



Démultiplexeur
DMX 8 voies



Interface disque
dur pour port
parallèle

Émetteur-
récepteur
IR pour PC

Récepteur
GPS à sortie
NMEA

Alim de labo
numérique

Apprenez à
programmer avec la

CARTE À 89S8252 FLASH



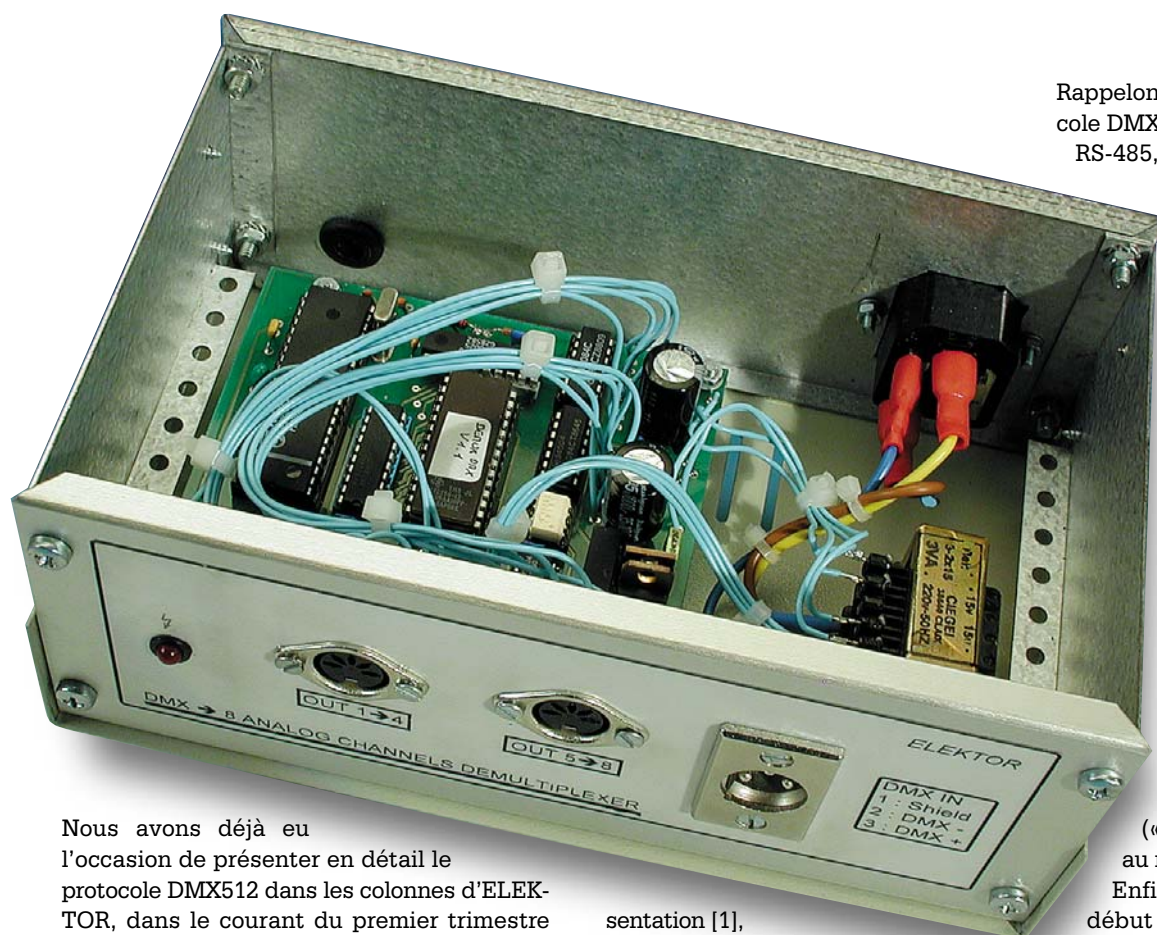
Démultiplexeur DMX 8 voies

8 canaux DMX = $8 \times 0 \text{ à } 10 \text{ V}$ (et bientôt 8 relais)

Projet : Benoît Bouchez

bbouchez@netcourrier.com

L'acronyme DMX512 recouvre les termes Digital MultipleX, technique consistant à envoyer 512 canaux successivement (« multiplexés ») sur le même câble. Le présent montage a pour fonction de permettre le pilotage d'appareils commandables par DMX512.

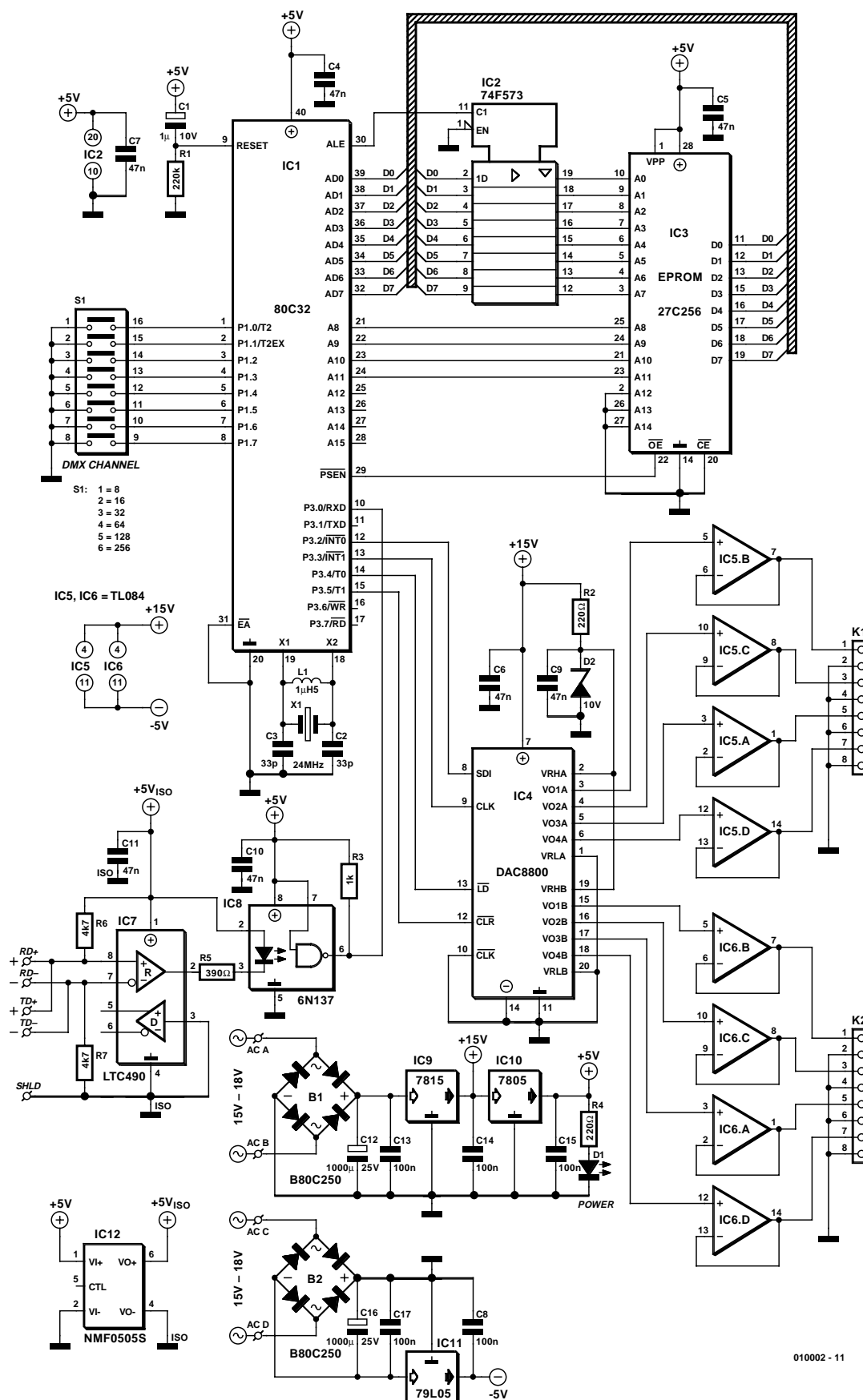


Nous avons déjà eu l'occasion de présenter en détail le protocole DMX512 dans les colonnes d'ELEKTOR, dans le courant du premier trimestre 2001. Nous invitons le lecteur friand de détails techniques à se reporter à cet article de pré-

sentation [1], ainsi qu'à l'article concernant l'interface MIDI<->DMX [2].

Rappelons simplement que le protocole DMX512 fait appel à une liaison RS-485, permettant de connecter jusqu'à 32 équipements sur 1 000 mètres. Les données sont transmises à 250 kbauds, en format asynchrone 8 bits de données, sans parité, 1 bit de début (start), 2 bits d'arrêt (stop). Les informations des différents canaux sont transmises successivement, sous la forme d'une valeur sur 8 bits, entre 0 (noir) et 255 (plein feu). Pour permettre de retrouver le début de la séquence des 512 valeurs, on provoque simplement une interruption de communication, en générant un Break (« 0 » logique) d'une durée au moins égale à 2 caractères. Enfin, pour retrouver le bit de début du premier octet, on provoque une remontée d'au moins 8 ms en fin de Break.

Définie par l'USITT (organisme américain chargé de concevoir et de diffuser les normes applicables pour les théâtres) en août 1986, le système DMX512 n'a connu son véritable envol qu'au milieu des années 1990, avec la démocratisation des projecteurs automatisés.



010002 - 11

Figure 1. Le schéma du démultiplexeur DMX est en fait un système micro réduit à sa plus simple expression.

Un démultiplexeur, pour quoi faire ?

Comme nous le disions plus haut, DMX512 est l'acronyme de *Digital Multiplex*, puisque les 512 canaux sont envoyés successivement (sous forme « multiplexée ») sur le même câble.

Les produits équipés d'une interface DMX512 d'origine sont en général plus chers que leurs homologues à télécommande analogique. De même, certains équipements n'ont aucune raison de disposer d'une interface DMX (ou autre) intégrée. C'est notamment le cas des projecteurs utilisés en théâtre ou dans les studios de TV, pour lesquels on ne commande que le niveau de lumière. On peut également citer les effets spéciaux comme les stroboscopes ou les générateurs de fumée.

Dans le cas d'équipements aussi simples, soit on intègre une simple télécommande 0 à +10 V, soit la télécommande est reportée au niveau de gradateurs externes (« *dimmer pack* »).

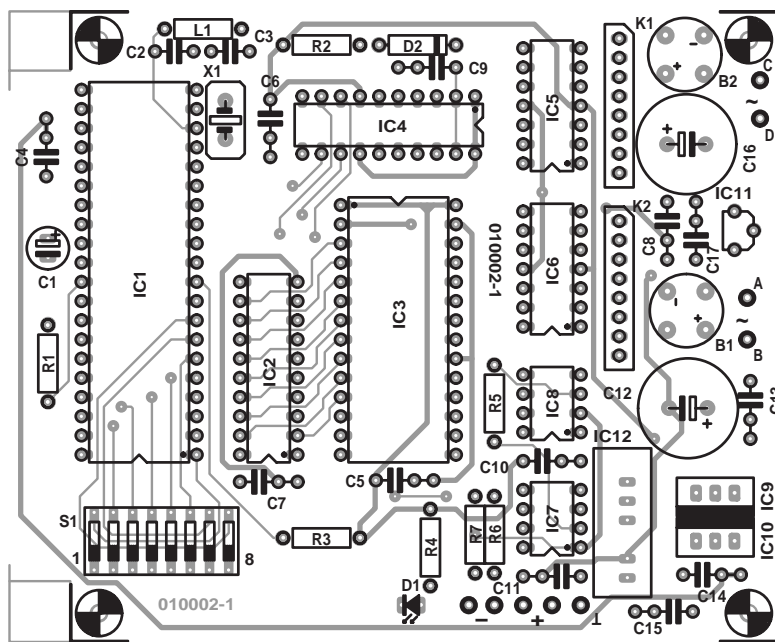
Le rôle d'un démultiplexeur est de fournir un ensemble de tensions de commande permettant à des équipements du genre de ceux cités ci-dessus, dont la commande est jusque là assurée par tension 0 à 10V, d'être commandables par DMX512. Notre démultiplexeur est également capable de recevoir une extension à relais, pour la commande d'équipements simples, de type Tout ou Rien, sans entrée analogique donc.

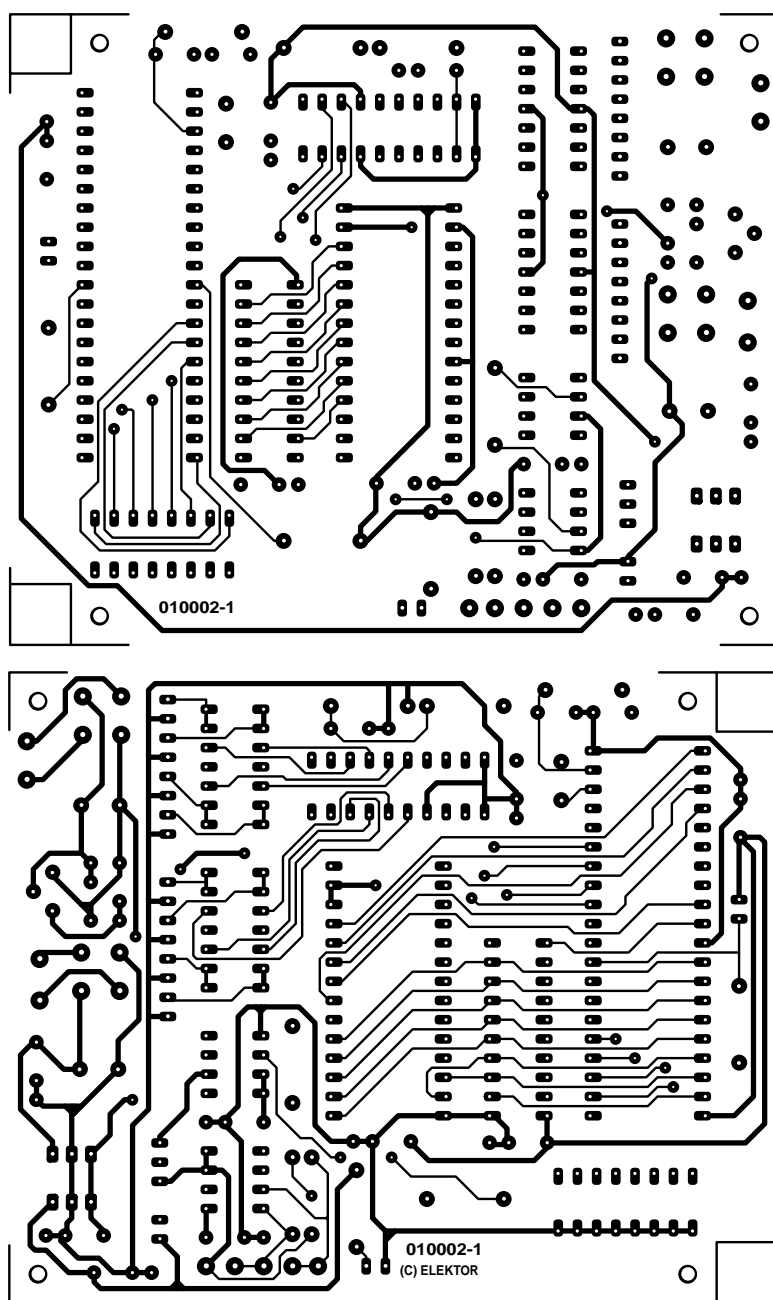
Avec un peu d'astuce, il est même possible de modifier des produits non équipés de télécommande de façon à y intégrer notre démultiplexeur, et de les transformer en produits quasi-professionnels, mais chaque chose en son temps...

Passons aux choses sérieuses...

Comme le montre un examen rapide de l'électronique dont le schéma est représenté en **figure 1**, c'est encore une fois un microcontrôleur qui est à l'honneur. Nous avons utilisé ici un 80C32 tout à fait classique, avec mémoire programme externe sous forme d'EPROM (nous aurions d'ailleurs tout à fait pu utiliser une version de processeur à EPROM intégrée, ce qui aurait encore plus simplifié le circuit, mais nous prévoyons toujours la possibilité d'une future évolution du logiciel).

Comme d'habitude dans ces cas-là, le bus d'adresse est démultiplexé par IC2, qui fournit les 8 bits de poids faible à l'EPROM IC3. Vous remarquerez que nous n'utilisons pas toutes les broches d'adresse de cette dernière. En effet, le programme est assez compact et se contenterait allégrement d'une





Liste des composants

Résistances :

R1 = 220 k Ω
 R2, R4 = 220 Ω
 R3 = 1 k Ω
 R5 = 390 Ω
 R6, R7 = 4k Ω 7

Condensateurs :

C1 = 1 μ F/10 V radial
 C2, C3 = 33 pF
 C4 à C7, C9 à C11 = 47 nF
 C8, C13 à C15, C17 = 100 nF
 C12, C16 = 1 000 μ F/25 V radial

Semi-conducteurs :

B1, B2 = B80C250
 D1 = LED
 D2 = diode zener 10 V/500 mW
 IC1 = P80C32 SFPN (40 broches DIL 0-70°)
 IC2 = 74F573 ou 74HCT573 *
 IC3 = 27C256 (programmée **EPS010002-21**)
 IC4 = DAC8800 FP (Analog Devices)
 IC5, IC6 = TL084
 IC7 = LTC490 CN8 (Linear Technology)
 IC8 = 6N137
 IC9 = 7815
 IC10 = 7805
 IC11 = 79L05
 IC12 = NMF0505S, TMA0505S

Divers :

K1, K2 = embase à 1 rangée de 8 contacts
 L1 = self compacte 1 μ H5*
 S1 = octuple interrupteur DIL
 X1 = quartz 24 MHz
 transfo secteur 3VA 2 x 15 V
 entrée secteur avec mise à la terre et fusible 25 mA
 boîtier 200 x 80 x 132 mm tel que, par exemple, Telet LC270
 * cf. texte

à la portée des 80C32. Notre programme reconnaît en fait simplement l'état du 9^{ème} bit de chaque octet. Comme les trames DMX512 sont formées sur des octets sans parité, ce 9^{ème} bit est en fait l'image du bit d'arrêt, qui donc doit être à 1. Si le processeur détecte un octet dont le 9^{ème} bit est à 0, c'est que la liaison DMX est en phase de synchronisation. Même si cet algorithme est peu orthodoxe, il fonctionne parfaitement bien (nous l'utilisons depuis plusieurs années sur des conceptions professionnelles sans aucun problème).

Concernant la périphérie du 80C32, il n'y a pas grand chose à dire, tant le schéma est classique. On soulignera cependant l'utilisation d'un quartz 24 MHz, nécessaire pour obtenir une vitesse de 250 kbauds sur le port série. En général, il est assez rare de trouver des quartz 24 MHz en fondamentale. Nous avons donc ajouté une petite self L1 pour forcer le quartz en régime harmonique. Nous verrons dans la partie consacrée à la réalisation pratique la méthode pour déterminer s'il vous faut la self ou non. Le port P1 est utilisé pour la lecture

des cavaliers de sélection d'adresse. Comme notre démultiplexeur fournit 8 consignes de tension, il occupe de ce fait 8 canaux DMX. La configuration par S1 se fait « modulo 8 », c'est à dire que l'interface ne peut être placée que tous les 8 canaux DMX (1, 9, 17, 25, etc.). Pour déterminer le canal DMX de « départ », il suffit d'additionner les valeurs représentées par chaque cavalier fermé et d'ajouter 1. Par exemple, une interface sur laquelle tous les cavaliers sont ouverts occupe les canaux DMX 1 à 8. Une interface sur laquelle le premier et le troisième cavalier sont fermés occupe les canaux DMX à partir de (8+32+1)= 41, soit 41 à 49, etc. Vous remarquerez que seules les six pre-

mières lignes sont utilisées pour déterminer l'adresse DMX. Les deux dernières servent en fait à activer des options du démultiplexeur. À ce jour, seule P1.6 est utilisée pour activer l'option « sortie sur relais ».

La liaison avec le bus DMX est assurée par IC7, un classique émetteur/récepteur (*transceiver*) RS-485, dont la sortie n'est pas utilisée ici. R6 et R7 sont utilisées pour polariser la ligne au repos à l'état 1, ce qui évite un fonctionnement erratique lorsque l'interface est déconnectée du bus. Nous avons préféré sortir la liaison DMX sur picot, pour laisser le libre choix du modèle de prise XLR car on trouve aussi bien des modèles 3 broches que 5 broches. Reportez vous à la fin de l'article pour les détails concernant cette liaison.

