mais on peut sans difficulté l'étoffer pour en faire une application plus judicieuse. Ici, nous avons vu deux tâches exécutées à la suite l'une de l'autre, mais comme le RCX est capable de s'occuper de plusieurs tâches en même temps, il est également possible de traiter les messages dans une tâche séparée, en parallèle avec la principale. La structure du programme doit alors en tenir compte et le listage 2 donne à voir comment s'y prendre en pareille hypothèse.

En fin de compte, il est encore possible, naturellement, de guetter l'arrivée des messages et, au gré de leur contenu, de leur donner une suite

spécifique par le lancement de la tâche appropriée.

## Entre PC et RCX

Nous avons déjà indiqué que le PC

n'est pas un interlocuteur comme les autres dans les conversations entre blocs RCX. La difficulté repose sur le fait qu'un RCX ne peut pas prendre l'initiative d'expédier un message à destination du PC. Mais si le PC, à des moments déterminés, vient voir subreptice-

## Listage 1.

```
Private Sub Timer1_Timer()
Rem si le label <> 0, le RCX a envoyé le message
Label1.Caption = Str(PBrickCtrl.Poll(VAR, 0))
End Sub

Private Sub Up_Slave_Click()
Rem télécharger le logiciel pour l'esclave
With PBrickCtrl
    Rem sélectionner la fenêtre 4 du logiciel
    .SelectPrgm 3

    .BeginOfTask MAIN
        Rem signaler acoustiquement le démarrage du logiciel
        .PlaySystemSound SWEEP_DOWN_SOUND
        Rem par sécurité, effacer le message
        .ClearPBMMessage
        Rem et attendre un message d'un autre RCX
        .While PMESS, 0, EQ, CON, 0
            Rem petite boucle d'attente de 50 ms
            .Wait CON, MS_50
        .EndWhile
        Rem message reçu !
        .PlaySystemSound SWEEP_DOWN_SOUND
    .EndOfTask
End With

End Sub

Private Sub Up_master_Click()
With PBrickCtrl
    .SelectPrgm 0
    .BeginOfTask MAIN
        .ClearPBMMessage
        Rem par sécurité, mettre var 0 à zéro
        .SetVar 0, CON, 0
        Rem envoyer le message n° 11
        .SendPBMMessage CON, 11
        Rem rendre la variable 0 égale à 11
        Rem le PC voit, par le temporisateur, si le message a été
        envoyé
        .SetVar 0, CON, 11
    .EndOfTask
End With
End Sub

Private Sub Form_Load()
    PBrickCtrl.InitComm
End Sub
```

*N.B. Ce logiciel utilise le module RCXDATA.BAS, dans lequel les constantes (MAIN, EQ, CON, etc) sont déclarées et initialisées. Veiller donc à ce que les liens avec ce module soient établis.*

Listage 1. Un programme élémentaire pour analyser l'émission de messages.

## Listage 2.

*:REM structure d'un programme à tâches parallèles*

```
PBrickCtrl.BeginOfTask MAIN
```

```
PBrickCtrl.StartTask 1
```

```
PBrickCtrl.StartTask 2
```

```
PbrickCtrl.EndOfTask
```

*:REM definition of tasks within MAIN*

```
PBrickCtrl.BeginOfTask 1
```

```
.....
```

```
.....
```

```
.....
```

```
PBrickCtrl.EndOfTask
```

```
PBrickCtrl.BeginOfTask 2
```

```
.....
```

```
.....
```

```
.....
```

```
PBrickCtrl.EndOfTask
```

fournit au PC la période d'une seconde, tandis que la commande Poll interroge pour connaître la contenu de VAR 0, par l'instruction : PBrickCtrl.Poll (VAR, 0). Elle obtient ainsi une chaîne de texte qu'elle attribue à Label1 pour ensuite l'afficher à l'écran.

Certains lecteurs se demandent probablement ce que l'on cherche à obtenir ainsi. Bien, disons que le PC est incapable de recevoir directement le moindre message émanant d'un RCX. La méthode que nous venons de montrer par un exemple, dans lequel le contenu du message est également communiqué à une variable auxiliaire, rend possible la vérification de ce qui se passe dans un RCX. On peut utiliser cette fonction pour conférer à un RCX une nouvelle tâche à exécuter dès que la précédente est terminée ou par

exemple, après lecture du RCX, entamer une nouvelle session.

On comprend mieux maintenant pourquoi, dans le premier exemple, les deux tâches choisies ne peuvent pas s'effectuer en même temps. Si le PC interroge pour connaître la valeur de la variable auxiliaire et qu'il y a plusieurs RCX en service, aucun d'eux ne peut savoir lequel doit expédier sa variable 0. Chacun va donc réagir en émettant la sienne et en fin de compte, le résultat que le PC en obtiendra ne sera pas fiable. Les deux exemples ne doivent servir que d'illustration de la façon de faire communiquer PC et RCX ou ceux-ci entre eux. Mais au moins, pourrez vous sans souci appliquer ces principes dans un véritable environnement de robotique.