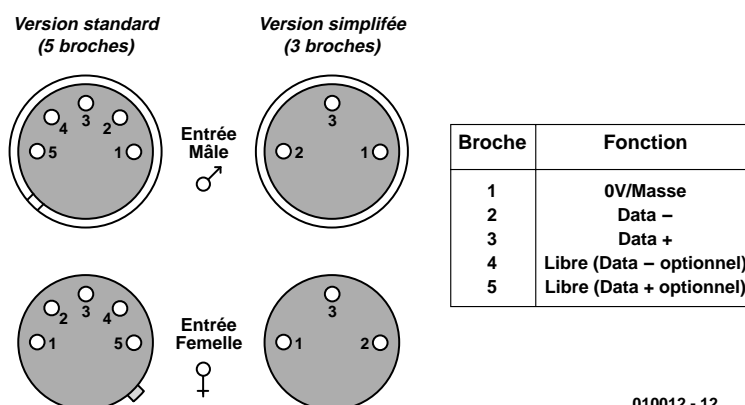
Pour éviter les problèmes posés par les boucles de masse, le LTC490 est isolé galvaniquement du reste du circuit par l'opto-coupleur rapide IC8. Toute cette section du circuit est alimentée à travers un petit convertisseur statique DC/DC, du type NMF0505S (entrée 5 V, sortie 5 V).

Pour générer les 8 tensions de commande, nous avons fait appel à un octuple DAC à sortie tension, fabriqué par Burr-Brown, le DAC8800. Outre le fait qu'il intègre 8 convertisseurs R-2R, le DAC8800 possède aussi le gros avantage d'être commandé par une liaison série synchrone, ce qui simplifie d'autant le circuit imprimé. Sans ce composant, il aurait fallu installer 8 DAC parallèles (de type DAC08 par exemple), avec décodage d'adresse, ce qui impliquait un circuit imprimé notablement plus complexe.

Le fabricant du DAC8800 recommandant que l'alimentation sur la broche 7 dépasse d'au moins 4 V la tension de sortie maximale pour assurer une linéarité maximale, IC4 est donc alimenté sous 15 V. À noter qu'un régulateur interne assure la compatibilité avec les entrées TTL 5 V.

Les entrées de référence VRHA et VRHB permettent de sélectionner la tension sortie maximale des DAC. Elles sont donc reliées à une simple diode zener 10 V, qui sert de référence de tension. Même si cette solution est loin d'être aussi précise qu'une « vraie » source de référence, elle s'avère parfaitement adaptée à nos besoins, et surtout nettement moins coûteuse.

L'impédance de sortie du DAC8800 étant assez élevée, des problèmes potentiels peuvent surgir si les sorties sont directement reliées à certains équipements dont l'impédance d'entrée n'est que de quelques kilohms. Nous avons donc placé entre chaque sortie du DAC et les borniers de sortie des tampons de tension, constitués par des amplis TL084. La tension de saturation des étages finaux des amplis opérationnels empê-



010012 - 12

Figure 4. Brochages (pour mémoire) des connecteurs XLR 5 et 3 broches. Nous avons opté pour une représentation 2D sachant qu'il existe des connecteurs mâles et femelles ainsi que des embases mâles et femelles.

chant la sortie de descendre jusqu'à 0V avec un rail négatif à la masse, les TL084 sont polarisés à -5 V sur leur rail négatif.

Dernier point du schéma, l'alimentation n'appelle aucun commentaire particulier, si ce n'est concernant le transformateur à utiliser. Les entrées AC A et AC B devront être reliées à un secondaire 18V, pour que la tension à l'entrée de IC9 soit suffisante. Pour l'alimentation négative, tout va dépendre du fabricant d'IC11. Si votre circuit est un MC79L05 (Motorola devenu On Semiconductor) ou équivalent : pas de problème. En effet, les circuits MC79Lxx de ce constructeur sont spécifiés jusqu'à 30 V (18 V redressés/filtrés donnent 27 V crête). Avec ce circuit, vous pourrez utiliser le deuxième enroulement secondaire du transformateur connecté à AC A/AC B. Le problème est que certains fabricants de ce type de circuit ne le valide que jusqu'à 15V en entrée. En cas de doute, il faudra utiliser pour cette partie de l'alimentation un (deuxième) transformateur avec un secondaire 9 V.

À vos fers à souder

La réalisation pratique du démultiplexeur ne pose pas, grâce à la platine proposée en **figure 2**, de réels problèmes, vu le nombre assez faible de composants et l'absence de composants critiques.

Même si vous décidez de faire l'économie de supports pour les circuits intégrés, nous ne saurions trop vous

recommander de monter au moins IC7 et IC8 sur support. Leur remplacement éventuel en sera facilité en cas de gros problème sur la liaison DMX (surtension, décharge électrostatique du câble, etc.). Attention au positionnement de IC3, l'EPROM, son orientation étant inversée par rapport au reste des circuits intégrés. Notons que cette EPROM programmée (**EPS010002-21**) est disponible auprès des adresses habituelles.

Etant donnée la faible consommation du montage, IC9 et IC10 pourront se contenter d'un simple morceau d'aluminium comme radiateur.

Concernant K1 et K2, à vous de choisir ce qui convient le mieux à vos besoins (borniers à vis, socles DIN ou XLR...). En ce qui nous concerne, comme le montre les photos, celle en début d'article et celle proposée en **figure 3**, nous avons décidé d'utiliser des sorties sur deux socles DIN, regroupant chacun quatre sorties, mais cette approche n'a rien d'obligatoire.

Une fois tous les composants installés et leur implantation vérifiée, on pourra mettre le montage sous tension. On vérifiera alors la présence des différentes tensions d'alimentation, notamment sur la partie isolée de la carte.

Le test de la carte implique de disposer d'une console d'éclairage DMX ou d'un logiciel sur PC avec l'interface adéquate (par exemple, l'interface MIDI<->DMX d'ELEKTOR [2]...). Après avoir sélectionné

une adresse de départ par S1 (ouvrir S1.7 et S1.8 pour indiquer le mode « tension »), on utilisera le maître DMX pour envoyer les consignes de niveau sur les 8 canaux à partir de cette adresse. En mettant les curseurs de la console au minimum, la tension de chaque sortie doit tomber à 0 V (à quelques millivolts près). En montant les curseurs au maximum, la tension de chaque sortie doit atteindre 10,0 V (ne pas oublier de vérifier l'étalonnage des curseurs de la console, si celui-ci est programmable !).

Si la carte ne semble pas fonctionner correctement, placer un oscilloscope sur la broche 9 d'IC4. On doit y trouver des rafales d'impulsions, signalant que le processeur pilote bien le DAC.

L'absence d'impulsions sur cette ligne d'horloge indique une anomalie au niveau du processeur. Etant donnée la grande vitesse du processeur, il est très fortement recommandé d'utiliser un 74F573 (Série Fast TTL). Sans que nous sachions trop bien pourquoi, l'un de nos prototypes s'est montré capricieux avec cette famille de circuits. En le remplaçant par un 74HCT573, le problème a été résolu.

Vérifiez également sur la broche ALE la présence d'impulsions à 2 MHz. Si vous trouvez une autre fréquence d'impulsions, c'est que le quartz n'oscille pas ou pas à la bonne fréquence. Si vous avez monté L1, faites un essai après l'avoir des-soudée. Si le circuit fonctionne bien sans cette self, c'est que votre quartz oscille à 24 MHz en fondamentale. La self est alors inutile.

Enfin, vérifiez, si vous décidez de programmer vous-même votre EPROM, IC3, qu'elle est bien du type 120 ou 150 ns (suffixe -12 ou -15). Les modèles plus lents ne conviennent pas à cette application. Nous mettons à votre disposition, par le biais des canaux habituels, une disquette baptisée **EPS010002-11** comportant les fichiers source et hexadécimal de cette réalisation.

Si la section processeur fonctionne bien, mais que vous n'avez pas d'action sur les sorties analogiques, vérifiez l'interface DMX. Commencez par débrancher tout câble relié à RD+/RD- et TD+/TD-. La broche 2 d'IC7 doit être à l'état haut, ainsi que

la broche 6 d'IC8. N'oubliez pas qu'IC7 est isolé du montage et que la sonde négative du voltmètre doit être sur la masse **ISOLEE**! Prendre une alimentation stabilisée réglée à environ 5 V, et connecter le - à RD+ et le + à RD-. La sortie d'IC7 doit maintenant passer à l'état bas, ainsi que la sortie d'IC8. Si ce test échoue, il faut vérifier le circuit autour d'IC7, IC8 et IC12.

Si les différents tests sont positifs mais que l'interface refuse de fonctionner avec une console DMX, il faut au préalable vérifier qu'il n'y a pas d'inversion entre les fils RD+ et RD-. Certaines marques, comme MARTIN, ont produit des équipements dont les points chaud et froid sont inversés par rapport à la norme. D'autre part, il ne faut pas oublier les fameuses terminaisons RS-485, qui sont impératives en bout de câble. Ces terminaisons sont formées simplement d'une résistance d'environ 100 Ω prise entre RD+ et RD- (ou TD+ et TD-). Vérifier également l'absence de court-circuit entre l'un des conducteurs et le blindage (SHLD).

Une fois que tout fonctionne correctement, on pourra passer à la phase finale, redoutée de nombre d'électroniciens : la mise en boîte. Ce genre d'interface est destiné le plus souvent à être placé près des gradateurs, ou des luminaires avec une entrée télécommande. En ce qui nous concerne, nous avons pris l'habitude de placer nos démultiplexeurs dans des boîtiers plastiques étanches, du genre « boîte de dérivation » (la version présentée en photo est une exception). Pour limiter les problèmes posés par les interférences, vous pouvez recouvrir l'intérieur du boîtier de graphite par pulvérisation (produit vendu pour les tubes cathodiques). Si vous prenez un boîtier métallique, reliez-le à la terre. Nous prévoyons en général sur un côté du boîtier des boulons permettant de placer le cas échéant des pattes de fixation standardisées, pour le montage sur des supports luminaires. La LED D3 est placée sur le dessus du boîtier pour permettre de vérifier depuis le « plancher des vaches » que les interfaces sont correctement alimentées.

Dernier point à évoquer : la connectique DMX. Le standard DMX prévoit l'utilisation de connecteurs XLR

5 broches, mâle du côté RD et femelle du côté TD, en vue de réaliser une « daisy-chain » (équipements DMX reliés à la queue leu leu). Néanmoins, certains constructeurs préfèrent utiliser des XLR 3 broches, moins coûteuses. À vous de voir ce que vous préférez. Nous nous permettons de rajouter le brochage des connecteurs XLR 3 et 5 broches en figure 4.

Bon sang !

Où est l'entrée télécommande ?

Ca y est ! Votre interface fonctionne ! Et maintenant, il faut la connecter à vos équipements lumière (ou autres).

Si les équipements que vous souhaitez piloter disposent d'une entrée de télécommande 0 à +10 V (c'est le cas de la très grande majorité des gradateurs, stroboscopes et fumigènes professionnels, voire certains luminaires simples), aucun problème : la liaison est directe. Le cas échéant, il faudra prévoir une adaptation du niveau des tensions par un pont diviseur, si l'entrée de l'équipement est du genre 0 à 5 V ou toute autre valeur bâtarde. Dans le cas (rare) où les équipements sont commandés en courant, une résistance en série réglera le problème.

Au niveau grand-public, il existe assez peu d'équipements avec une entrée télécommande en tension. Le plus souvent, on ne trouve que des entrées de validation. On se rabattra alors sur la version à relais qui sera proposée dans un prochain article.

Si les équipements que vous souhaitez commander ne sont munis d'aucune entrée de télécommande ou si les entrées de télécommande mises à votre disposition ne sont pas compatibles avec les standards supportés par notre démultiplexeur, il vous faudra jouer du fer à souder dans l'appareil, en vue d'y apporter les modifications nécessaires.

Par manque de place dans cet article, nous ne pouvons évoquer ici ce genre de modifications. Si les demandes des lecteurs sont suffisantes, nous pourrions consacrer un article complet à ce genre de modifications.

À très bientôt sur les ondes DMX...

Dans un prochain article nous vous proposons une extension à relais qui pourra se connecter au présent montage. Cette extension à relais se substituera au convertisseur DAC8800 et se connectera à la présente platine par le biais d'un câble en nappe.

(010002)

Bibliographie :

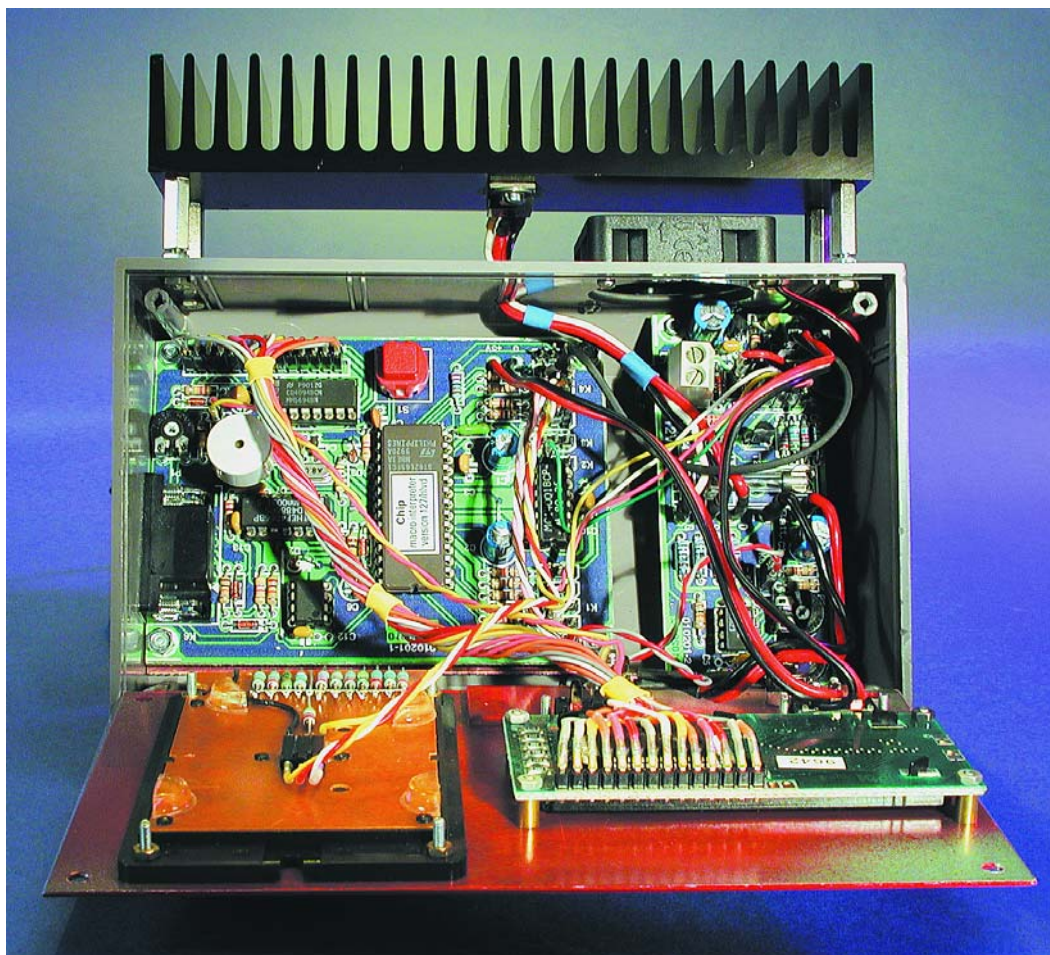
- [1] Les mystères du DMX512, Elektor n°275, mai 2001, page 52 et suivantes
- [2] Interface DMX <-> MIDI, Elektor n°280, octobre 2001, page 32 et suivantes

Vide-accumulateur capacimètre

2^e partie : clavier, déchargeur et assemblage

Bob Stuurman

Nous en avons déjà parlé dans la première partie, pour utiliser l'unité de commande en tant que déchargeur, nous aurons encore besoin de quelques composants. Il nous faut, bien sûr, un clavier pour introduire les paramètres et une platine pour le déchargeur proprement dit. Et aspect fondamental du projet, de petits logiciels pour vérifier le matériel.



Le clavier

Pour introduire les commandes, rien de tel qu'un clavier. Nous sommes partis d'un pavé, prêt à l'emploi, de 12 touches, à branchements individuels ou groupés. Comme l'unité de commande peut mesurer des tensions analogiques, un diviseur de tension pour chaque touche sera suffisant. Le tableau situé sous le schéma de la **figure 5** reprend, pour chaque touche du clavier, les symboles gravés, les valeurs de résistance à la deuxième rangée et en dessous, les valeurs hexadécimales obtenues. Les résistances peuvent se placer directement sur la face arrière du clavier. Le point commun va à la masse (Gnd) et le +5 V à la résistance de 24,7 k Ω , partie commune du diviseur de tension, reliée aux résistances des touches. On connecte le clavier à

Input 0 du circuit de commande par un cordon à 3 fils, long de 20 cm et terminé par une embase femelle à trois pôles.

Le programme « *KEYTEST.HEX* », présent sur la disquette qui accompagne ce projet, nous permettra de vérifier le bon fonctionnement du clavier. Une fois le programme chargé, on peut le lancer en tapant « Chip ». Tant qu'aucune touche n'est pressée, l'afficheur indique « FF_{HEX} » ou « FE_{HEX} ». Lorsqu'on appuie sur une touche, c'est la valeur hexadécimale, celle reprise dans le tableau, qui apparaît. En fait, la valeur peut différer, puisque seule la moitié haute est utilisée, la partie basse peut se situer entre 5_{HEX} et A_{HEX}.

Déchargeur

Comme le montre la **figure 6**, le déchargeur se compose de trois parties. Celle du haut comprend l'alimentation et le commutateur du ventilateur. Une alimentation stabilisée à brancher sur le secteur (6 à 12 V / 300 mA) suffira. Pour le stabilisateur (IC1), on choisira un modèle à faible chute directe, qui délivre encore précisément 5 V quand on ne lui fournit que 6 V. En utilisant une alimentation à tension variable, on s'offre le loisir de régler à son gré la vitesse du ventilateur de 12 V. La basse tension se branche à K3, le ventilateur à K2 et sur K1, le connecteur d'alimentation de la platine de commande.

On peut mesurer la tension de l'accumulateur sur K6 (+a V et -a V). Un petit cordon les relie directement au support d'accumulateurs, de manière à éviter que la chute de tension sur le fusible et le câblage de puissance ne fausse les mesures. La cellule R4 et C3 filtre le signal de K6, qui traverse ensuite un atténuateur réglable à trois calibres : 10,2 V, 20,4 V et 40,8 V. La valeur à fond d'échelle de chacun d'eux correspond à FF_{HEX} (255_D). Ce sont les niveaux de Out 0 et Out 1 qui déterminent la position de l'atténuateur et les rapports de réduction ont été choisis de manière à éliminer tout calcul superflu. Le programme « *DIVITEST.HEX* » sur la disquette servira aux tests.

Le reste du montage assure le réglage et le maintien du courant de

décharge. La grille du MOSFET de puissance (T4) reçoit un signal PWM de la sortie Servo1, filtré par R9 et

C4. les résistances R13 à R16 dans le circuit de source de T4 sont les résistances de mesure du courant. La chute de tension à

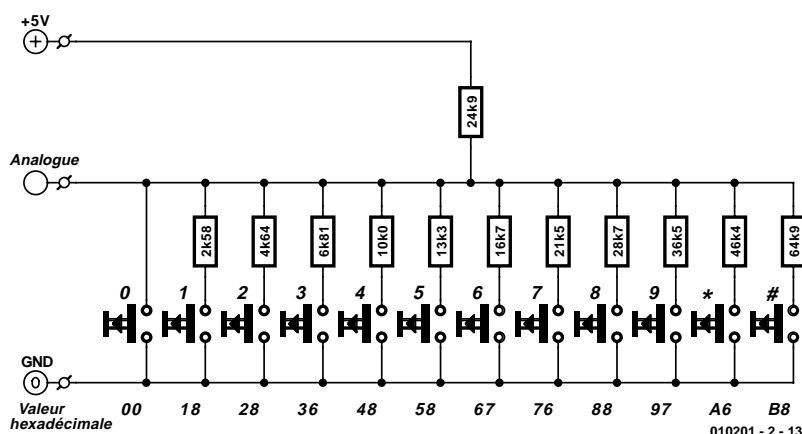


Figure 5. Le clavier et les indispensables diviseurs de tension.

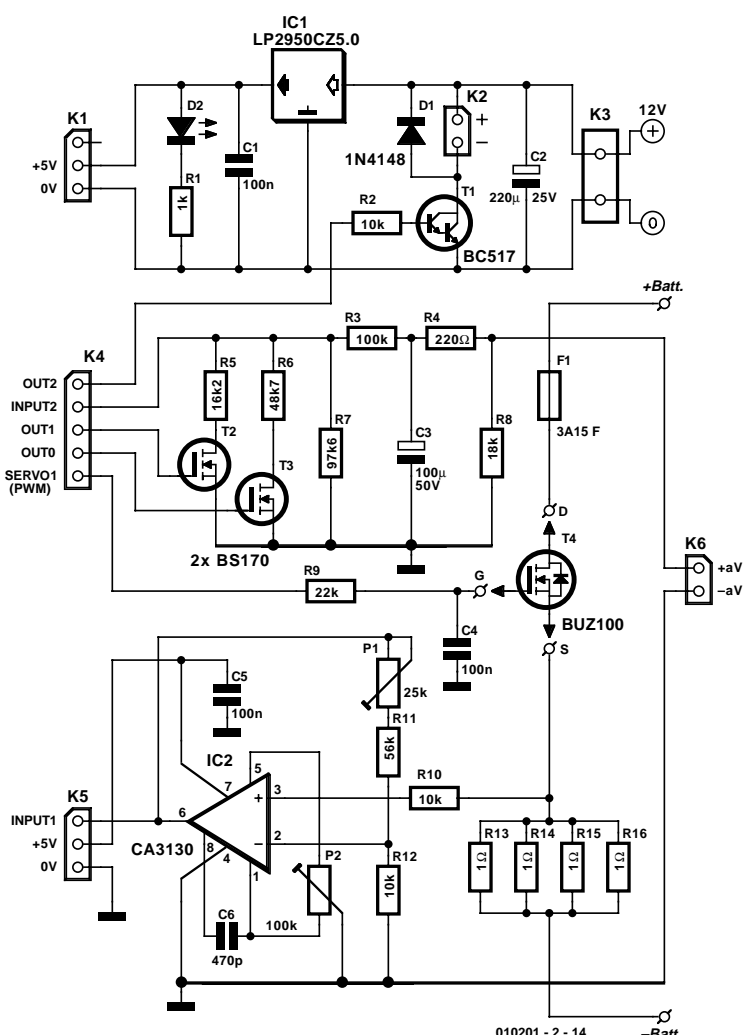


Figure 6. Schéma du déchargeur proprement dit. Le MOSFET de puissance T4 se tape le gros du travail.

Liste des composants de la platine du déchargeur

Résistances :

R1 = 1 k Ω
 R2, R10, R12 = 10 k Ω
 R3 = 100 k Ω
 R4 = 220 Ω
 R5 = 16 k Ω
 R6 = 48 k Ω
 R7 = 97 k Ω
 R8 = 18 k Ω
 R9 = 22 k Ω
 R11 = 56 k Ω
 R13 à R16 = 1 Ω /0W5
 P1 = ajustable 25 k Ω
 P2 = ajustable 100 k Ω

Condensateurs :

C1, C4, C5 = 100 nF céramique
 C2 = 220 μ F/25 V radial
 C3 = 100 μ F/50 V radial
 C6 = 470 pF

Semi-conducteurs :

D1 = 1N4148
 D2 = LED rouge à haut rendement
 T1 = BC517
 T2, T3 = BS170 (ou BSN10A)
 T4 = BUZ100
 IC1 = LP2950CZ5.0
 IC2 = CA3130

Divers :

K1, K5 = embase mâle à 1 rangée de 3 contacts
 K2, K6 = embase mâle à 1 rangée de 2 contacts
 K3 = bornier encartable à 2 contacts au pas de 5 mm
 K4 = embase mâle à 1 rangée de 5 contacts
 F1 = fusible 3,15 A (rapide) + porte-fusible

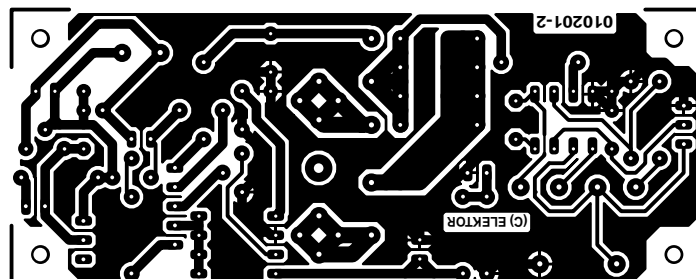
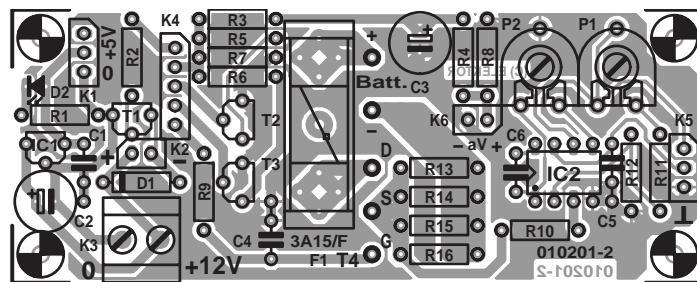


Figure 7. Dessin des pistes et disposition des composants sur la platine.

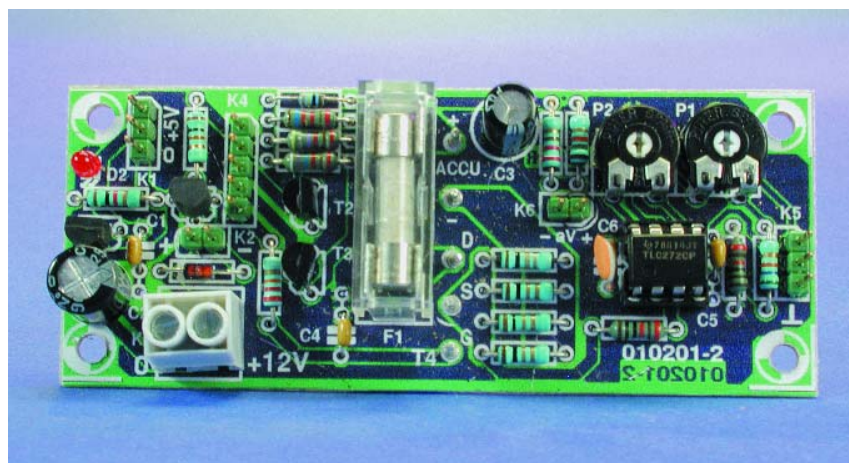


Figure 8. Le circuit complet, une fois assemblé.

leurs bornes est appliquée à l'amplificateur opérationnel IC2, lequel l'adapte et, par K5, l'envoie à Input 1 de la platine de commande. P1 règle le gain et P2 compense la dérive d'entrée. Pour le réglage du rapport cyclique du signal PWM, nous utilisons une variable qui nous indiquera si le courant reste stable. Si la tension descend sous 1 V, il est possible que le MOSFET ne soit plus capable de fournir le courant nécessaire. Dans ce cas, la variable passe par zéro et l'événement sera signalé.

En parallèle, mais de polarité inversée, sur drain et source du MOSFET de puissance T4, nous avons placé une diode. Si d'aventure quelqu'un relie un accumulateur à l'envers, le fusible F1 limitera la casse. Le tout, c'est

d'employer un fusible rapide. Le dessin du circuit imprimé ainsi que la disposition des composants se trouvent à la **figure 7**. Aucune indication particulière n'est nécessaire. Tous les connecteurs sont des embases mâles, à l'exception de K3 (bornes pour platine) et des connecteurs d'alimentation pour lesquels on a utilisé des clous pour circuit imprimé. On montera les résistances R13 à R16 à 4 mm du circuit pour mieux évacuer la chaleur. Sur la maquette, nous avons ajouté des tresses à souder aux pistes qui mènent aux accumulateurs pour en diminuer la résistance. Vous pouvez

vous fier à la **figure 8** pour comparer votre montage à notre résultat.

Assemblage

Notre prototype, nous l'avons facilement introduit dans un boîtier Teko de type 362, mais naturellement d'autres modèles conviendraient aussi. Voyez, sur la photo du titre, comment nous avons intégré le tout au boîtier.

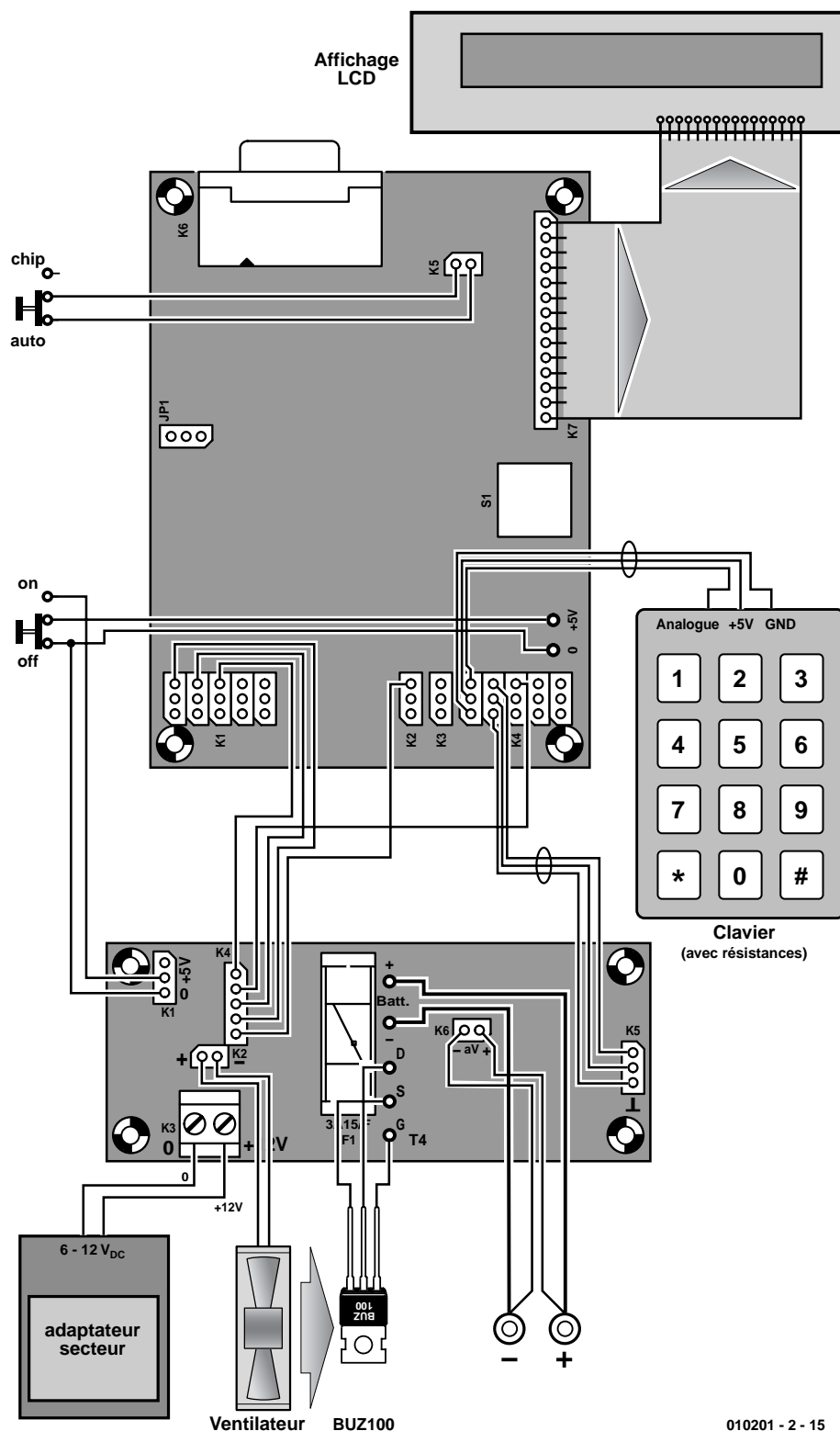
La face avant est percée de huit trous de 14 mm de diamètre pour laisser entrer l'air, que le ventilateur chasse par une large ouverture dans la face arrière. On fixera les deux pla-

tines au plancher du boîtier à l'aide d'entretoises. Pour le clavier et l'écran à cristaux liquides, quelques coups de scie dans la face avant devraient faire l'affaire. N'oubliez pas de ménager des ouvertures pour les deux interrupteurs à glissière et les bornes de raccordement pour fiche banane. Sur notre prototype, le connecteur RS-232 a été fixé au côté gauche du boîtier.

Il ne reste plus que quelques connexions à effectuer, le schéma de la **figure 9** lèvera tout doute à ce propos. Le cordon d'alimentation arrive sur K3 (attention à la polarité !) Il faut encore souder trois fils au MOSFET de puissance, nous avons pris un fil plus épais pour le drain et la source, et à l'autre extrémité, trois cosses femelles à glisser sur les broches de raccordement de T4 sur le circuit imprimé. Nous avons choisi également du gros fil pour relier à la platine les bornes des accumulateurs et les connecteurs de la face avant. Pour toutes les autres connexions, nous avons utilisé des rangées de connecteurs femelles encartables, sciées à la longueur voulue. À l'autre extrémité du cordon pour K6, nous avons fixé deux cosses à souder que nous avons ensuite vissées au bus de raccordement. Le ventilateur se branche sur K2 et la fiche du cordon d'alimentation de la platine de commande sur K1. Pour relier K5 à

Input 1 de la platine de commande, nous avons pris un cordon à trois fils terminé de part et d'autre par une embase femelle à trois pôles. Ce câble véhicule également le +5 V et

le 0 V. Ce qui correspond à K4, c'est une embase femelle à 5 contacts, reliée à cinq fils séparés, chacun porteur, à l'autre extrémité, d'une embase femelle unipolaire. Ils se brancheront à Out 2, Input 2, Out 1, Out 0 et PWM



010201 - 2 - 15

Liste des composants pour l'assemblage

- boîtier tel que, par exemple, Teko type 362
- radiateur pour T4 : Fischer SK-132/37,5/SA par exemple
- ventilateur 40 x 40 mm, 12 Vdc
- adaptateur secteur 300 mA, 6 à 12 Vdc, non régulé
- pince crocodile rouge telle que, par exemple, Hirschmann PKI-100
- pince crocodile noire idem
- clavier 12 touches à connexion commune et 12 lignes individuelles tel que 19-55-61 (Conrad)
- interrupteur à glissière miniature bipolaire
- barrette de contacts femelles au pas de 2,54 mm

Figure 9. Schéma de câblage du déchargeur d'accumulateur. Beau spaghetti, non ?

(servo 1), dans l'ordre. On arrive ainsi à éliminer les boucles de masse.

Il ne reste plus qu'à relier K5 à l'interrupteur « Chip-Auto » du tableau de bord de la platine de commande.

Logiciel

Ce déchargeur d'accumulateurs vous propose deux modes de fonctionnement : ou vous activez le processeur de commandes ou vous lui faites exécuter un programme. Pour communiquer avec l'appareil ou charger le logiciel, nous devons disposer, dans le PC, d'un logiciel « *CHIPTERM* » ou « *VBTERM* » (la version pour Windows) pour assurer le transit de données via le port sériel. Le commutateur « Chip-Auto » doit être dans la position Chip et le déchargeur allumé.

La combinaison <Alt+l> permet alors le chargement d'un programme. Dans le bas de l'écran une invite vous demande le nom du programme choisi. Ils se terminent tous par l'extension « .HEX ». Confirmez par Entrée et le programme se charge. Le logiciel « *KEY-TEST* » permet d'inspecter le clavier. On peut le lancer par la commande « Chip » ou glisser le commutateur « Chip-Auto » en position Auto et mettre le déchargeur en marche.

Nous devons aussi pouvoir contrôler le diviseur de tension. Pour ce faire, nous disposons du programme « *DIVITEST* ». Appliquons une tension continue de 5 V aux bornes de raccordement. L'afficheur devrait indiquer 125 (5 / 10,2 x 255) et une atténuation de 2. Si l'on applique 15 Vcc, la valeur doit passer à 188 et l'atténuation à 4. Finalement, pour 25 Vcc, on obtiendra une valeur de 156 et 8 de facteur d'atténuation. Les grandeurs mesurées peuvent s'écarter de 2 à 3 unités. Au-delà de cette fourchette, partant de diviseurs de tension corrects, on peut en déduire la présence d'un courant de fuite de T2 ou de T3.

Une fois ces examens réussis, nous pouvons charger le programme « *NICADIS* ». L'appareil est prêt à fonctionner, reste le réglage du courant. Raccordez au déchargeur un bloc d'accumulateurs, en série avec un ampèremètre. Indiquez le nombre de cellules qui le composent, la tension par élément et programmez le courant à 2 500 mA. À l'aide de P1, réglez le courant à 2 500 mA. Choisissez ensuite un courant de 100 mA et ajustez P2 pour obtenir cette valeur. L'affichage fluctue quelque peu, à cause de la boucle d'asservissement, choisissez donc un juste milieu.

Élaboration de la courbe de décharge

À chaque minute, pendant la décharge, l'appareil mémorise un échantillon de tension

instantanée. Les adresses 600_{HEX} à 7FF_{HEX} de l'EEPROM sont réservées à cet usage. L'adresse 600_{HEX} contient le facteur d'atténuation et l'adresse 601_{HEX}, la grandeur du courant. À la fin du processus, ces échantillons pourront être sauvegardés dans un fichier. Dans le processeur de commandes, on ouvre un fichier .LOG par la combinaison <Alt+o>. La commande « prog 600 » affiche l'adresse 600_{HEX}, suivie de l'octet qui s'y trouve et le suivant. La touche « + » permet d'afficher les adresses suivantes. Vous pouvez continuer jusqu'à ce que vous rencontriez une donnée égale à

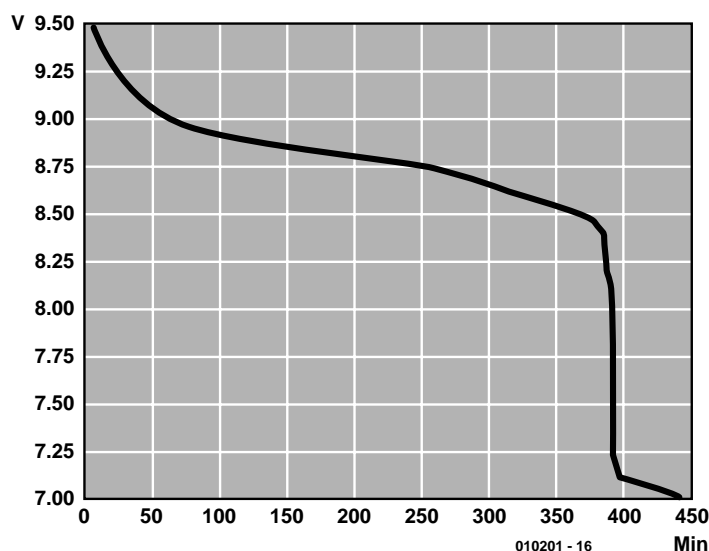
00_H, elle marque la fin des échantillons. Pour fermer le fichier .LOG utilisez la combinaison <Alt+c>. Le logiciel « *LOGDATEX* » convertit les données échantillonnées en valeurs de tension et les numérote. En outre, il les interpole pour que la courbe apparaisse plus fluide, calcule la tension moyenne et l'énergie dispensée durant la décharge. Le logiciel traceur est alors capable de relire le fichier .DAT que nous avons créé, puis de dessiner la courbe de décharge. Le fichier *PLOTR.ZIP* contient toutes les informations nécessaires.

(010201-II)

Analyse d'une courbe de décharge

La courbe de la **figure 10** illustre la décharge d'une batterie de sept accumulateurs CdNi, d'une capacité de 1 900 mAh. Nous avons fixé le courant à 190 mA (0,1 C) et la tension finale à 1 V par élément. Après 8 heures et 4 minutes, nos accumulateurs étaient déchargés et dévoilaient une capacité de 1 532 mAh. Un peu faiblarde, direz-vous, mais notre batterie n'était plus toute jeune non plus. La courbe se présente selon les prévisions, mais il y a quelque chose d'étrange. Lorsque la batterie atteint 8,4 V, elle est à plat et sa tension chute brutalement. Un comportement classique... Par contre, à 7,1 V, la courbe se redresse nettement. Comme la chute de tension en ce point vaut 1,3 V, il est manifeste qu'un des éléments s'est polarisé en inverse et qu'il a commencé alors à se charger à l'envers. Peu après, la tension finale était atteinte. Pour un courant de décharge correspondant à 0,1 C, il aurait été plus judicieux de fixer la tension finale à 1,1 V par élément. Nous aurions probablement évité cette inversion.

Les accumulateurs au plomb sont encore plus sensibles au renversement de tension que les CdNi ou les NiMH. En pratique on choisira une tension finale de 75% à 80% de la tension nominale. Bien entendu, le courant de décharge interviendra également dans cette décision.



Courbe de décharge d'une batterie de 7 accumulateurs au CdNi.