(000040-4)

Listage 2. Modifier l'agencement du logiciel permet de tenir à l'œil le trafic des messages tout en exécutant la tâche principale.

ment si le RCX n'a pas, par hasard, quelque chose à transmettre, l'empêchement est levé. Comprendons-nous bien, il n'est question ici que d'un seul RCX, le stratagème ne s'applique pas à un système dans lequel plusieurs RCX sont actifs.

L'astuce tient dans l'emploi d'une variable auxiliaire dans le RCX (dans l'exemple cité : VAR 0). Au démarrage de la routine, elle affiche la valeur 0. Après l'envoi d'un message, la variable prend la même valeur que le message lui-même. C'est à la ligne SetVar 0, CON, 11 que cela se passe. Le nombre 11 (il est arrivé par hasard, il aurait pu être différent dans la même application) correspond au contenu du message. Dans la ligne de programme correspondante, c'est finalement la commande SendPBMMessage CON, 11 qui est définie.

Le PC utilise un temporisateur pour vérifier à chaque seconde la valeur de VAR 0. La manière dont cela se produit, nous la découvrons dans les premières lignes du programme. La routine Private Sub Timer1\_Timer() s'en charge. C'est Timer1 qui

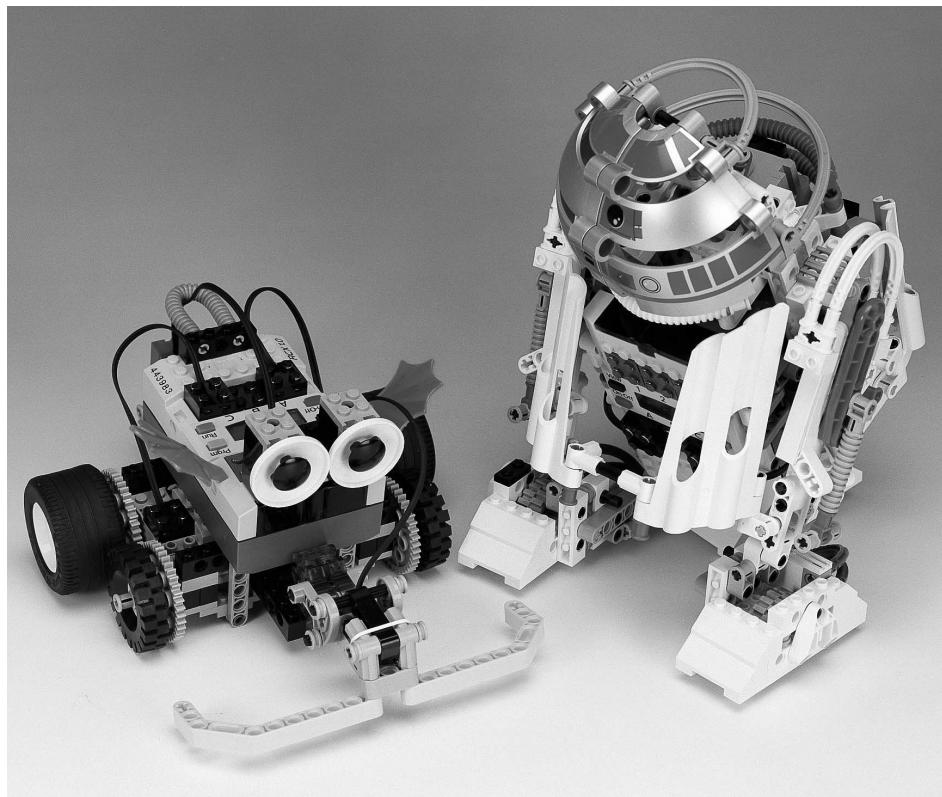


Figure 3. C'est grâce à l'échange de PBMessages que les blocs RCX peuvent converser entre eux. Ici, R2D2 communique avec le robot qui nous occupera dans le prochain numéro.

# L'USB

## Partie I : développement d'applications personnelles

Par B.Kainka

L'interface USB décrite dans le présent numéro prouve, si besoin en était, que rien n'interdit de mettre l'USB à contribution pour des applications personnalisées ayant trait à la mesure, la commande (automates) ou la régulation. Ceci n'empêche pas que l'on se pose un certain nombre de questions telles que, comment développer un tel appareil, quel cahier des charges faut-il respecter, où trouver les informations indispensables ?