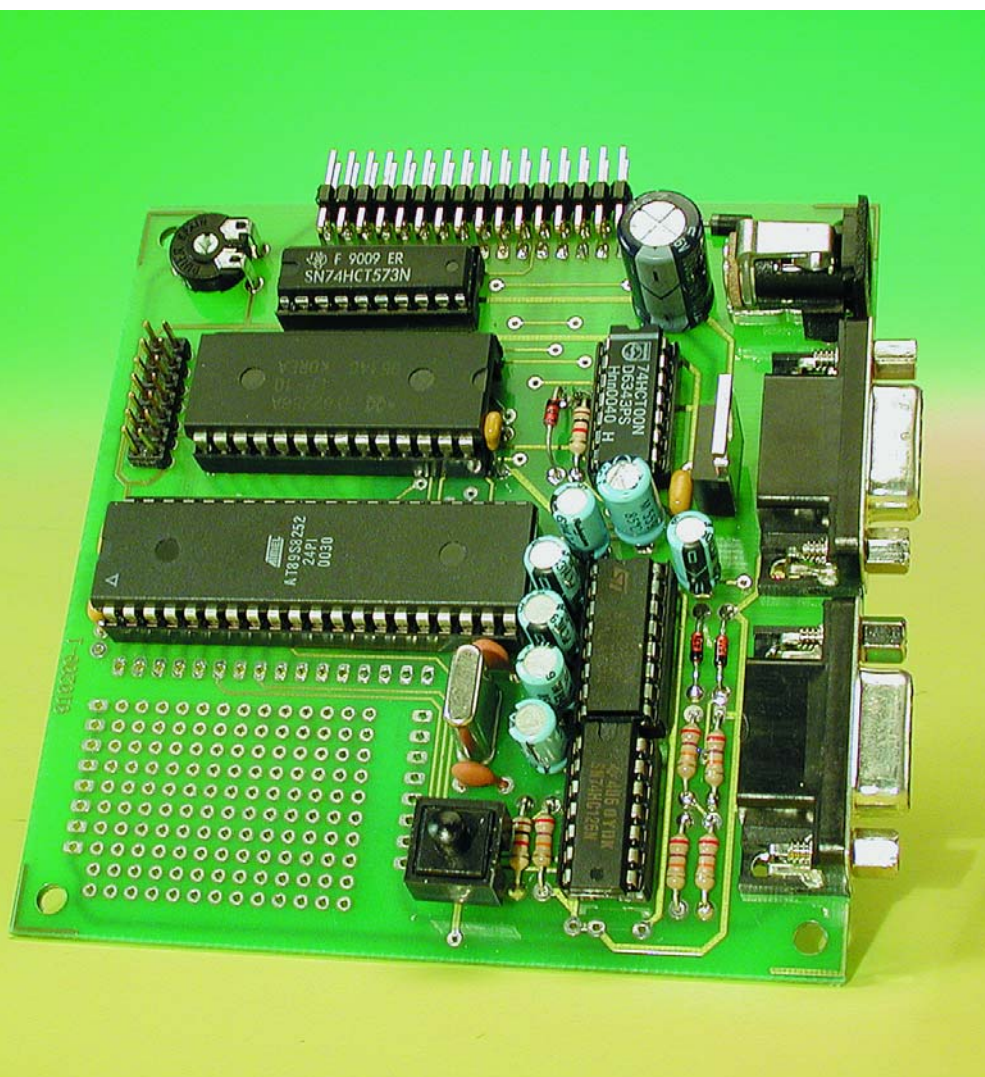
Matériel et logiciel didactiques

Carte 89S8252 Flash

Base d'un cours sur les microcontrôleurs

par B. Kainka

La carte microcontrôleur sur laquelle est basée un cours de programmation doit être d'un emploi simple tout en offrant les avantages d'un microcontrôleur moderne avec périphérie. Cela la rend aussi intéressante pour les vieux routiers des contrôleurs !



Qui veut convertir ses idées en circuits y parvient la plupart du temps plus vite avec un microcontrôleur qu'avec l'électronique conventionnelle. Mais la programmation de nombreux systèmes est relativement complexe. C'est pourquoi Elektor a mis au point une nouvelle carte Flash. La mémoire de programme Flash et une conception élaborée du système garantissent une programmation simple et rapide.

Les réflexions préliminaires au sujet d'une nouvelle carte microcontrôleur sont parties des buts suivants : Il faudrait développer un circuit qui s'adresse à l'utilisateur expérimenté autant qu'au débutant. Le système devait « comprendre » d'emblée les divers langages de programmation, de l'assembleur à Basic en passant par C. Il devrait aussi être facilement extensible et simple à programmer. En outre, un programme transmis à la carte devait être conservé après la mise hors tension. On s'évite ainsi la tâche pénible de graver une EPROM. La carte elle-même devait donc servir également de programmeur.

Il existe de nombreux microcontrôleurs à EEPROM internes ou à mémoire Flash qui offrent en principe des capacités de programmation aisée. Devait-on choisir un PIC, un contrôleur de la famille 68, un

AVR ou un 8051 étendu ? La question difficile quant au type de contrôleur a été tranchée en faveur de la famille 8051. Il y a de bonnes raisons pour cela : la plupart des autres contrôleurs ne sont fabriqués que par une seule entreprise, ce qui pourrait conduire à des problèmes de livraison dans des cas extrêmes. La famille 8051, au contraire, peut être désignée à juste titre comme norme industrielle, car de nombreuses entreprises produisent des dérivés du standard originellement développé par Intel. Si donc un appareil a été développé et si la société A est subitement incapable de livrer, on peut passer facilement à un contrôleur de la société B ou de la société C. Il existe en outre de nombreux logiciels pour la famille 8051 et une littérature abondante à son sujet. Qui mise sur la norme 8051 ne peut pas perdre. Cela est encore valable aujourd'hui bien que de nouveaux

concepts ait entre temps vu le jour. L'AT89S8252 d'Atmel utilisé (dont on retrouve la structure interne en **figure 1**) correspond essentiellement au 8052 d'Intel. Cela signifie par exemple que le BASIC-52 bien connu peut être utilisé sans problèmes avec cette puce. Certaines caractéristiques supplémentaires feront plaisir au réalisateur. Il y a par exemple une surveillance du type « chien de garde » (*watchdog*) interne et une EEPROM de données de 2 Koctets (K). La mémoire de programme de 8 K en technologie Flash constitue toutefois l'argument décisif. Le programmeur de cette mémoire se trouve aussi sur la puce. On n'a même pas besoin d'une tension de programmation particulière : la tension normale de 5 V suffit. La programmation est effectuée en série et de façon synchrone par une interface SPI. Seules quelques liaisons sont nécessaires pour charger

Microcontrôleur 89S8252

- 32 K RAM
- 8 k de mémoire Flash
- 2 k d'EEPROM de données
- Temporisateur de surveillance (Watchdog-Timer) interne
- Interface série
- Interface de programmation
- Tension de fonctionnement de 9 à 12 V, régulateur de tension 5 V
- Interface LCD
- Raccordement du bus par connecteurs à picots
- Raccordement des ports par connecteurs à picots
- Zone d'essais pour extensions

un programme dans la zone interne de mémoire Flash.

La carte

La carte Flash étend la mémoire du contrôleur de 32 K de RAM, comporte une interface LCD, une interface de série et une interface de programmation. Elle offre aussi une zone de prototypage comportant 130 îlots de soudage. Il est donc facile d'ajouter de petites extensions, d'une commande de moteur à des interfaces supplémentaires en passant par des capteurs. Le bus de données et d'adressage ainsi que tous les ports sont accessibles par des barrettes de connexion. Le système peut donc être étendu presque sans limites. La carte possède 2 embases DB9. La communication avec l'interface série s'effectue par l'une d'entre elles. Comme chaque microcontrôleur de la famille 8051, le 89S8252 possède aussi un UART interne pouvant être par exemple connecté à une interface COM du PC par un pilote d'interface MAX232. L'utilisateur et le logiciel sont les seuls à décider si le PC transmet des commandes au contrôleur, lit des résultats de mesures ou communique avec l'interpréteur BASIC interne.

La seconde embase DB9 (COM2) ne sert qu'à la programmation du contrôleur. En raccordant un PC, celui-ci peut réinitialiser le micro-contrôleur, le commuter en mode de programmation, transférer et lancer un programme. Toute la carte peut donc être pratiquement télécommandée. Elle pourrait donc être programmée sans difficulté même si elle se trouvait en un lieu inaccessible.

La première embase DB9 COM1 est reliée à l'une des interfaces sérieelles du PC. L'utilisateur écrit et compile alors le programme, le transmet au contrôleur et le teste sur la carte.

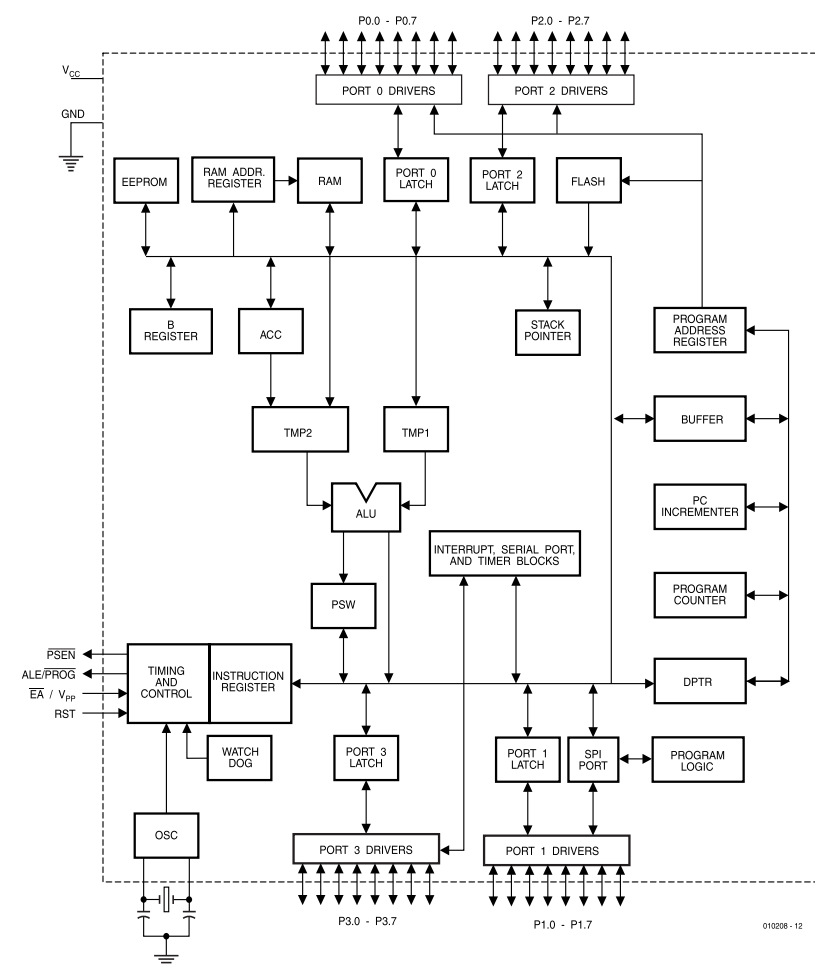
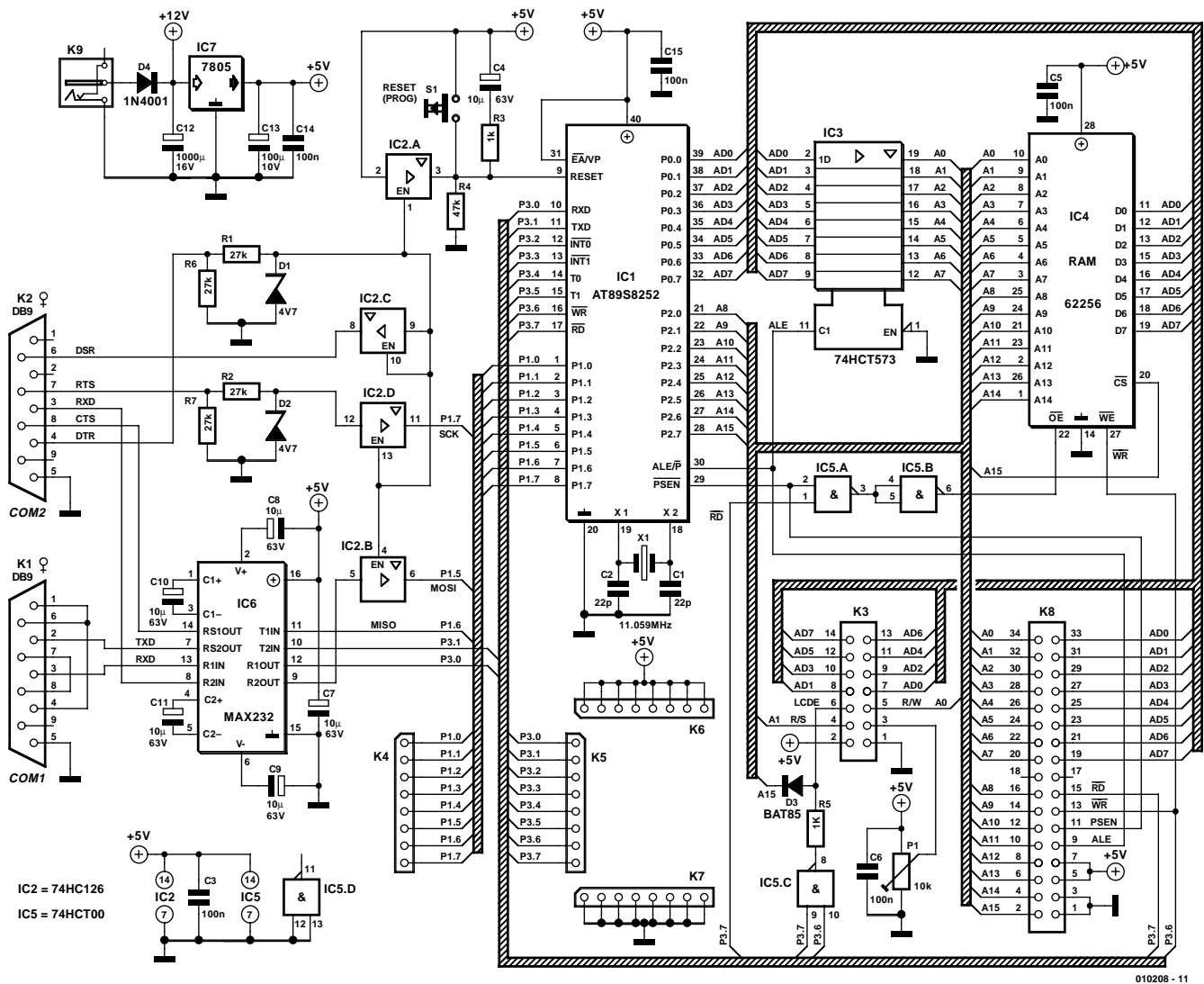


Figure 1. Schéma du contrôleur AT89S8252.



010208 - 11

Figure 2. Un système microcontrôleur classique et bien conçu.

Ce n'est que si le programme transmis a besoin de communiquer avec le PC qu'il faut aussi utiliser K1. Il existe 2 possibilités. Soit permuter le câble de liaison, soit utiliser 2 interfaces COM du PC. C'est possible sur de nombreux PC dont les 2 interfaces COM sont disponibles, la souris occupant une connexion PS2 ou fonctionnant sur USB. L'utilisation en parallèle de 2 interfaces COM se révèle souvent particulièrement pratique et conviviale. La programmation et les tests vont de paire si les outils nécessaires sont utilisés parallèlement.

Le circuit de la **figure 2** est basé en grande partie sur la solution habituellement utilisée avec les systèmes 8051. Un verrouillage d'adressage 74HCT573 (IC3) fournit les 8 adresses inférieures pour les 32 K de RAM (IC4). Comme la ligne CS de la RAM est raccordée à la ligne d'adressage A15, la mémoire ne couvre que la zone de 0 à 7 FFF_{HEX} ; la

zone supérieure est disponible pour des extensions à partir de 8000_{HEX}. La ligne WR de commande d'accès d'écriture est excitée directement par le processeur. La ligne CE des accès lecture reçoit par contre un signal engendré à l'aide des deux portes du 74HCT00 (IC5) à partir de la fonction logique ET appliquée au signal PSEN pour les accès au programme et au signal WR pour les accès aux données. Les contrôleurs 8051 utilisent normalement des domaines d'adressage parallèles pour les programmes et les données. Cette séparation ne s'applique pas ici. Le contrôleur utilise, certes, le plus souvent un programme dans la ROM Flash interne de 8 K et ne place que des données dans la RAM. Il demeure toutefois possible d'utiliser

des parties de programme dans les zones de la RAM situées au-dessus de 2000_{HEX}. Puisque ces zones peuvent aussi contenir des données, il est possible de rendre les programmes auto-modifiables. Le processeur écrit son propre code dans la RAM puis l'exécute.

L'EEPROM interne du processeur est par ailleurs une pure mémoire de données parallèle à la RAM externe. L'EEPROM doit être activée par le logiciel. On peut alors y placer des données de façon permanente. Les 2 K inférieurs de la RAM externe sont inactifs dans ce cas.

L'afficheur LC est raccordé lui aussi au bus de données et occupe le domaine supérieur à 8000_{HEX}. Une fonction logique NON-ET (NAND) de RD et WR fournit le signal d'acti-

tion (*Enable*) de l'afficheur. Une simple logique basée sur de diodes bloque ce signal dans le domaine d'adressage s'étendant jusqu'à $7FFF_{\text{HEX}}$. L'afficheur occupe alors les adresses suivantes :

8000_{HEX} : écrire une commande

8001_{HEX} : lire une commande

8002_{HEX} : écrire des données

8003_{HEX} : lire des données

Ce domaine se répète jusqu'à $FFFF_{\text{HEX}}$. On peut donc utiliser aussi bien le domaine de $F000_{\text{HEX}}$ à $F003_{\text{HEX}}$. L'interface série avec les lignes

TXD et RXD est raccordée comme d'habitude au port du PC par un circuit intégré d'interfaçage du type MAX232. L'acquittement n'est pas utilisé. Les liaisons internes des lignes d'acquittement permettent de tenir compte des besoins de certains programmes d'émulation de terminal qui attendent un signal d'acquittement XON.

La construction de la carte Flash est enfantine et devait être terminée en 1 heure. On peut utiliser de (bons) supports pour tous les circuits intégrés, mais il faut alors veiller à la fin à les placer dans le bon sens (comme les diodes et les condensateurs électrolytique). Pour K8, on se servira d'un connecteur à picots de 34 broches à 2 rangées (ergot de détrompage vers l'extérieur). Les broches du milieu ne sont pas utilisées et doivent être sectionnées pour que les 4 connecteurs à 1 rangée de 8 broches placés aux extrémités des câbles puissent se trouver côte à côte.

Programmation

L'interface de programmation du contrôleur comporte une interface SPI avec les lignes suivantes :

SCK (P1.7) : ligne d'horloge

MISO (P1.6) : lire des données

MOSI (P1.5) : écrire des données

RST : ligne de réinitialisation

La programmation se déroule lorsque la ligne de réinitialisation est active. Un outil de pro-

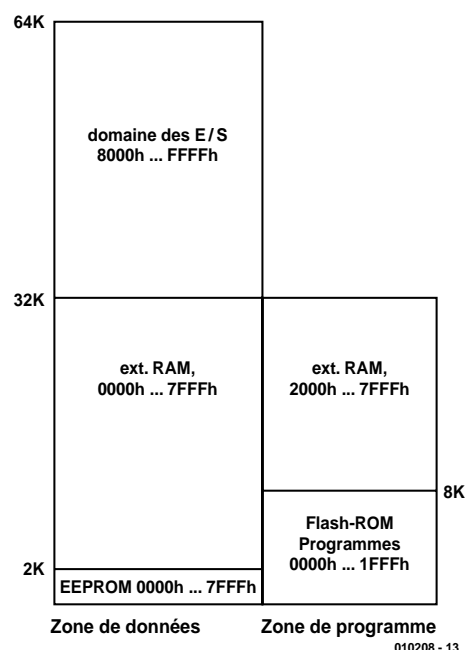
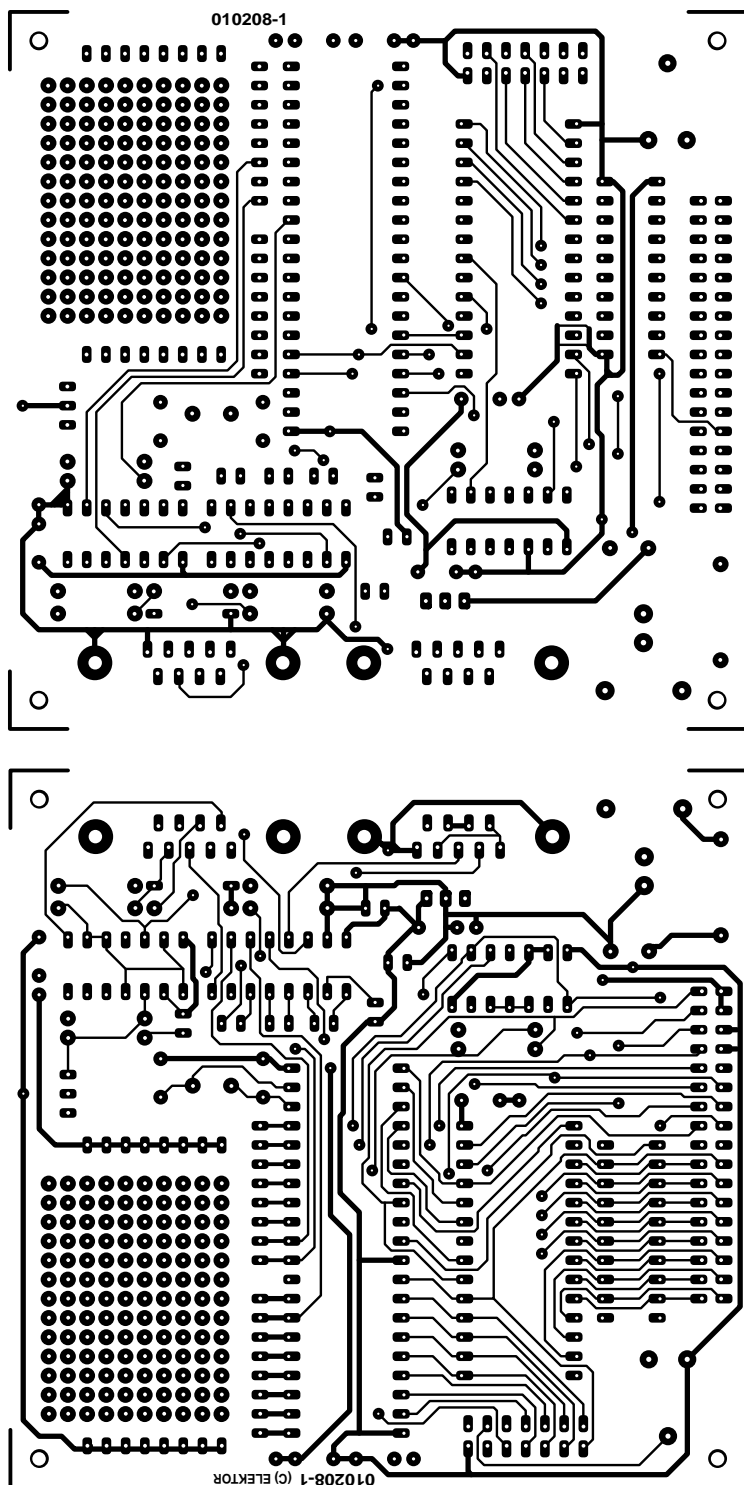


Figure 3a. Disposition de la carte et ...