Comme le suggère son nom, l'USB (*Universal Serial Bus*) est un bus série auquel il est possible de connecter plusieurs périphériques et par le biais duquel il est possible de les adresser. L'USB est sensiblement plus complexe que RS-232 mais avec un taux de transfert de 1,5 Mbit/s (périphériques Faible Vitesse) ou de 12 Mbits/s (périphériques Grande Vitesse) également notablement plus rapide. À cela s'ajoute que les concepteurs de l'USB ont attaché une importance capitale aux 2 aspects primordiaux que sont la simplicité et à la sécurité de mise en oeuvre. Ce n'est qu'avec l'arrivée de P&P (*Plug&Play*) que l'USB a trouvé sa vraie place. En effet, normalement, il suffit de connecter un périphérique USB au PC, qu'il soit à l'arrêt (connexion à froid) ou en fonction (connexion à chaud) pour qu'il s'initialise automatiquement (si tant est que le PC soit en route ou lors de sa mise en fonction ultérieure).

Finis tous les problèmes avec les interruptions déjà occupées, les erreurs d'adresse et les pilotes (*drivers*) absents.

Cette simplification au niveau de l'utilisation se traduit par une augmentation sensible de la complexité de l'électronique en ce qui concerne le concepteur d'un appareil devant être branché à un port USB. Il vous faudra, s'il est dans vos intentions de développer un tel appareil, assimiler la matière complexe qu'est l'USB et ses normes, programmer un micro-

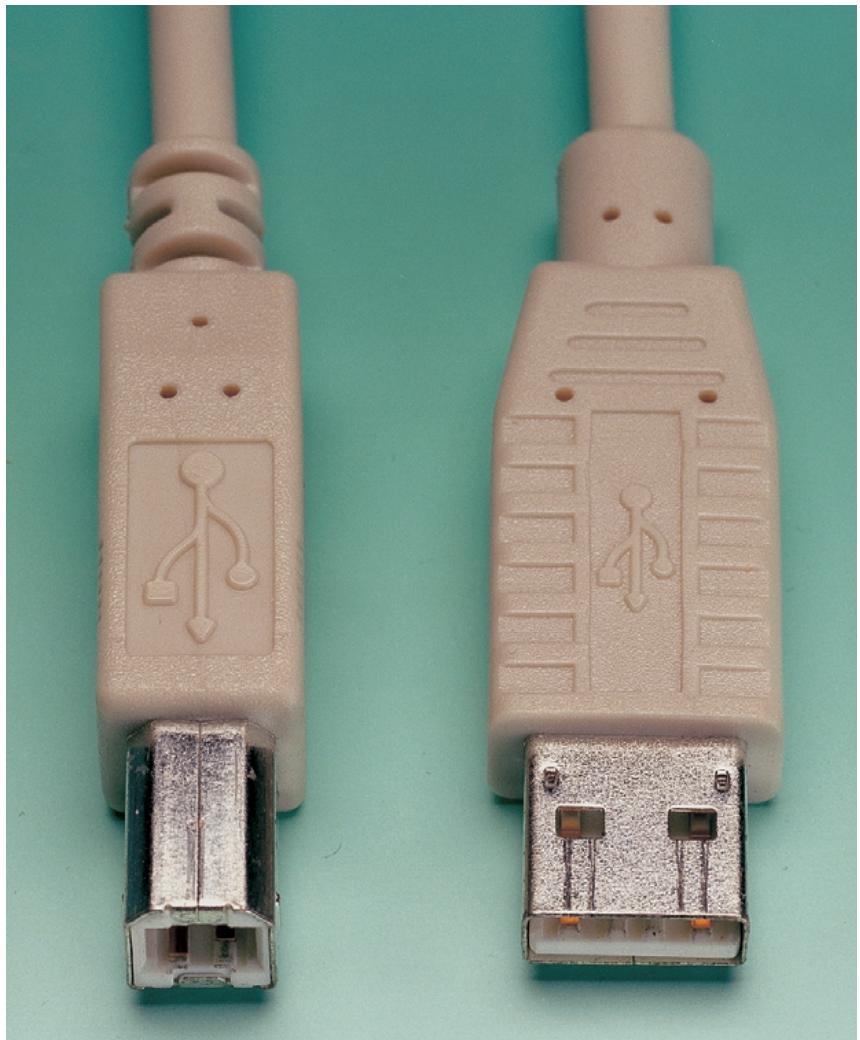


Figure 1. Physionomie des connecteurs USB.

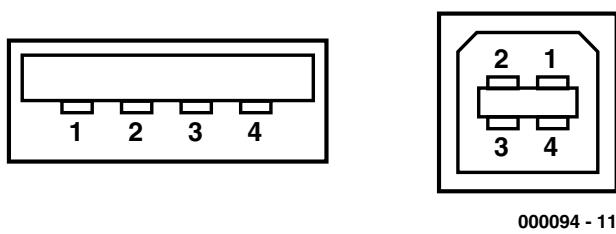


Figure 2. Brochage des connecteurs de type A et de type B.