Figure 4. Mémoire interne et externe de la carte microcontrôleur.

grammation engendre donc un état de réinitialisation et écrit ensuite les données par MOSI, chaque bit étant transmis dans un registre à décalage interne du processeur à la fréquence de SCK. La ligne MISO permet de surveiller de la même manière le succès de la programmation ou de lire un programme. Tous les signaux sont produits directement par les lignes RS-232 qui fournissent ± 10 V. Il est donc nécessaire de les adapter au niveau 5 V du processeur. La moitié encore libre du MAX232 se charge de l'adaptation du niveau des lignes de données MOSI et MISO. Le signal d'horloge et le signal de réinitialisation sont par contre remis en forme au moyen d'une limitation de tension par diodes zener et des tampons à 3 états (*Tristate*) dans le 74HC126 (IC2). On a bien entendu tenu compte de ce que ces connections du port peuvent exercer une autre fonction dans un programme du microcontrôleur. C'est pourquoi l'interface de programmation est complètement hors circuit lorsqu'elle n'est pas

utilisée. Les tampons se trouvent alors dans un état à haute impédance et sont pour ainsi dire absents. Ce n'est que lors d'une réinitialisation du processeur sur la ligne DTR que les tampons redeviennent actifs. Cela n'est par ailleurs pas le cas lors d'une réinitia-

lisation effectuée en pressant le bouton sur la carte. Un étage tampon fonctionne aussi dans le 74HC126 parallèlement au bouton de réinitialisation. En fait, un FET canal P avec drain ouvert commandé par DTR joue le rôle d'un deuxième bouton de réinitialisation. Le signal de réinitia-

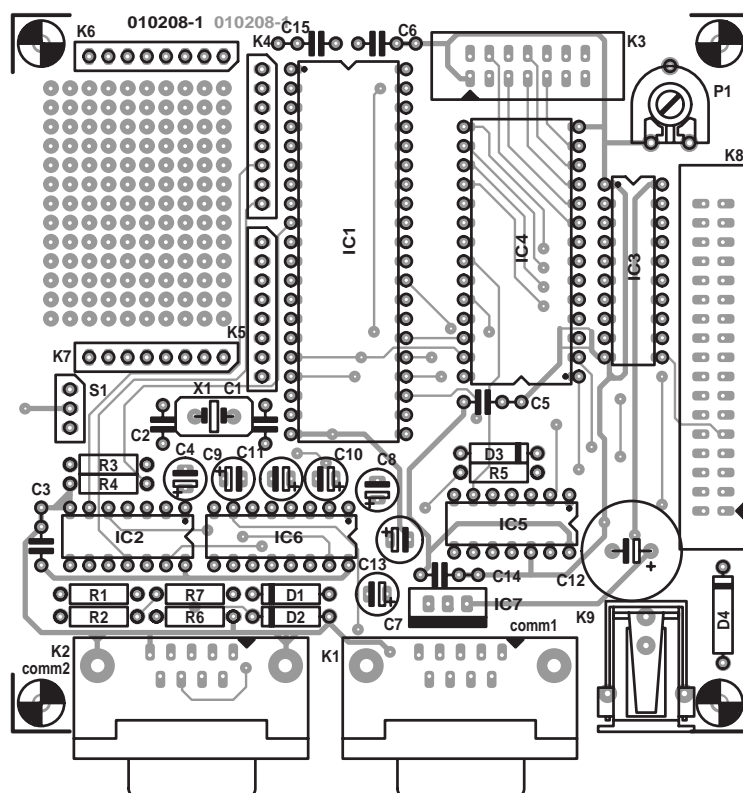


Figure 3b. ... schéma de montage.

Brochage de K8

1	GROUND
2	A15
3	GROUND
4	A14
5	+5V
6	A13
7	+5V
8	A12
9	ALE
10	A11
11	PSEN
12	A10
13	WR
14	A9
15	RD
16	A8
17	NOT CONNECTED
18	NOT CONNECTED
19	AD7
20	A7
21	AD6
22	A6
23	AD5
24	A5
25	AD4
26	A4
27	AD3
28	A3
29	AD2
30	A2
31	AD1
32	A1
33	AD0
34	A0

Liste des composants

Résistances :

R1, R2, R6, R7 = 27 k Ω
 R3, R5 = 1 k Ω
 R4 = 47 k Ω
 P1 = ajustable 10 k Ω

Condensateurs :

C1, C2 = 22 pF
 C3, C5, C6, C14, C15 = 100 nF céramique
 C4, C7 à C11 = 10 μ F/63 V vertical
 C12 = 1000 μ F/16 V vertical
 C13 = 100 μ F/10 V vertical

Semi-conducteurs :

D1, D2 = diode zener 4V7/ 500 mW
 D3 = BAT85
 D4 = 1N4001

IC1 = AT89S8252 24PC (Atmel)

IC2 = HC126

IC3 = 74HCT573

IC4 = 62256 (120 ns)

IC5 = 74HCT00

IC6 = MAX232 (Maxim)

IC7 = 7805

Divers :

K1, K2 = embase Sub-D à 9 contacts encartable en équerre

K3 = embase HE10 à 2 rangées de 7 contacts avec détrompeur

K4 à K7 = embase à 1 rangée de 7 contacts, mâle

K8 = embase HE10 à 2 rangées de 17 contacts avec détrompeur

K9 = embase jack pour adaptateur secteur

S1 = bouton-poussoir à contact travail

X1 = quartz 11,0592 MHz

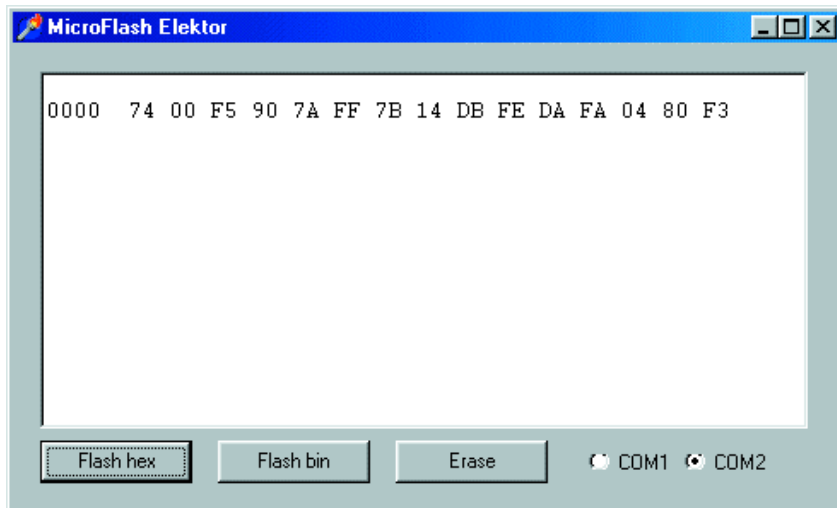


Figure 5. Programme de téléchargement.

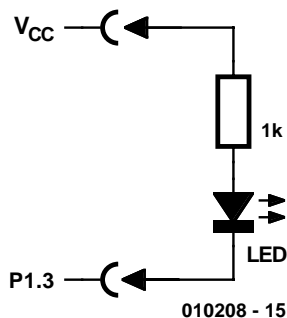


Figure 6. Premier test par LED branchée à une ligne du port.

lisation est donc toujours positionné à l'état actif haut et n'est jamais mis activement à la masse. Cela est possible parce que l'entrée du tampon

se trouve en permanence à Vcc alors que l'entrée d'activation est commandée par l'interface de programmation.

La programmation du contrôleur est un processus relativement complexe qui requiert tout d'abord la transmission de certains octets de commande. L'utilisateur n'a toutefois pas à s'en soucier : un outil de programmation sous Windows s'en charge. Le programme *MikroFlash.exe* se trouve sur la disquette **EPS010208-11** disponible auprès des adresses habituelles et peut aussi être obtenu par Internet sur la page d'accueil Elektor. Il est simple à manier : on active l'interface souhaitée COM1 ou COM2 et on choisit tout d'abord le format de fichier (Intel Hex ou binaire). On

sélectionne le fichier de programme approprié dans le menu de fichier qui s'affiche. La programmation démarre lorsque le fichier a été choisi et son état d'avancement peut être vérifié dans la fenêtre de texte. À la fin de la transmission, le programme remet automatiquement le contrôleur en mode RUN, ce qui fait démarrer le programme.

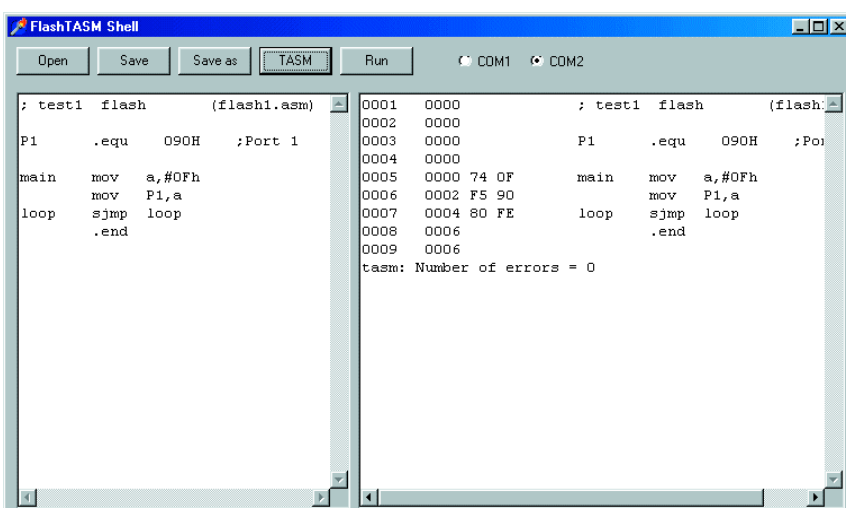
Test préliminaire

Un premier programme de démonstration est disponible sur la disquette. Le programme *Flashtest1.hex* ou *Flashtest1.bin* contient une courte boucle de programmation avec sortie sur le port en ordre ascendant. Une fois la transmission terminée, on trouve un signal rectangulaire sur toutes les broches de Port1. Le bon fonctionnement peut être vérifié facilement si on a un oscilloscope sous la main. Mais une LED avec résistance de protection raccordée entre VCC et une ligne de Port1 fait tout aussi bien l'affaire. On verra P1.7 clignoter lentement, P1.6, P1.5 à une fréquence 2 fois plus élevée et ainsi de suite. Cet exemple a par ailleurs été écrit en langage assembleur. Le **listage 1** donné ci-dessous reproduit le premier programme de démonstration *flash1.asm*.

```
; test1 flash #include 8051.H
.org 0000Hmain      mov
a,#00next      mov  P1,a
mov  r2,#255loop1  mov
r3,#20loop2      djnz  r3,loop2
                  djnz  r2,loop1
                  inc   a
                  sjmp  next
                  .end
```

Si la carte a réagi correctement, on peut commencer à développer ses propres programmes. L'utilisateur peut avoir recours aux outils de développement déjà disponibles. La carte accepte tous les programmes, qu'ils soient développés en langage assembleur, en Basic-52, Pascal ou C.

Les rôles sont pour ainsi dire inversés lors de l'utilisation de Basic-52. Les programmes Basic sont en effet traités comme des données par l'interpréteur. L'interpréteur lui-même est fichier programme de 8 K chargé directement dans le contrôleur. Il provient du légendaire BASIC-52-AH et a été mis gratuitement sur le marché par Intel. Il est donc permis de distribuer le logiciel au moyen de la disquette de programme ou par Internet. Il est aussi possible d'utiliser la version améliorée Basic-52 V1.3 actuellement disponible (Elektor n°272, février 2001, page 44 et suivantes). Le transfert dans le contrôleur demande de la patience car la transmission de 8 Koctets en



Listage 1. Le programme de test en assembleur permet de faire clignoter les LED.

série et leur écriture successive dans la mémoire morte Flash durent environ une minute. On peut se contenter d'utiliser le système comme ordinateur BASIC. Mais les programmes BASIC ne sont pas sauvegardés après la mise hors tension car ils se trouvent en RAM. Il faut avoir recours au langage assembleur ou à C pour développer le logiciel d'une utilisation permanente.

Toutes les applications n'ont pas besoin d'une RAM. Les programmes simples en assembleur s'en passent souvent. Dans ce cas, la carte peut être utilisée avec un équipement réduit ne comportant par exemple que IC1, IC2 et IC3. Pour une application définitive, il est possible de simplifier plus encore

la configuration. Le microcontrôleur IC1 suffit lorsqu'il s'agit d'une véritable application mono-puce. On peut évidemment effectuer un pas de plus et retirer la puce programmée de son support pour la placer sur sa propre carte. La carte sert à la fois de système de développement et de programmeur. Un exemple rendra les choses plus claires.

Il s'agit de développer un petit robot à commande par microcontrôleur. On monte tout d'abord les capteurs et les pilotes nécessaires dans la zone d'essai de la carte, puis on développe le programme de commande.

Dès que le fonctionnement est satisfaisant, on retire le microcontrôleur de son support et on réalise une carte de commande aussi petite que possible pour le robot. Le contrôleur se contente en gros d'un quartz et de quelques condensateurs. Et pendant que le robot se dirige automatiquement en cercle sur le tapis, un nouveau processeur se trouve déjà dans la carte de développement pour un nouveau projet. Il s'agira peut-être cette fois d'un appareil de mesure spécial pour le laboratoire d'électronique.

(010208-1)

Aperçu du cours de base sur les microcontrôleurs

Les générations d'accros de la technique se suivent. Mais même l'électronicien expérimenté ne cesse jamais d'apprendre. Elektor a reçu ces derniers temps un grand nombre de demandes dans ce sens : comment se familiariser avec la technique des microcontrôleurs ? Notre réponse : le nouveau cours de base sur les microcontrôleurs qui commencera dans le fascicule de janvier 2002 et comportera probablement 6 parties.

Certes, il existe déjà de nombreux livres sur le sujet et Internet offre une foison d'applications concrètes. Mais on suppose la plupart du temps des connaissances de base. Il est difficile pour le débutant de s'y retrouver. Quel contrôleur doit-on adopter, quel logiciel utiliser ? Où se situent les limites de systèmes particuliers, quel langage de programmation se prête-t-il à quelle application ? Ce cours de base est destiné à dissiper les ombres du labyrinthe de la technique des microcontrôleurs.

Le cours utilise le système Flash présenté dans ce fascicule. Il mentionnera donc un grand nombre de possibilités d'emploi de divers langages de programmation. Le cours utilisera le langage assembleur, Basic et C. Chacun pourra donc comparer les avantages et les inconvénients de ces langages. Mais loin de n'être qu'une introduction aux langages de programmation, le cours sera principalement consacré à des applications typiques des microcontrôleurs. Les procédés et les techniques de mesure, de commande et de régulation seront aussi inculqués, tout comme les particularités du contrôleur utilisé et de la famille 8051 en général.

Quels avantages apporte la participation à ce cours ? Essayons de les résumer en quelques mots essentiels.

L'utilisateur reçoit une introduction approfondie en techniques générales des applications sur microcontrôleurs, de l'utilisation de la mémoire à la gestion de composants périphériques complexes en passant par l'utilisation des ports et interfaces. Une comparaison approfondie des langages de programmation permet de choisir en toute connaissance de cause les outils appropriés. Les programmes développés lors du cours peuvent être utilisés avec presque tous les systèmes 8051. Le participant au cours se verra

toujours plus en mesure de concrétiser ses propres idées avec un microcontrôleur.

Les langages de programmation suivants sont utilisés :

- assembleur (plus précisément : l'assembleur partiel TASM bien connu complété par une interface Windows)
- Basic-52 (également la nouvelle version 1.3 avec un éditeur Windows et un programme d'émulation de terminal)
- Le compilateur C Reads51 gratuit de la société Rigel qui fonctionne lui aussi sous Windows

Pour ceux dont la phrase précédente a excité la curiosité, le logiciel READS51 peut être chargé directement sur Internet :

(<http://www.rigelcorp.com/>). La société Rigel Corporation, Gainesville, FL, USA fabrique des produits pour microcontrôleurs, entre autres dans le domaine de l'enseignement, soutenus par le logiciel gratuit pour les utilisations non commerciales. Elektor a été aimablement autorisé à utiliser le logiciel pour son cours de base sur les microcontrôleurs. Le programme d'émulation de terminal en Basic RbHost est, lui aussi, digne d'attention, au même titre que les nombreux renseignements sur la page d'accueil de Rigel.