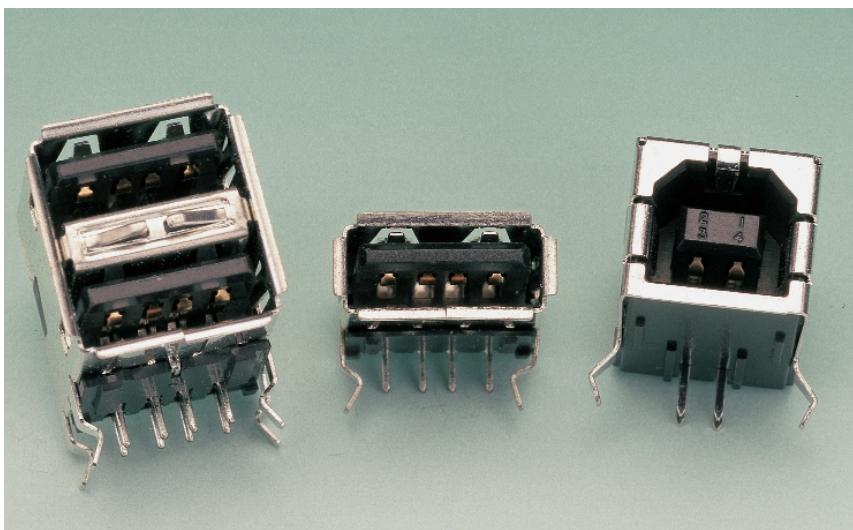


Figure 3. Aspect des embases USB de type B.

contrôleur USB et écrire le pilote requis pour le fonctionnement correct de l'appareil. L'encadré vous donne quelques points de repère et autres sources d'informations.

## Câble USB et alimentation

La norme USB connaît 2 types de connecteurs, le type A et le type B. Le système a été conçu de manière à interdire toute inversion malencontreuse. Contrairement à ce qui est le cas avec l'interface RS-232, il n'existe pas de différence entre une liaison directe et une liaison croisée. Les interconnexions sont se font toujours 1:1 et sont toujours les mêmes :

- 1 +5V
- 2 Data -
- 3 Data +
- 4 Masse

On trouve, sur l'arrière des PC modernes (voire sur le devant des PC d'avant-garde) une paire d'em-

Full-Speed, un câble Low-Speed ne pouvant qu'être fixé à l'origine sur un périphérique sortant d'usine.

Il existe, en cas de besoin d'établissement d'une liaison, des câbles-rallonge du type A-A.

S'il est impossible de mettre la main sur des connecteurs USB, il existe par contre dans le commerce des embases USB encartables (à monter sur une platine). Il devient partant possible de procéder à ses propres expérimentations. L'embase USB met à disposition une tension d'alimentation de +5 V dont on peut drainer, sans risque de problème, jusqu'à 100 mA. Ce courant permet d'alimenter une minuscule lampe de chevet, mais pourra permettre bien d'autres applications plus utiles. Nombreux sont les montages numériques et les microcontrôleurs à travailler avec une tension de 5 V. Pour peu qu'un tel montage soit connecté à un PC on dispose du même coup de sa tension d'alimentation. Il faudra veiller impérativement à la mise en place d'une protection contre un court-circuit efficace. On pourra utiliser à cet effet tout aussi bien un fusible classique qu'un fusible Polyswitch (réarmable).

Les 2 lignes de données, D+ et D-, ne sont utilisables qu'en association avec des composants USB spéciaux, à savoir des microcontrôleurs prévus à cet effet. Ces circuits intégrés peuvent être alimentés directement par l'embase USB. Si l'on a pris la précaution d'effectuer une désignation spécifique (annonce) d'un périphérique donné, il lui sera possible de consommer jusqu'à 500 mA maximum.

La tension d'alimentation disponible sur une sortie USB peut atteindre jusqu'à 5,25 V (à vide) mais elle peut également descendre, en cas de charge importante, jusqu'à 4,2 V. Même dans ces conditions limites un régulateur de tension peut fournir un 3,3 V stable. L'ensemble du système, câble et périphérique, est conçu de manière à ce que la tension d'alimentation ne tombe pas en dessous

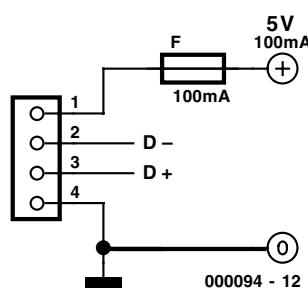


Figure 4. La sortie USB en tant qu'alimentation régulée.

de 4,2 V et ce même à charge maximale. Tout périphérique requérant un courant supérieur à 100 mA doit en faire mention auprès du système, ce dernier n'en autorisant la mise en circuit qu'à condition de disposer de réserves suffisantes.