Le cours de base sur les microcontrôleurs doit permettre de comparer ces 3 langages de programmation en accomplissant des tâches identiques ou similaires dans chacun d'entre eux. Cela permet de comparer leurs particularités, leur vitesse, les styles de programmation, etc. Si la table des matières du cours n'existe encore qu'à l'état d'ébauche, on dispose déjà d'un sommaire grossier de chaque sujet. Mots-clés :

- Premiers pas dans les langages de programmation, accès aux ports, boucles temporelles
- Utilisation de l'interface série
- Emploi des temporisateurs du 8051, des compteurs, du fréquence-mètre, de la modulation PWM
- Commande de l'afficheur LCD, unité de sortie de texte par RS232
- Connexion et commande de convertisseurs analogiques-numériques et numériques-analogiques
- Extensions par le bus I²C, extension de ports, A/D-D/A, horloge en temps réel
- Particularités du 89S8252, l'EEPROM interne 2 K et le temporisateur de surveillance (Watchdog-Timer)

LTC490

Pilote interface RS485/422

Microcontrôleurs & Périphériques

ELEKTOR

INFOCARTE 12/2001

Caractéristiques électriques ($V_{CC} = 5\text{ V} \pm 5\%$, T_{amb} sur la plage des températures de service)

Symbole	Paramètre	Conditions	Min.	Typ.	Max.	Unité
V_{OD1}	Tension de sortie du pilote différentielle (hors charge)	$I_0 = 0$			5	V
V_{OD2}	Tension de sortie du pilote différentielle (en charge)	$R = 50\ \Omega$ (RS422)	2			V
		$R = 27\ \Omega$ (RS485)	1,5		55	V
DV_{OD}	Différence des tensions en mode push-pull des pilotes pour niveaux de sortie complémentaires	$R = 27\ \Omega/50\ \text{W}$			0,2	V
V_{OC}	Tension de sortie en mode commun du pilote	$R = 27\ \Omega/50\ \text{W}$			3	V
$\Delta V_{OC} $	Différence des tensions en mode commun des pilotes pour niveaux de sortie complémentaires	$R = 27\ \Omega/50\ \text{W}$			0,2	V
V_{IH}	Tension d'entrée High (D)		2,0			V
V_{IL}	Tension d'entrée Low (D)				0,8	V
I_{IN1}	Courant d'entrée D				± 2	μA
I_{IN2}	Courant d'entrée A, B	$V_{CC} = 0\text{ V}/5,25\text{ V}$, $V_{IN} = 12\text{ V}$			1	mA
		$V_{CC} = 0\text{ V}/5,25\text{ V}$, $V_{IN} = -7\text{ V}$			-0,8	mA
V_{TH}	Tension de seuil à l'entrée différentielle du récepteur	$-7\text{ V} \leq V_{CM} \leq 12\text{ V}$	-0,2		0,2	V
ΔV_{TH}	Hystérésis du récepteur	$V_{CM} = 0\text{ V}$		70		mV
V_{OH}	Tension de sortie du récepteur High	$I_0 = -4\text{ mA}$, $V_{ID} = 0,2\text{ V}$	3,5			V
V_{OL}	Tension de sortie du récepteur Low	$I_0 = 4\text{ mA}$, $V_{ID} = -0,2\text{ V}$			0,4	V
I_{OZR}	Courant de sortie du récepteur Tri-state	$V_{CC} = \text{Max.}$, $0,4\text{ V} \leq V_0 \leq 2,4\text{ V}$			± 1	μA
I_{CC}	Consommation de courant	Hors charge, D = Masse ou V_{CC}		300	500	μA
R_{IN}	Résistance d'entrée du récepteur	$-7\text{ V} \leq V_0 \leq 12\text{ V}$	12			k Ω
I_{OSD1}	Courant de court-circuit du pilote, $V_{OUT} = \text{High}$	$V_0 = -7\text{ V}$		100	250	mA
I_{OSD2}	Courant de court-circuit du pilote, $V_{OUT} = \text{Low}$	$V_0 = 12\text{ V}$		100	250	mA
I_{OSR}	Courant de court-circuit du récepteur	$0\text{ V} \leq V_0 \leq V_{CC}$	7		85	mA
I_{OZ}	Courant de sortie du pilote (Tri-State)	$V_0 = -7\text{ V} \dots +12\text{ V}$		± 2	± 200	μA

LTC490

Pilote interface RS485/422

Microcontrôleurs & Périphériques

ELEKTOR

INFOCARTE 12/2001

Fabricant :

Linear Technology

www.linear-tech.com/pdf/lt0490.pdf

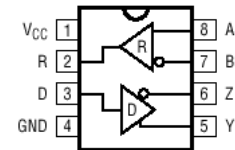
Caractéristiques techniques :

- Consommation de courant faible : $I_{CC} = 300\ \mu\text{A}$ typique
- Convient aux applications RS-485 et RS-422
- Alimentation asymétrique : $+5\text{ V}$
- La plage de mode commun qui s'étend de -7 à $+12\text{ V}$, permet une différence de masse de $\pm 7\text{ V}$ entre les périphériques connectés au bus
- Mise hors-fonction en cas de surcharge thermique
- Sorties de circuits de commande libres de parasites lors de la mise en et hors-fonction permettant l'insertion et la déconnexion des pilotes d'interface en cours de fonctionnement
- Lors de la mise hors-fonction le pilote reste à haute impédance
- Les impédances d'entrée et de sortie des circuits de commande permet la connexion d'un maximum de 32 émetteurs/récepteurs au bus.
- Hystérésis d'entrée (typique) : 70 mV
- Délais de transfert de 28 ns avec tolérance de 5 ns entre canaux
- Compatible broche à broche avec le SN7519

Description :

Le LTC490 est un circuit de commande de ligne ou de bus différentiel à dissipation faible conçu pour le transfert de données multipoint selon le standard RS-485 avec une plage de mode commun étendue (de -12 à $+7\text{ V}$). Le LTC490 respecte également le cahier des charges de RS-422.

La technologie CMOS permet des économies de courant significatives par rapport aux circuits intégrés bipolaires, sans que cela n'ait de conséquence sur l'insensibilité aux surcharges ou aux DES (Décharges Electro-Statiques). La présence d'une fonction de protection thermique élimine tout risque de dissipation dépassant les normes pouvant naître à la suite d'un conflit de bus, vu que dans ce cas-là les sorties passent à haute impédance. Le récepteur possède lui aussi un dispositif de protection qui met la sortie au niveau haut lorsque les entrées se trouvent en l'air.



Broche	Désignation	Fonction
1	V_{CC}	Tension d'alimentation positive ($4,75\text{ V} \leq V_{CC} \leq 5,25\text{ V}$)
2	R	La sortie du récepteur passe au niveau haut pour $A > B + 200\text{ mV}$ et au niveau bas pour $A < B - 200\text{ mV}$
3	D	Entrée pilote. Un niveau bas sur D force la sortie A au niveau bas et la sortie B au niveau haut ; un niveau haut sur D force A au niveau haut et B au niveau bas
4	GND	Masse
5	Y	Sortie du pilote
6	Z	Sortie du pilote
7	B	Entrée récepteur
8	A	Entrée récepteur

LTC490

Pilote interface RS485/422

Microcontrôleurs & Périphériques

ELEKTOR

INFOCARTE 12/2001

Application typique

Démultiplexeur DMX 8 voies

Elektor 12/2001

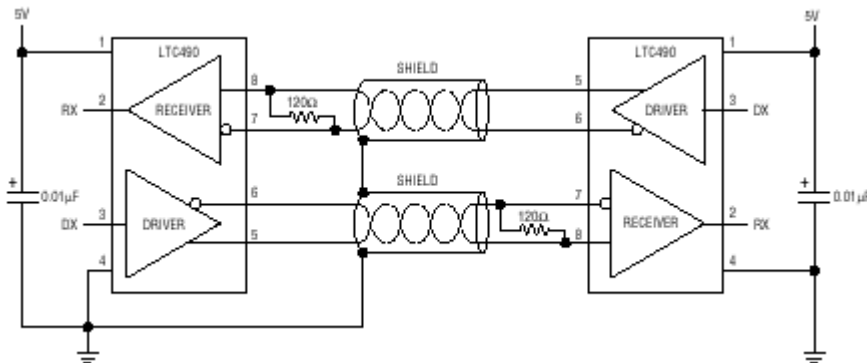


Types de boîtier et brochage

- LTC490CN8 DIP 8 broches plastique, plage de température de fonctionnement de 0 à 70 °C
- LTC490CS8 SOIC 8 broches plastique, plage de température de fonctionnement de 0 à 70 °C
- LTC490IN8 DIP 8 broches plastique, plage de température de fonctionnement de -40 à +85 °C
- LTC490IS8 SOIC 8 broches plastique, plage de température de fonctionnement de -40 à +85 °C

Valeurs maximales

Tension d'alimentation (V _{CC})	12 V
Courant d'entrée du pilote	-25 à +25 mA
Tension d'entrée du pilote	-0,5 à V _{CC} +0,5 V
Tension de sortie du pilote	±14 V
Tension d'entrée du récepteur	±14 V
Tension de sortie du récepteur	-0,5 à V _{CC} +0,5 V
Température de service (LTC490C)	0 à +70 °C
Température de service (LTC490I)	-40 à +85 °C



LTC490

Pilote interface RS485/422

Microcontrôleurs & Périphériques

ELEKTOR

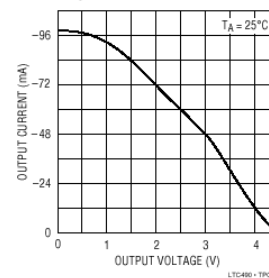
INFOCARTE 12/2001

Caractéristiques de commutation

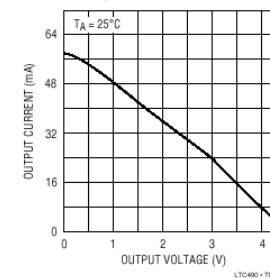
(V_{CC} = 5 V ± 5 %, T_{amb} sur la plage des températures de service, R_{DIFF} = 54 Ω, C_{L1} = C_{L2} = 100 pF)

Symbole	Paramètre	Min.	Typ.	Max.	Unité
t _{PLH}	Entrée vers sortie du pilote (de Low vers High)	10	30	50	ns
t _{PHL}	Entrée vers sortie du pilote (de High vers Low)	10	30	50	ns
t _{SKEW}	Différence d'une sortie à l'autre (pilote)		5		ns
t _r , t _f	Temps d'établissement/retombée du pilote	5	5	25	ns
t _{PLH}	Entrée vers sortie du récepteur (de Low vers High)	40	70	150	ns
t _{PHL}	Entrée vers sortie du récepteur (de High vers Low)	40	70	150	ns
t _{SKD}	Différence établissement/retombée (récepteur)		13		ns

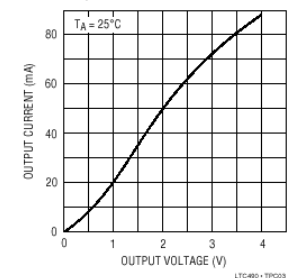
Driver Output High Voltage vs Output Current



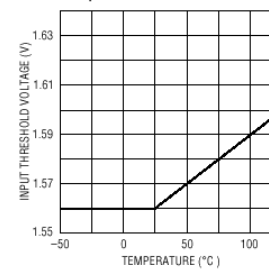
Driver Differential Output Voltage vs Output Current



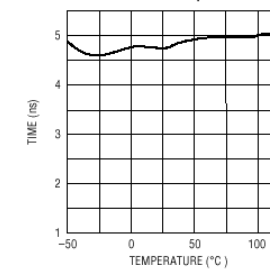
Driver Output Low Voltage vs Output Current



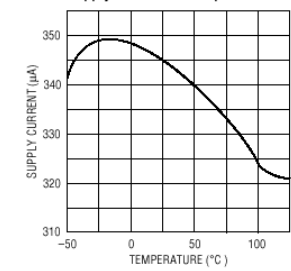
TTL Input Threshold vs Temperature



Driver Skew vs Temperature



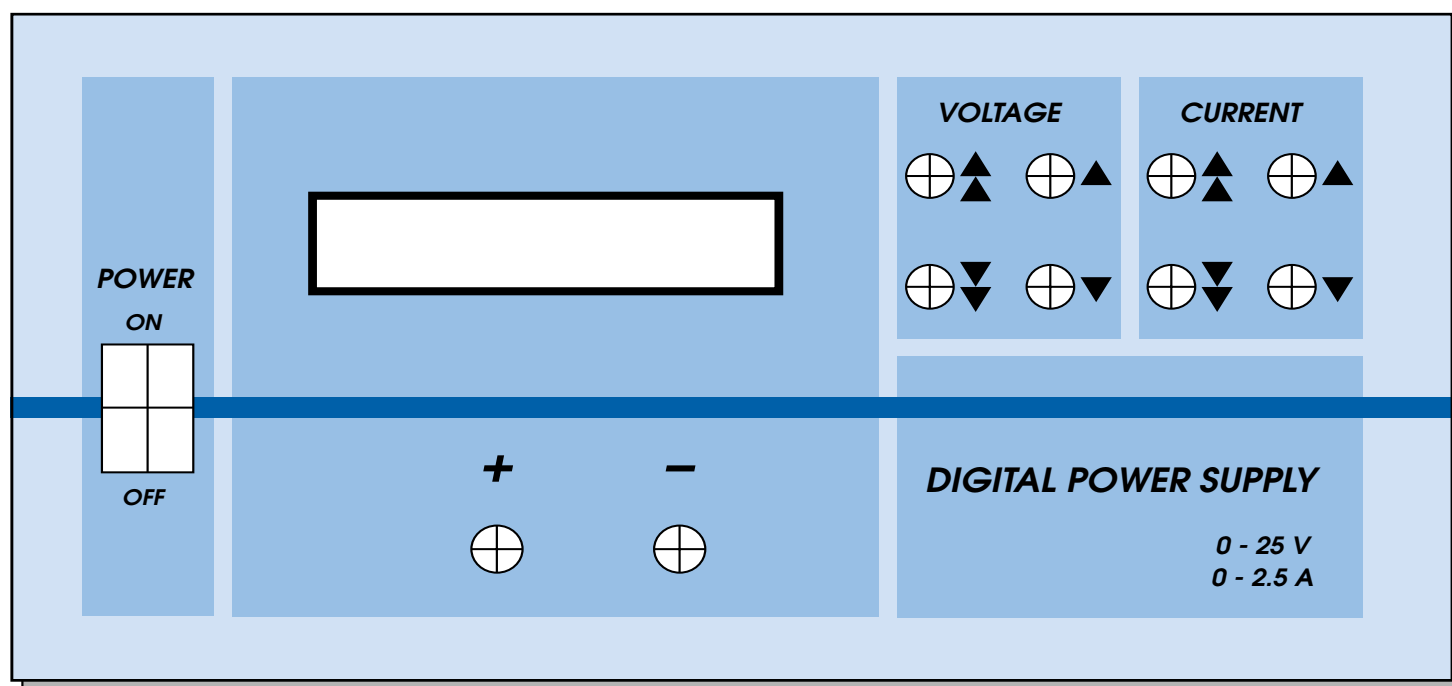
Supply Current vs Temperature



Alimentation numérique

Partie 2 : souder, scier, percer

La réalisation de cette alimentation numérique requiert une bonne dose de soin et de ponctualité. L'objectif final n'est-il pas en effet que, tant sous l'aspect mécanique qu'électronique, l'alimentation réalisée puisse prétendre faire partie des « professionnelles ».



000166 - F

Figure 1. Cette proposition de dessin de face avant peut se transformer en une face avant de toute beauté.

Il est possible, comme nous le disions dans la première partie, de réaliser 2 versions de cette alimentation. Nous n'avons pas dessiné de platine spécifique, qui partant aurait été plus petite, pour la version 1 A, ni prévu de dessin de face avant pouvant être utilisée sur un boîtier de dimensions plus faibles que celui prévu pour la « grosse » version de l'alimentation, mais nous nous sommes concentrés sur la version 2,5 A.

L'alimentation 2,5 A prend place dans un coffret métallique de 200 x 180 x 100 mm (l/p/h)

et requiert un transformateur torique 24 V/80 VA.

Il est facile dans ce cas-là de relier les parties du coffret avec la ligne de terre. Il faut en outre utiliser impérativement un filtre secteur si l'on veut éviter que le contrôleur ne procède à une réinitialisation au moindre problème au niveau de la tension secteur. La platine est fixée au-dessus du fond du coffret à l'aide d'entretoises, seuls le transformateur et,

éventuellement, comme sur la photo, le filtre secteur. Le dos du coffret reçoit le radiateur et, à côté de l'embase RS-232, une bride anti-arrachement pour le câble secteur, voire une entrée secteur (à porte-fusible intégré). L'idéal consiste bien évidemment à utiliser une entrée secteur dotée d'un filtre secteur. Le filtre que nous avons utilisé possédait les caractéristiques suivantes : 2L = 2,4 mH, Y = 2 200 pF, X2 = 0,1 µF et

$R = 1,0 \text{ M}\Omega$. Il devra bien entendu être en mesure de supporter l'intensité de courant maximum.

Nous allons commencer par réaliser le coffret. La réalisation de la face avant requiert précision et soin, elle constitue en effet la carte de visite de tout appareil de réalisation personnelle.

La platine vierge pourra servir de gabarit de perçage pour la réalisation de cette face avant. Il faudra veiller à laisser un espace suffisant (6 mm au minimum) entre l'embase secteur et la platine ! L'orifice rectangulaire destiné à l'afficheur LCD sera réalisé par sciage et mis aux dimensions requises à la lime, opé-

ration qui le débarrassera du même coup de ses barbes. Si vous ne vous sentez pas très à l'aise pour ce genre de travail, il sera préférable d'utiliser un affichage doté d'un cadre qui cachera les éventuelles imperfections du perçage rectangulaire. On procèdera de la même façon pour l'orifice de l'interrupteur secteur. En cas d'utilisation d'un interrupteur secteur rond on effectuera bien évidemment

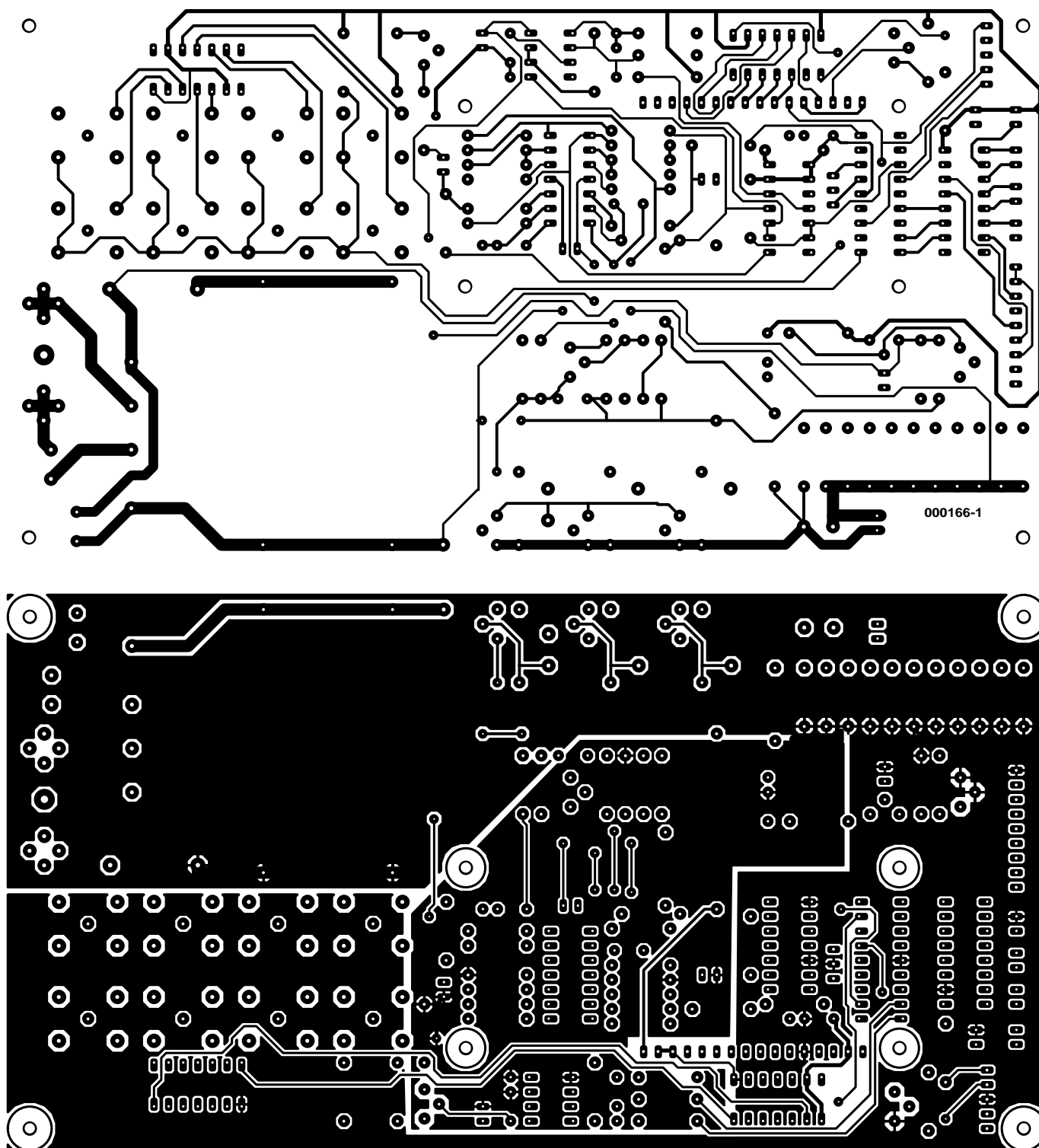


Figure 2a. Dessin des pistes de la platine double face à trous métallisés dessinée à l'intention de l'alimentation ...