Tout cela explique que l'on fait la distinction entre un périphérique doté de sa propre alimentation (*self powered*) et celui dont l'alimentation se fait par le biais de l'USB (*bus powered*). Il est possible, dans bien des cas, d'opter pour l'un ou l'autre de ces 2 modes d'alimentation. Le périphérique dispose dans ce cas-là d'un jack d'alimentation que l'on pourra, si nécessaire, connecter à un adaptateur secteur externe. Les spécifications USB stipulent que le drain de courant au niveau du bus est limité automatiquement. Partant, lorsqu'un périphérique draine un courant supérieur à la valeur autorisée, son alimentation devrait être coupée.

## Classes USB

Jusqu'à tout récemment on recevait, lors de l'achat d'une souris pour PC, une disquette (ou CD-ROM) comportant le pilote (*driver*) correspondant. Si l'on achète aujourd'hui une souris USB on sera peut-être surpris de constater l'absence de ce support. Une fois que l'on se sera décidé, après quelques instants de désarroi, à brancher la souris au PC on aura la (bonne) surprise de constater que le système d'exploitation Windows 98 trouve de lui-même le pilote. Il charge le pilote HID (*Human Interface Device*) prévu tout particulièrement à cet effet. La souris fait en effet partie d'une classe de périphériques définie pour lesquels il existe des pilotes standard tout faits.

On peut mettre au nombre des périphériques HID les souris, les claviers, les pointeurs et autres manches de commande (*joystick*). Il existe, outre la classe des périphériques HID, d'autres classes UUUSB pour les périphériques son, les imprimantes et bien autres encore. Tous les périphériques « classiques » se trou-

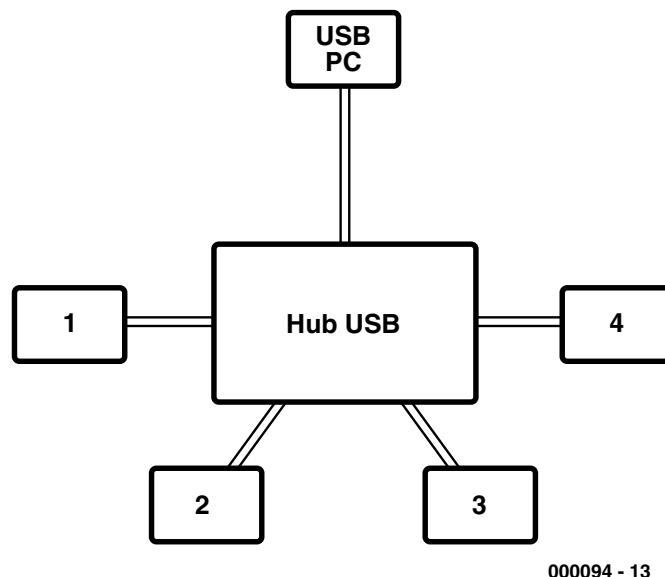


Figure 5. Structure en étoile d'un système USB basé sur un Hub.

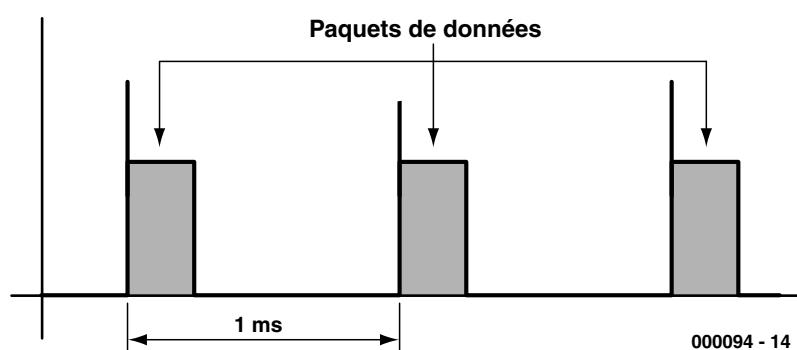


Figure 6. Paquets de données à l'intérieur des fenêtres de 1 ms.

vent ainsi subdivisés en classes pour lesquelles le système dispose de pilotes adéquats. Cela se traduit par une standardisation de fait vu que les fabricants de périphériques doivent respecter à la lettre les normes des définitions de classes.

Les périphériques USB standard deviennent de plus en plus abordables. On peut en arriver à se demander s'il n'y a pas moyen d'en utiliser l'un ou l'autre à des fins « détournées ». Un exemple : Un périphérique son USB s'installe

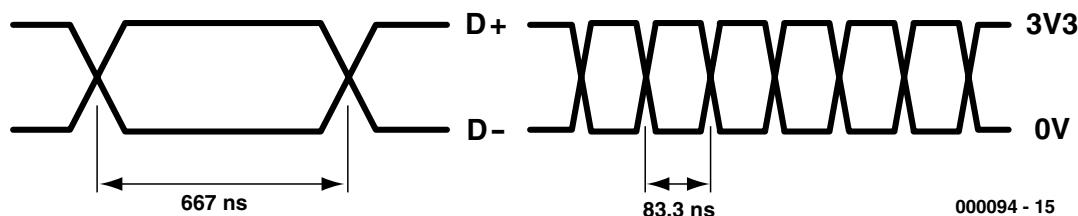


Figure 7. Signaux Low-speed et High-speed.

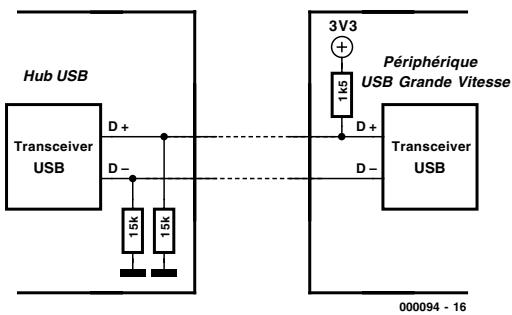


Figure 8. Reconnaissance d'un périphérique Full-speed.

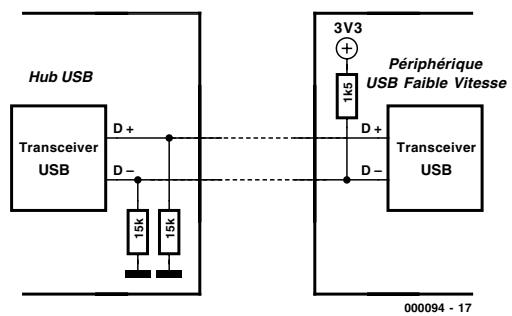


Figure 9. Détection d'un périphérique Low-speed.

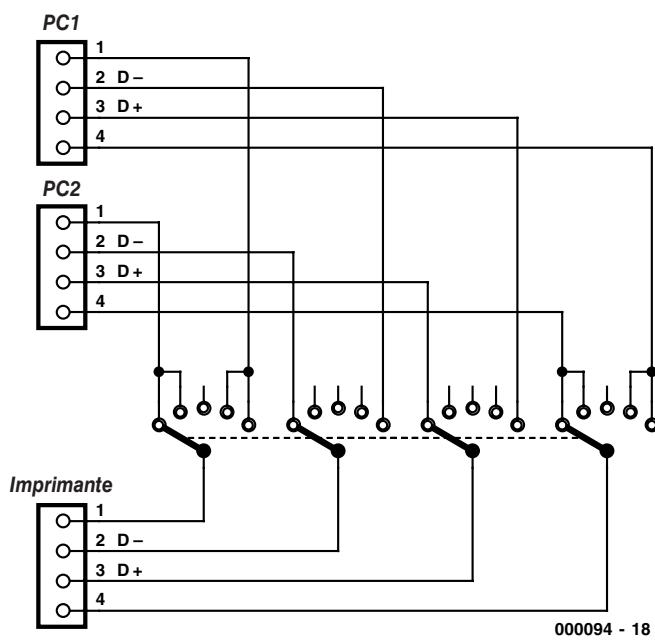


Figure 10. Commutateur USB pour imprimante.

dans le système de manière à pouvoir être utilisé comme le serait une carte-son interne classique. C'est d'ailleurs là le but de la manœuvre : tous les programmes existants doivent en effet tourner avec le dit périphérique USB. Il en va de même pour les programmes que l'on aura écrits soi-même. Nous avons procédé à un test couronné de succès en utilisant le fichier PORT.DLL proposé dans le livre « Piloter ses interfaces PC sous Windows ». Il est donc possible de faire tourner des programmes que l'on aura écrits en Delphi avec un périphérique USB. Nous n'avons cependant pas encore suffisamment d'expérience à ce niveau pour pouvoir affirmer sans autre forme de procès que les choses se passent toujours bien et que cela fonctionne à tous les coups.

Un essai effectué sur une manette de jeux USB nous apprit la chose suivante : un port manettes USB s'installe, entre autres, à l'aide d'une émulation DOS. Il devient possible ainsi d'accéder directement à des adresses de port virtuelles d'une carte de manettes de jeu. Certains de programmes prévus pour le port manette de l'ouvrage évoqué plus haut tournent également avec la variante USB. On dispose dans ces conditions d'une interface USB rudimentaire dotée de 4 entrées analogiques et de 4 entrées numériques.

## Topologie de bus

L'USB est un bus en étoile comportant un maître. On a besoin, pour la connexion de plusieurs périphériques USB, de ce que l'on appelle un Hub. Un Hub est un répartiteur de bus disposant de plusieurs ports. Il doit son nom au terme anglais hub qui désigne un moyeu, image parlante pour représenter les rayons d'une roue. Normalement, un Hub externe dispose d'un port montant (*upstream*) et de 4 ports descendants (*downstream*). Notons que le PC lui-même dispose déjà d'un Hub en vue de créer les 2 sorties USB dont il dispose. Ce Hub dit Hub-racine (*Root-Hub*) se trouve sur la carte-mère.