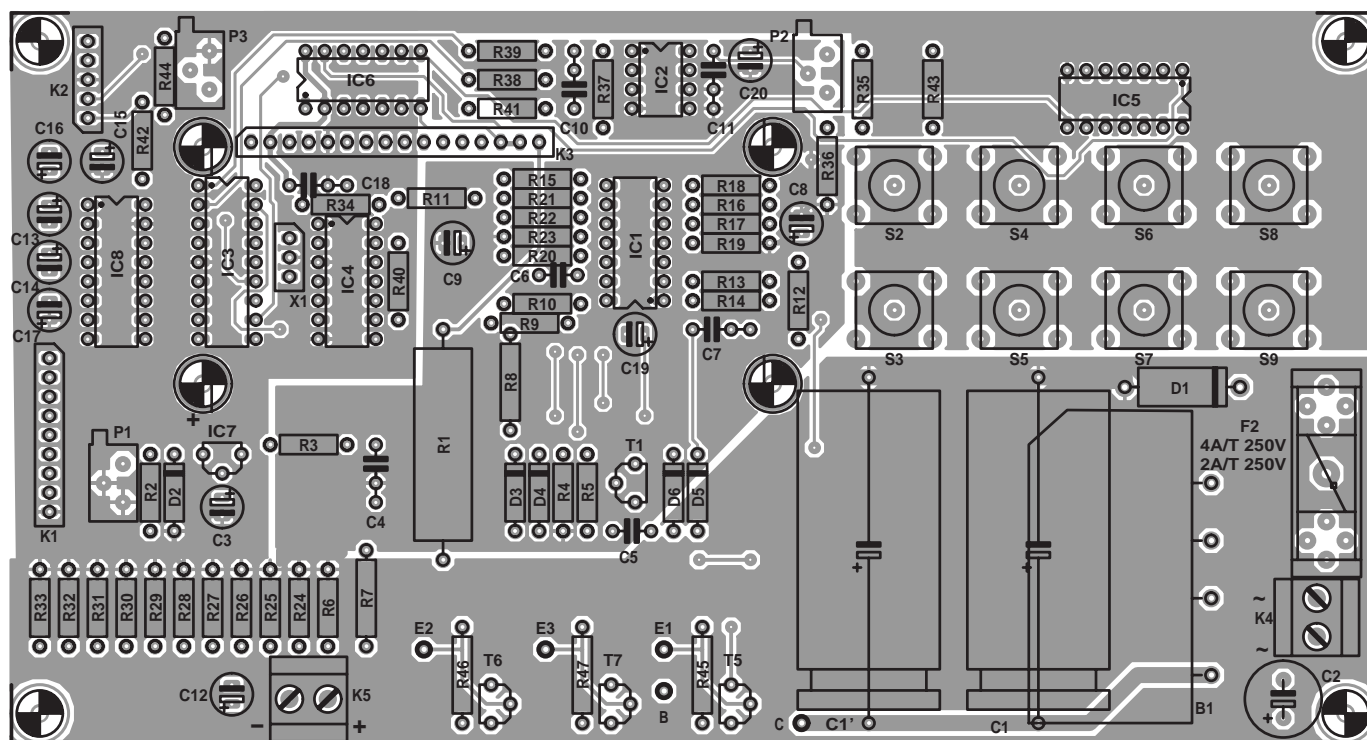


Figure 2b. ... et la sérigraphie de l'implantation des composants.

l'orifice à l'aide d'un dispositif à mèche ronde. Les orifices destinés aux boutons-poussoirs se feront à l'aide de mèches de diamètre de plus en plus important jusqu'à ce que l'orifice ait un diamètre plus grand de 0,5 mm à celui du bouton-poussoir et qu'il soit ébarbé proprement. Il reste à effectuer les 4 orifices destinés aux entretoises (vis) qu'il faudra intégrer de manière à ce que les vis à tête fraisée de 3 mm y disparaissent totalement. Sur l'arrière de la face avant on dote les filetages de vis d'écrous plats qui seront serrés avec force. Comme il n'est plus question de dévisser ces vis sans risquer d'abîmer la face avant, il est recommandé de sécuriser la fixation des écrous à l'aide d'une goutte de colle instantanée. Le filetage devra dépasser de quelques millimètres de manière à ce que les entretoises y trouvent un appui suffisant. Les orifices de fixation des 2 bornes de sortie seront percés en fonction du matériau auquel on a affaire.

Il suffira, si tant est que la face avant anodisée ou brossée soit encore, après toutes ces opérations, propre (sans rayures ou autre dégâts), mettre en place les décalcomanies des mentions et les protéger par quelques couches de vernis. On pourra également utiliser l'exemple de la figure 2 pour réaliser un film pour face avant. Il suffit de la reproduire par photocopie à la taille originale (échelle 1:1, vu qu'elle est représentée ici à taille réduite). On l'enduit ensuite de vernis (tenir

la bombe à 40 cm au moins pour éviter que le toner de la photocopie ne se mette à couler). Une fois le papier sec on l'enduit sur le dos de colle avant de le placer sur une page d'étiquettes autocollantes blanches qui en permettra la fixation simple sur presque n'importe quel arrière-plan et une éventuelle modification.

On attendra, avant de coller le film plastique autocollant sur la face avant, d'avoir bien testé l'alimentation; le lissage du film plastique doit toujours se faire du centre vers l'extérieur pour éviter la naissance de bulles d'air. Il ne reste plus qu'à passer un rouleau en caoutchouc ou un crayon parfaitement rond pour fixer définitivement le film sur le support. Les perçages et les orifices sont effectués ensuite, avec beaucoup de précaution, à l'aide d'un cutter. Le perçage des orifices destinés à recevoir les boutons-poussoirs requiert le plus grand soin sachant que cette opération ne peut se faire que depuis le côté film plastique et qu'elle exige de procéder par petits coups de cutter. On pourra également utiliser son PC et son imprimante pour fabriquer une face avant (cf. Elektor n°263, mai 2000 page 55).

Implantation des composants

Une fois que l'on n'a plus besoin de la platine à des fins détournées de gabarit de perçage, on pourra se mettre à l'implantation des composants sur la platine dont on retrouve le dessin des pistes et la sérigraphie en **figure 3**. Les problèmes sont rares vu qu'il n'y a ni composants CMS, ni pont de câblage ni pistes ultra-fines ni pastilles de soudage à trouver à la loupe. L'implantation des composants se fera dans l'ordre classique. Avant de vous y lancer, il faudra vérifier que les touches dépassent de 10 mm environ la section des condensateurs. Plusieurs composants prennent place sur le dos de la platine, à savoir le pont de redressement (pour des raisons de place), les ajustables, tous les picots, les oeillets de soudage pour les transistors de puissance ainsi que le fusible de manière à en faciliter le remplacement. En dernier lieu, les picots de connexion vers les bornes banane viennent, eux, prendre place sur le côté « composants ». Les composants se trouvant sous l'affichage, C19 par exemple, devront, si nécessaire, être montés à plat.

Liste des composants

pour la version 2,5 A (Version 1 A entre parenthèses)

Résistances :

(* à film métallique + tolérance de 1 %)

R1 = 220 Ω /5 W

R2 = 330 Ω /0W5

R3 = 150 Ω

R4 = 100 k Ω

R5 = 330 Ω (1 k Ω 8)

R6 = 1 k Ω /0W5

R7, R18, R23 = 1 k Ω *

R8, R17 = 39 k Ω *

R9, R10, R13, R14, R40, R41, R43 = 10 k Ω

R11, R12 = 47 k Ω

R15, R16 = 10 k Ω *

R19 = 8 k Ω 2

R20, R39 = 1 k Ω

R21 = 1 k* (0 Ω)

R22 = 18 k* (39 k Ω *)

R24 à R34 = 1 Ω /0W6*

R35 à R38 = 100 k Ω *

R42 = 47 k Ω

R44 = 3 k Ω 3

R45 à R47 = 0 Ω 51/0W5 (non implantées)

P1 = ajustable 10 Ω 64 X

P2 = ajustable 500 Ω 10LH

P3 = ajustable 2 k Ω 5 10LH

Condensateurs :

C1 = 10 000 μ F/50 V (4 700 μ F/35 V) horizontal

C2 = 470 μ F/50 V vertical

C3, C13, C14, C15, C16, C17 =

10 μ F/25 V vertical

C4, C7, C11 = 100 nF céramique RM5

C5 = 10 nF céramique RM5

C6 = 1 nF céramique RM2,5

C8, C9 = 10 μ F tantale

C10 = 4 nF7 FKS-2

C12 = 4 μ F/63 V MKS-4

Semi-conducteurs :

D1 = P600D (1N4007)

D2 = diode zener 12 V/1W3

D3, D4, D5, D6 = 1N4148

B1 = B80C5000 (B80C1500)

T1 = BC557B

T2, T3, T4 = TIP142 (1 seul exemplaire)

T5, T6, T7 = BC547B (non implantés)

IC1 = LT1491

IC2 = TLC272

IC3 = PIC16F84A-04P programmé (EPS000166-42)

IC4 = CD4066

IC5, 6 = 74HC164

IC7 = 78L05

IC8 = MAX232 (Maxim Integrated)

Divers :

F1 = fusible 2 A retardé

F2 = fusible 4 A retardé (2 retardé)

S1 à S8 = bouton-poussoir C&K

3FTL6 + IS09 22,5 ou No name 10-10-20, RM5

TR1 = transfo secteur torique 24 V, 80 VA, tel que, par exemple, UI 39/1721 V, 2-571 mA

X1 = résonateur céramique 4 MHz à 3 contacts

module LCD à 1 rangée de 16 caractères rétro-éclairé

porte-fusible encartable

porte-fusible châssis (ou intégré dans l'entrée secteur)

2 douilles pour châssis (noir, rouge)

F11 = filtre secteur 2 A (non implanté)

K1 = embase à 1 rangée de 9 contacts + embase sub-D à 9 contacts châssis droite

K2 = embase à 1 rangée de 5 contacts droite

K4, K5 = bornier encartable à 2 contacts RM5

support 14 broches de qualité

support 18 broches de qualité

12 picots

radiateur 180 x 75 x 48 mm à résistance thermique de 0,6 K/W (100 x 50 x 31 mm à résistance thermique de 2,4 K/W)

boîtier (l/p/h) 200 x 180 x 100 mm tel que, par exemple LC950 (Telet) interrupteur secteur, câble secteur

Si vous n'avez pas l'intention de programmer jamais le microcontrôleur sur la platine vous pourrez bien entendu vous passer du connecteur de programmation K2. On utilisera, au moins pour les 2 circuits intégrés les plus coûteux, IC1 et IC3, des supports de bonne qualité (tulipe). R1, R2 et R6 risquent de chauffer; il faudra partant les souder en laissant un petit espace entre la platine et eux. Il faudra s'assurer de l'absence (et le

cas échéant éliminer) de restes de flux de soudure à proximité de IC1, IC2 et IC4, sachant qu'ils pourraient entraîner des résistances parasites. Après avoir effectué le contrôle de toutes les soudures, on fixe l'affichage à l'emplacement prévu à l'aide d'entretoises de longueur convenable et on l'y visse. Les liaisons électriques se font à l'aide de morceaux de conducteur dénudé. On procède ensuite au montage de

la face avant, du transformateur secteur et l'interface PC, le tout étant câblé en respectant le plan de câblage représenté en figure 4.

Les transistors de puissance, isolés à l'aide de film aux silicones sont vissés sur le radiateur. La platine de liaison est reliée à la platine principale à l'aide 3 morceaux de conducteur d'une longueur de 10 cm environ (section de 0,75 mm²).

Première mise en fonction

Maintenant que vous en avez terminé avec la réalisation et le câblage vous n'avez sous doute pas envie d'attendre un mois avant de disposer du logiciel et du mode d'emploi de cette alimentation et préféreriez pouvoir la mettre immédiatement en marche. On pourra, bien que IC1 et le microcontrôleur ne soient pas encore mis en place, procéder à un premier test en mettant l'alimentation sous tension. Si tant est qu'il n'y ait pas de fumées suspectes, on pourra la laisser en marche.

On devrait mesurer alors entre les broches 4 et 11 du support de IC1, une tension de l'ordre de 27 V pour une alimentation 1 A et de quelque 32 V pour la version 2,5 A. On devrait relever sur la diode zener D2 une tension de 12 V et entre les broches 5 et 14 du contrôleur une tension de 5,1 V. Si toutes ces valeurs collent, on pourra, enfin, après avoir coupé l'alimentation, implanter les circuits intégrés restants. Met ensuite P2 en butée vers la gauche. Après remise sous tension l'affichage devrait indiquer :

00,0V 0,00A

Lors d'une action (maintenue) sur l'une des touches l'affichage doit basculer en mode de réglage et afficher la grandeur que l'on vient juste de modifier soit en milliampères soit en

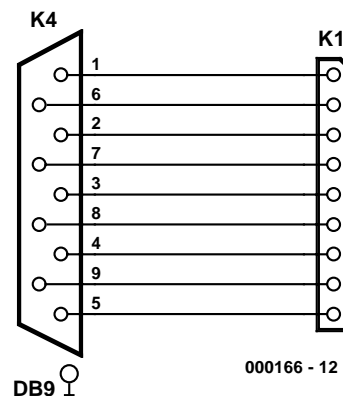


Figure 3. Liaison entre l'embase K1 et l'embase Sub-D.

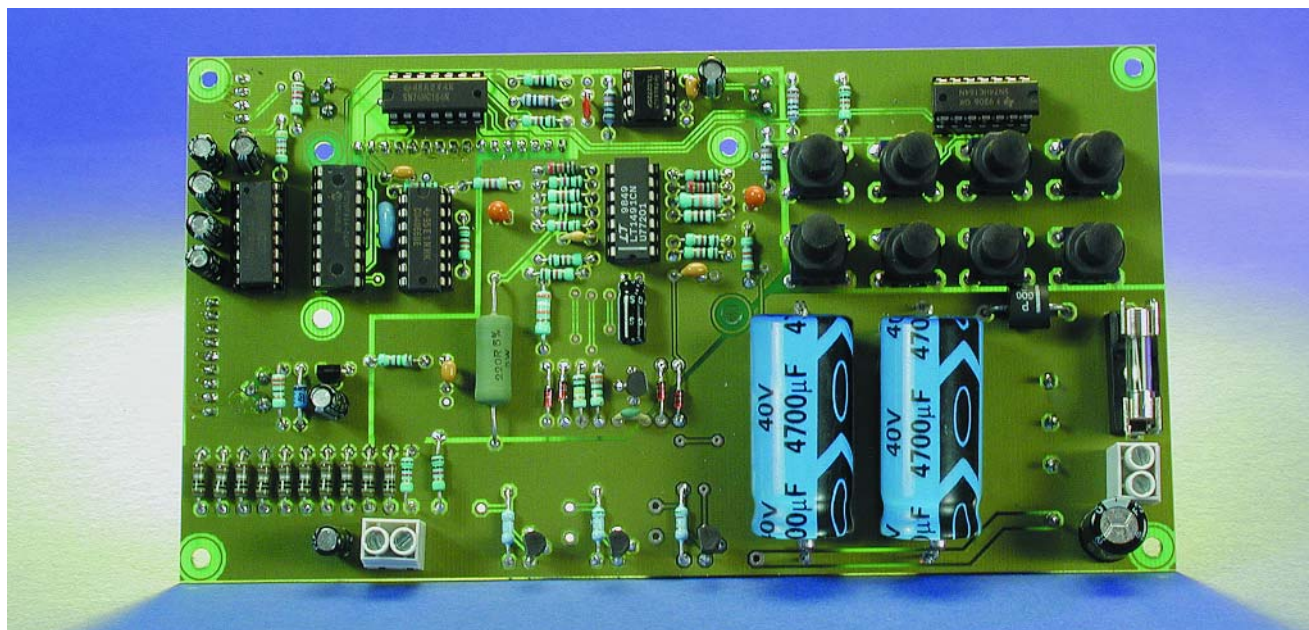
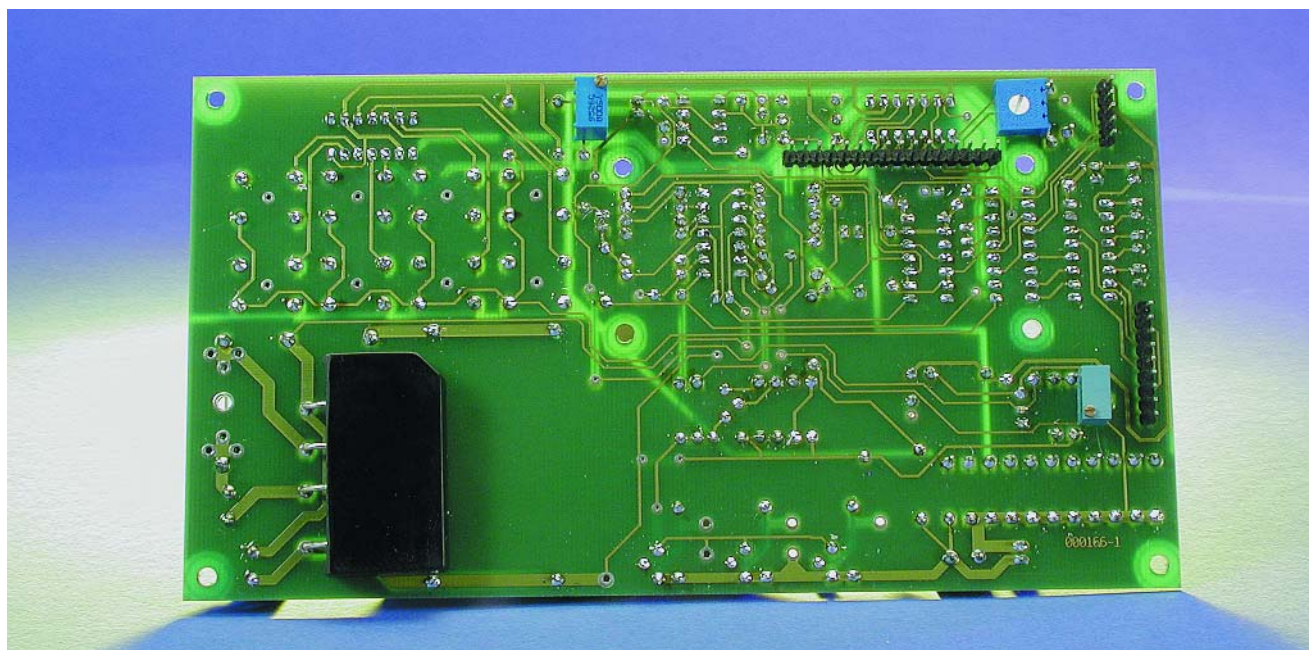


Figure 4. Coup d'oeil panoramique de la platine principale.



millivolts. On pourra, pour le réglage du point zéro, tourner progressivement P2 vers la droite et voir si l'affichage continue de changer. Si l'affichage ne change plus cela signifie que l'on est allé un peu trop loin et qu'il faudra tourner P2 dans le sens inverse (vers la gauche).

Autre possibilité, connecter l'interface de l'alimentation à un PC, lancer *Hyper Terminal* avec le paramétrage suivant :

9600,8N1, hardware handshake.

On joue alors sur l'ajustable de manière à ce que la valeur affichée après le D soit la plus

proche possible de zéro sans être nulle cependant. 2 serait une valeur idéale car elle évite que l'alimentation ne tombe en panne lorsque, ultérieurement, cette valeur varie légèrement suite à un vieillissement, des vibrations ou une éventuelle dérive en température.

Ce réglage terminé, on mesure la tension de sortie effective obtenue à une tension de consigne de 0,00 V. La valeur relevée devrait se situer aux alentours de 30 mV (± 15 mV). On demande ensuite à l'appareil une tension de consigne de 19,0 V et on

joue sur P1 jusqu'à obtenir en sortie une tension de 19,03 V précisément. Le courant de sortie et la valeur affichée sont alors automatiquement correctes. Il faudra, si l'on constate des dérives ou s'il est impossible d'obtenir lors de l'étalonnage les valeurs mentionnées plus haut, vérifier qu'il n'y a pas d'erreur au niveau de l'implantation des composants ou s'il ne reste pas des résidus de résine de soudage entre les pistes.

(000166-2)

ISAC (3)

L'environnement de développement logiciel

Prof. Dr. Bernd vom Berg, Dipl.-Ing. Peter Groppe et Dipl.-Ing. (FH) Michael Müller-Aulmann

Le sujet de ce 3^{ème} article de la série consacrée à ISAC, le contrôleur capteur/actionneur intelligent, est l'environnement de développement logiciel du MicroConverter d'Analog Devices. Il est temps, après avoir décrit ce composant dans le premier article et réalisé un système d'expérimentation dans le second, de nous mettre à écrire notre premier programme pour notre ISAC-Cube.

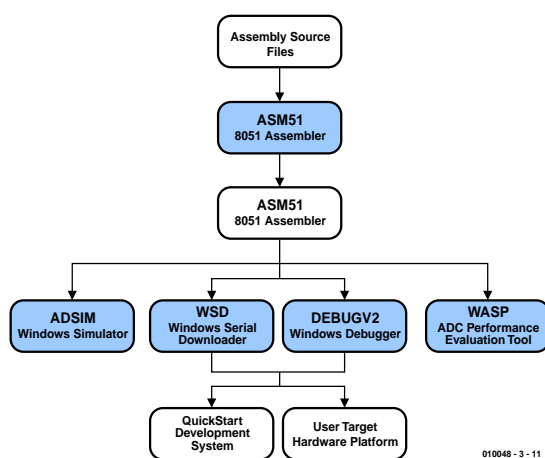


Figure 1. Développement de programmes pour le MicroConverter au niveau de l'assembleur.

Il est possible, en principe, d'utiliser, pour la programmation du MicroConverter d'Analog Devices, n'importe quel langage de programmation connu pour le 8051, Assembler51, Basic52, Pascal51, C51 pour ne citer que ceux-là. Tout ce dont on a besoin en fait est le fichier de format Intel Hex obtenu après l'opération de conversion, selon le cas, de l'assembleur ou du compilateur, fichier qu'il ne restera plus qu'à transférer, par voie série (ou parallèle), vers la mémoire Flash dont est doté le MicroConverter. Analog Devices propose pour sa part un envi-

ronnement de développement logiciel complet adapté de façon optimale à la famille du MicroConverter que l'on peut télécharger gratuitement depuis son site Internet, à savoir le logiciel baptisé « *MicroConverter QuickStart Development System, V3.0* » [2]. Cet environnement permet de développer des programmes tant en assembleur qu'en C.