Il est possible de connecter un nouveau Hub au port descendant d'un Hub. Il est possible ainsi de connecter en cascade jusqu'à 7 Hubs, ce qui nous donne un nombre maximum de 127 périphériques. Il s'agit là d'une valeur théorique sachant que les différents périphériques connectés au système doivent se partager la bande passante totale du bus.

### Signaux USB

Les signaux véhiculés par les 2 lignes D+ et D- sont des signaux différentiels présentant des niveaux de tension de 0 et 3,3 V. Le microcontrôleur connecté au système s'alimente bien souvent à 3,3 V. L'USB est un bus ne comportant qu'un maître, la totalité des

## Adresses où trouver des informations

[www.usb.org](http://www.usb.org) : « siège » des concepteurs de l'USB, c'est-à-dire lieu de réunion des firmes importantes ayant défini le standard USB. Le matériau le plus intéressant et le plus important sont les spécifications USB, normes disponibles au téléchargement sous la forme de fichiers au format .pdf (Acrobat Reader). On y trouve tout ce dont on peut avoir besoin. Il existe heureusement d'autres sources d'information pour corroborer ce que l'on trouve ici.

Le Starterkit de Cypress ([www.cypress.com](http://www.cypress.com)) destiné au microcontrôleur USB CY7C63000 présentant un thermomètre USB comme exemple d'application. Ce kit de d'évaluation bon marché n'est malheureusement plus disponible. La petite interface USB décrite dans Elektor a été développée à partir de ce thermomètre USB. Ceux qui n'ont pas pu mettre la main sur le Starterkit de Cypress se voient offrir une seconde chance.

Craig Peacock ([www.beyondlogic.org](http://www.beyondlogic.org)) a écrit son propre pilote pour le thermomètre USB. Il offre un coup d'œil déterminant au cœur d'un pilote USB et en documente la structure. Cette page présente un intérêt indiscutable en ce qui concerne le thème USB.

Le Starterkit de Anchor-Chips ([www.anchorchips.com](http://www.anchorchips.com)) (à trouver actuellement sous la houlette de Cypress, [www.cypress.com](http://www.cypress.com)) pour le EZ-USB (AN2131). Il s'agit d'un processeur compatible 8051 doté de RAM interne et d'un noyau USB. Il est possible de télécharger des programmes 8051 directement dans la RAM du processeur par le biais de l'interface USB et de les démarrer. Le noyau USB soulage le concepteur à un point tel qu'il lui sera possible d'écrire des applications USB intéressantes à l'aide de minuscules programmes de quelques octets seulement.

On trouvera sur la page d'accueil de l'auteur mise à l'adresse (<http://home.t-online.de/home/B.Kainka>) des informations additionnelles et une bibliographie détaillée.

activités se faisant sous la houlette du PC et depuis ce dernier. L'émission et la réception des données se font par paquets comprenant entre 8 et 256 octets. Le PC peut demander à un périphérique de lui envoyer des données, mais aucun périphérique n'est en droit ni en mesure d'ailleurs, de lui-même, d'envoyer des données.

L'ensemble du transfert de données se fait à l'intérieur d'un intervalle d'une milliseconde très exactement. Il est possible, à l'intérieur de cette fenêtre, de traiter des paquets de données destinés à différents périphériques. Rien n'interdit de juxtaposer des paquets Grande et Faible Vitesse. En cas d'accès à différents périphériques c'est la tâche d'un répartiteur de bus (*Hub*) d'assurer l'aiguillage des données.

Les périphériques Low-speed travaillent à un taux de transmission de 1,5 Mb/s, ce qui donne une longueur de bit de 666,7 ns très exactement. Les liaisons Full-speed tra-vailtent, elles, à 12 Mbits/s ce qui correspond à une longueur de bit de 83,33 ns. C'est au maître de décider de la vitesse de transmission, les esclaves devant faire en sorte de se synchroniser sur le flux de données. En raison de l'absence de signal d'horloge distinct, il faudra dériver l'horloge du signal de données. On utilise pour cela le processus NRZI (*Non Return to Zero*). Les données se trouvant à zéro entraînent un changement de niveau, les données à 1 n'entraînant pas de changement d'état.

Un périphérique USB dispose, en règle générale, de plusieurs FIFO (*First In First Out*) qui servent à la transmission de données. L'adresse de périphérique est complétée par une adresse de destination (*Endpoint*) qui indique où doivent arriver les données ou où

il faut aller les chercher. Ainsi, une souris USB possède toujours un Endpoint 0 et un Endpoint 1. On utilise le Endpoint 0 pendant l'initialisation. Le microcontrôleur transfert, à intervalle défini, les données utiles proprement dites vers le FIFO Endpoint 1 d'où elles sont transmises au PC.

Le logiciel USB crée des tuyaux (*pipe*) vers les différents Endpoints. Un tuyau est un canal logique vers un Endpoint d'un périphérique. On peut s'imaginer un tuyau sous la forme d'un canal de données constitué par un unique conducteur.

En fait, les données d'un tuyau sont transmises, dans la fenêtre de 1 ms, sous la forme de paquets de données et distribuées, en fonction de leur adresse terminale, sur différents emplacements de mémoire réels. Un périphérique peut utiliser plusieurs tuyaux simultanément, ce qui se traduit par une augmentation substantielle du taux de transmission d'ensemble.

### Énumération

Une liaison USB inutilisée n'est pas active, le Hub n'envoie pas de fenêtre de données. Les 2 lignes de signal se trouvent au niveau bas et présentent une résistance interne de 15 kΩ. Chaque périphérique USB comporte une résistance de 1,5 kΩ qui force l'une des 2 lignes de signal au +3,3 V. Dans le cas d'un périphérique Full-speed c'est la ligne D+,

dans celui d'un périphérique Low-speed, ce sera la ligne D-. Ceci permet au Hub de reconnaître le type de périphérique auquel il a affaire et partant d'établir une liaison de données au taux de transmission correspondant.

Lors de l'énumération d'un périphérique on a chargement automatique du pilote requis. À l'inverse, le système d'exploitation détecte la déconnexion d'un périphérique du bus. Son pilote est alors éliminé de la mémoire. Il est possible ainsi d'utiliser un périphérique USB avec plusieurs PC. Il suffit de déconnecter le câble du premier PC pour l'enficher dans l'embase USB du second. Une nouvelle énumération prend alors place. Il est possible ainsi, sans problème, d'utiliser une seule imprimante avec 2 PC différents (un PC de bureau et un portable par exemple).

### Un commutateur USB

Il existe, dans le commerce, pour les autres types d'interfaces connus, des commutateurs évitant d'avoir à effectuer ce genre de manipulations de déconnexion/reconnexion. De par le faible nombre de lignes mises en jeu, la réalisation d'un commutateur USB est une affaire rapidement menée. Son comportement devra être celui d'une déconnexion/reconnexion manuelle de câble. L'important est de veiller à ce que l'on ait, lors de la commutation, une interruption de la tension d'alimentation

en provenance du PC pendant une durée suffisante de manière à permettre au périphérique de procéder à une réinitialisation (*reset*). Le microcontrôleur interne doit redémarrer pour pouvoir effectuer une nouvelle énumération. Le commutateur dispose pour cette raison d'une position centrale neutre dans laquelle l'imprimante n'est connectée à aucun des PC.

Il reste à tenir compte d'un petit point. Un examen attentif d'un connecteur USB montre que les contacts 1 et 4 destinés respectivement à la tension d'alimentation et à la masse sont légèrement plus longs que les contacts des lignes de signal, 2 et 3. Ainsi, lors de la mise en place du câble, ces broches assurent le contact en premier. Ainsi, le périphérique USB concerné de trouve déjà sous tension lors de l'établissement du contact au niveau des lignes de signal. Cette approche diminue très sensiblement les problèmes dus aux

charges statiques et à la circulation, entre différents périphériques, de courants d'équilibrage de potentiel dangereux. Les crêtes de courant naissant lors de la connexion ne peuvent constituer de risque pour les lignes de signal. C'est pour la même raison d'ailleurs que le câble parallèle d'une imprimante est doté d'un capuchon métallique qui entre en premier en contact avec la masse de l'ordinateur.

Il serait possible d'éviter ces problèmes dus aux courants d'équilibrage par une interconnexion permanente des masses au cœur du commutateur USB. Cette approche a cependant l'inconvénient de créer une gigantesque boucle de masse par le biais des lignes de terre des PC connectés au système. Cela pourrait se traduire, dans des conditions défavorables, par des parasites plus importants encore et faire perdre toute fiabilité à l'ensemble du système. On préférera éviter cette solu-

tion et assurer une commutation des 4 lignes. On pourra reproduire le comportement introduit par les contacts allongés par l'utilisation d'un commutateur à 5 positions. On commencera, lors du basculement d'un système à l'autre, par connecter la tension d'alimentation avant d'interconnecter les lignes de données D+ et D-. La position centrale reste neutre. La nécessité ou non d'introduire une pause de 1 seconde dépend des caractéristiques du périphérique USB connecté.

Il s'agit dans le cas présent d'une approche expérimentale que nous n'avons pas encore pu tester de façon exhaustive. Il se peut, en fonction de l'imprimante concernée, que la vitesse de commutation d'une position à l'autre soit très variable voire s'il est effectivement nécessaire de respecter une pause d'une durée donnée en position médiane du commutateur. Il ne devrait pas être très complexe, à l'aide d'une paire de relais et d'un rien d'électronique, d'automatiser l'ensemble du processus de commutation.

(000094-I)

*(Suite au prochain numéro)*