Notons qu'il est également possible d'acheter chez Analog Devices leur « *QuickStart Development Kit* » qui comprend, outre le logiciel de développement, une carte d'expérimentation pour le ADuC812 avec adaptateur secteur, câble de transfert de données et 2 exemplaires de le ADuC812. Ceux d'entre nos lecteurs qui auront opté pour l'approche matérielle décrite dans le numéro du mois dernier (c'est-à-dire ISAC-Cube + platine d'expérimentation) pourront télécharger gratuitement d'Internet, outre l'environnement de développement logiciel, tout le reste de la documentation technique, à savoir la fiche de caractéristiques de ce composant, des exemples d'application et bien d'autres documents intéressants. Nous vous proposons dans les **tableaux 1** et **2** un récapitulatif des outils logiciels disponibles. Il faut distinguer, lors du développe-

ment de programmes pour les composants de la famille du MicroConverter, 2 trajets fondamentalement différents :

1. Développement et test des programmes en s'aidant du langage de programmation proche du langage machine Assembler51.
2. Développement et test des programmes en travaillant avec un langage de haut niveau tel que Basic52, Pascal51 voire C51, sachant que le dernier nommé, le C, peut être considéré actuellement comme la solution de professionnels de l'industrie.

Assembler51

Le trajet Assembler51 est illustré en **figure 1**. On utilise un éditeur de texte tout ce qu'il y a de plus standard pour écrire le fichier en source assembleur en code assembleur 8051. Ensuite, on pourra transférer le fichier Intel-Hex produit par le biais de l'assembleur 8051 vers le simulateur (*adsim812*) qui tourne sous Windows. Cet outil permet de simuler, on s'en serait douté, sur le PC, le comportement total du MicroConverter, y compris celui de ses périphériques enfouis (*on-chip*) tels que les convertisseurs A/N et N/A, au

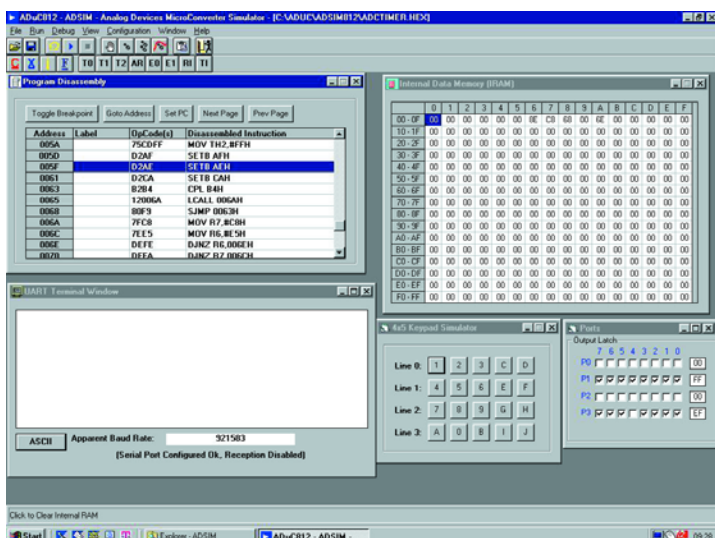


Figure 2. Le simulateur pour ADuC812 à base de logiciel.

niveau assembleur, sans qu'il ne soit nécessaire de connecter quelque matériel-cible que ce soit. De nombreuses possibilités de débogage, telles que le placement de points d'arrêt (*breakpoint*), l'exécution en mode pas à pas (*single step*), la manipulation des registres, la visualisation de la mémoire de donnée et bien d'autres fonctions sont égale-

ment intégrées dans cet outil. Grâce à lui on dispose d'un simulateur et d'un débogueur 100% logiciel permettant de recréer le ADuC812 dans son intégralité sur son PC (figure 2). Si l'on veut inclure dans la simulation ou le processus de débogage le matériel-cible concret en vue, par exemple, de tester les réactions réelles des périphériques (capteurs,

affichage LCD et autres claviers) on utilisera le programme appelé ADuC-Debugger qui se trouve relié directement à l'interface série du ADuC812 par le biais d'un port COMx du PC sur lequel tourne le logiciel de développement. Dans ces conditions, les résultats de débogage sont « extraits » directement du vrai matériel, les instructions de débogage sont transmises directement au matériel où elles sont également exécutées. Ici encore, on dispose de toutes les possibilités offertes par le programme de test : exécution en mode pas à pas, points d'arrêt, préprogrammation des registres, etc. On dispose, avec cet outil, d'un débogueur à base de matériel (figure 3).

Il sera possible, avec ce simulateur et/ou débogueur, dans de très nombreuses applications, d'éliminer les erreurs et de supprimer les sources de problèmes dans un programme. En conclusion : les outils Assembler51, le simulateur logiciel et le débogueur à base matériel constituent pour l'utilisateur un premier marche-pied pour faire ses premiers pas dans la programmation en assembleur des membres de la famille du MicroConverter.

C51

Analog Devices propose, pour le développement de programmes pour le MicroConverter en langage de haut niveau C51, une version « de mise en appétit » de l'EDI (**E**nvironnement de **D**éveloppement **I**ntégré, IDE pour *Integrated Development Environment* en anglais) μ Vision de la maison Keil, cette étant cependant limitée à une taille de programme de 2 Koctets. Keil est l'un des fabricants de EDI pour tous les dérivés du 8051 les plus connus [4]. Pour l'utilisateur, le choix de μ Vision présente le gros avantage de lui permettre de se faire la main sur un outil de développement professionnel extrêmement puissant, sachant cependant que cet apprentissage présente le « petit » inconvénient de

Tableau 1

L'environnement de développement de logiciels pour le MicroConverter

Programmation en langage machine : Assembler51

Cross-assembleur 8051 tournant sous DOS de MetaLink : asm51 (V1. 2h).

Simulateur pour le ADuC812 tournant sous Windows : adsim812.

Émulateur pour le ADuC812 tournant sous Windows : ADuC (V2).

Programmation en langage de haut niveau : C51

Environnement de développement intégré (IDE) μ Vision de Keil pour le MicroConverter (limité cependant à une taille de programme de 2 Koctets).

Transfert de programme (program downloader)

Téléchargeur sériel tournant sous Windows pour le transfert du fichier Intel-Hex dans la mémoire de programme en EEPROM Flash embarquée et la programmation de la mémoire de données enfouie (en EEPROM Flash elle aussi).

Outils d'analyse

Outil d'analyse tournant sous Windows pour le traitement des valeurs de mesure fournies par le convertisseur analogique/numérique (analyse FFT entre autres).

Documentations

Getting Started - Introduction

Guides utilisateurs (user guides) pour le MicroConverter

Tool Tutorials

Notes d'application / Exemples de code

Fiche de caractéristiques

Support en tout genre par outils tierce partie (3rd Party Tools)

Tableau 2

Autres outils en « C » pour le MicroConverter

Versions de test de produits commerciaux :

Raisonance Rkit-51 : (www.raisonance.com)

IAR Embedded Workbench (ICC8051) (www.iar.com)

Outils gratuits :

Reads51 (www.riegelcorp.com)

Small Device C Compiler -

SDCC (sdcc.sourceforge.net) pour Windows et Linux

demander un certain effort (si ce n'est un effort certain, cf. **figure 4**).

On pourra, pour se faciliter le premier contact, télécharger d'une part l'application UC002 « *Writing an ADuC812 Application in C* » de la page d'accueil du MicroConverter [2] et le guide d'application apnt 151 « *Using the Keil Monitor51 / μ Vision 2 Debugger with the Analog Devices ADuC812 Evaluation Board* » sur le site de Keil (www.keil.com) d'autre part. Ces documents décrivent très précisément, pas après pas, comment procéder pour réaliser son premier projet en C51 sous μ Vision et en effectuer la conversion pour disposer en fin de processus, le code Intel-Hex transférable vers le ADuC812.

Le dit EDI permet également une simulation confortable de votre programme écrit en C. Nous vous recommandons partant instantanément de télécharger ce guide d'application (en anglais) de bien l'étudier et d'exécuter les différentes étapes de l'installation de μ Vision sachant qu'au cours de nos prochaines excursions dans le monde logiciel du ADuC812 nous utiliserons exclusivement C51.

On pourra, comme nous le disons plus haut, outre les outils que nous venons de passer en revue, utiliser, pour le MicroConverter, tous les autres outils de programmation utilisables pour la famille 8051. Nous vous proposons dans le **tableau 2** une sélection d'outils utilisables avec le langage de programmation C.

Serial Downloader

Il faudra, après avoir généré le code Intel-Hex (code machine pour le ADuC812) définitif, que ce soit à l'aide de Assembler51, C51 ou tout autre langage de programmation pour le 8051, transférer le fichier de code de programme vers le ADuC812. Cette opération pourra se faire par le biais du ADuC-Debugger ou par l'intermédiaire du Serial Downloader d'Analog Devices tournant sous Windows, logiciel utilisable avec tous les produits de la famille MicroConverter (**figure 5**).

Ce processus très important de transfert de programme en direction du ADuC812 voire de notre ISAC-Cube se déroule en 3 étapes majeures (**figure 6**).

Étape 1 : il faut commencer par mettre le ADuC812 en mode Download/Debug (**figure 6a**). Pour ce faire il faut forcer la broche \overline{PSEN} du ADuC812 à la masse au travers d'une résistance de 1 k Ω (niveau bas sur la broche \overline{PSEN}) avant d'appuyer sur le bouton-poussoir de RAZ (Reset). Dans ce cas de l'ISAC-Cube cela se traduit par une action sur le bouton-poussoir T2 (« Boot ») suivie d'une action sur T1 (« Reset »), T2 restant enfoncé. On relâche ensuite d'abord T1 puis T2.

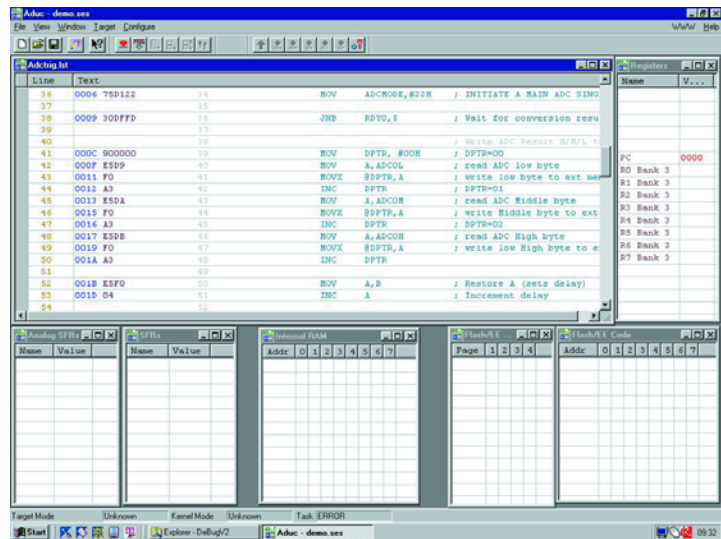


Figure 3. Le simulateur pour ADuC812 à base de matériel.

Entrons dans le détail pour les experts du 8051 :

1. Sur tout 8051 « standard » la broche \overline{PSEN} est toujours une sortie (signal de lecture pour l'EPROM contenant le programme). Dans le cas du MicroConverter la ligne \overline{PSEN} est activée, à l'instant de la RAZ voire immédiatement après le relâchement du bouton-poussoir de RAZ, pendant un court instant, en sortie, le niveau logique qu'elle véhicule à cet instant étant pris en compte. En présence d'un niveau bas sur la ligne \overline{PSEN} le ADuC612 bascule immédiatement en mode dit Download/Debug, ce qui signifie qu'il est mis en mesure de rece-

voir, par le biais de son interface série, un programme à transférer avant de s'en servir pour programmer sa mémoire de programme Flash embarquée.

2. Tout MicroConverter dispose d'une mémoire de programme « secrète, dissimulée » dans laquelle le fabricant a stocké les routines permettant un téléchargement sériel de programme, la programmation de l'EEPROM Flash et l'exécution des fonctions de débogage (Debug) évoquées plus haut. Lors d'un lancement de programme normal (mise sous tension du système ou action sur la touche de RAZ sans mise au niveau bas de la ligne \overline{PSEN}) ce domaine de mémoire secrète spécial

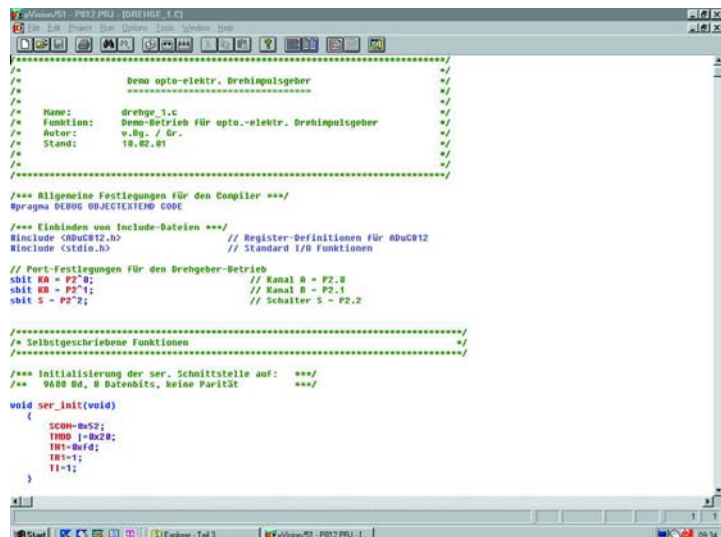


Figure 4. L'EDI μ Vision.

reste caché et partant inactif, ce qui signifie que le ADuC812 ne tient pas compte de son contenu et exécute tout simplement le programme d'application se trouvant dans l'EEPROM Flash (cas de fonctionnement normal). Si le ADuC812 détecte, lors d'une action sur la touche de RAZ ou lors du relâchement de la dite touche, la présence d'un niveau bas sur la ligne PSEN, il ignore le programme utilisateur stocké en EEPROM Flash et bascule vers la mémoire de programme secrète dont il exécute le programme, à savoir, selon le cas, la routine de transfert (download) ou de débogage (Debug).

Étape 2 : il est temps maintenant de lancer le programme Serial Downloader (figure 5).

Ce programme se met automatiquement à la recherche d'un MicroConverter qui devrait être connecté (par défaut) au port COM1 et se met en liaison avec lui. Si vous avez opté de connecter le ADuC812 à un port autre que COM1, vous pourrez modifier en conséquence, par le biais du point de menu « Configuration » le paramètre concerné. Ce menu permet de configurer un certain nombre d'autres paramètres de transfert. On pourra, au début, garder les valeurs par défaut. On appuie ensuite sur le point de menu « Download » et on choisit, comme on le fait d'habitude sous Windows, dans le répertoire concerné, le fichier Inter-Hex que l'on veut transférer. Le téléchargement se fait alors automatiquement et sans le moindre problème.

Étape 3 : une fois le processus de téléchargement terminé, on pourra faire disparaître le Serial Downloader « vers le bas » de la liste de programmes. Une action sur la touche Reset T1 du Cube démarre l'exécution du programme stocké dans le ADuC812 (figure 6b). Rappelez-vous, comme vous n'avez pas appuyé sur la touche T2, la ligne PSEN ne se trouve pas au niveau bas de sorte que le ADuC812 ignore sa mémoire de programme interne secrète.

Chaque nouveau processus de transfert redémarre à l'étape 1 : appuyer sur T2, puis sur T1, relâcher T1 puis T2, lancer le Serial Downloader...

Durée de rétention minimum

L'EEPROM Flash du ADuC812 est une mémoire « non-volatile » qui partant conserve son contenu même en l'absence de tension d'alimentation, sachant cependant que cette durée de maintien des données n'est pas illimitée dans le temps et que le nombre de téléchargements de nouveaux programmes connaît lui aussi une limite.

Analog Devices donne dans sa fiche de caractéristiques du ADuC812 2 informations importantes :

- Les données stockées dans la mémoire Flash sont conservées (avec ou sans tension d'alimentation) pendant 10 ans au minimum sachant qu'il y a risque, après cette durée, de « pertes » dues à des modifications erronées du contenu des cellules de mémoire.

- Le nombre minimum de reprogrammations possibles (effacement et nouvelle écriture des cellules de mémoire avant que ne puisse se produire l'erreur au niveau de la structure de la mémoire) est, d'après le fabricant, de 10 000 cycles.

Il ne faut pas oublier que le ADuC812 dispose de 2 domaines d'EEPROM Flash différents.

En ce qui concerne la mémoire de programme, cette possibilité de 10 000 reprogrammations est indubitablement plus que suffisante : on pourra se permettre d'opérer 500 jours à raison de 20 reprogrammations par jour !

Il en va tout autrement de la mémoire de données de 640 octets en EEPROM Flash dans laquelle il est possible de stocker n'importe quelle valeur et ce aussi souvent qu'on le voudra. Dans certaines circonstances il peut se faire que l'on atteigne très rapidement cette limite critique de 10 000 processus d'effacement/réécriture, voire qu'on la dépasse si l'on stocke, par exemple, dans les dites cellules de mémoire, des données de mesure ne cessant de changer. Il serait préférable, dans ces conditions, d'envisager une extension externe de la mémoire de données sous la forme d'une RAM (nous ne manquerons bien évidemment pas, ultérieurement, de vous montrer comment faire...)

Analog Devices ne s'est pas endormi



Figure 5. Le Serial Downloader.

sur ses lauriers et ne cesse d'améliorer des MicroConverters : sur les versions ultérieures (cf. le tableau 2 du premier article), il est mentionné une rétention des données pouvant aller jusqu'à 100 (!) ans et ce avec un minimum de 100 000 cycles effacement/réécriture.

Les émulateurs

Signalons, en guise de conclusion à cet article, qu'il existe aussi, pour le MicroConverter, des émulateurs. MetaLink (www.metaice.com) propose un émulateur classique pour le ADuC812, ce qui signifie que l'on remplace le MicroConverter du système-cible par un émulateur.

Accutron (www.accutron.com) propose le ACE « Single-Pin-Emulator » qui supporte tous les dérivés de la famille du MicroConverter y compris le « C »-Source-Debugging.

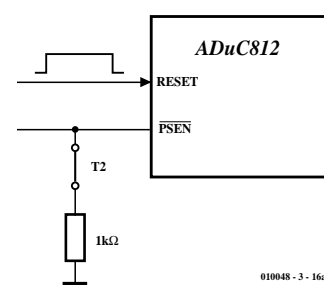


Figure 6a. Activation du mode de débogage (Debug) ou de transfert (programm download).

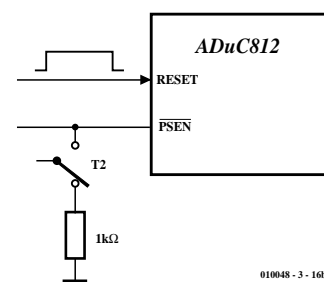


Figure 6b. Mode de fonctionnement normal.

Annexe technique :

Convertisseur A/N : mesure de la température interne de la puce

Le ADuC812 comporte un capteur de température à même la puce qui fournit une tension proportionnelle à la température, tension qui attaque le canal interne n° 9 (canal N° 8) du convertisseur. La fiche de caractéristiques donne les caractéristiques de ce capteur : tension de sortie de 600 mV à une température (u) de 25 °C, cette tension variant de -3 mV/°C, ce qui signifie que pour une augmentation de la température de 1 °C la tension diminue de 3 mV et inversement.

Il y a cependant, à ce niveau, un point extrêmement critique dont il faut tenir compte : ces valeurs de référence (600 mV, -3 mV/°C) sont des valeurs typiques et partant varient d'un lot de production à l'autre. Ceci implique que si vous utilisez des ADuC812 de lots différents, il se peut fort bien que les capteurs de température embarqués aient des valeurs sensiblement différentes. La seule garantie donnée par Analog Devices est le rapport (parfaitement) linéaire entre la température du composant et la tension de température qui en est dérivée : cette relation est exprimée par l'équation (linéaire) $U_v = f_1(u)$ ou $U_v = f_2(u)$.

Pour l'utilisateur cela signifie : commencer, à l'aide de 2 mesures de contrôle, par vérifier les paramètres de l'équation; ce n'est qu'ensuite que l'on pourra effectuer des mesures de température qui « tiennent la route ».

Si l'on utilise ensuite des ADuC812 du même lot (ce qui est souvent le cas lorsque l'on achète ce type de composant par barrette complète) il ne sera plus nécessaire de modifier quoi que ce soit. Si l'on utilise des ADuC812 de lots différents il faudra, dans le pire des cas, effectuer cet étalonnage un certain nombre de fois et le risque existe que vous releviez, pour chacun des ADuC812 un set de paramètres d'équation différents.

Si vous voulez utiliser le capteur de température embarqué pour mesurer non pas la température du composant mais la température ambiante, vous pourrez vous y prendre de la manière suivante pour déterminer les paramètres de l'équation (cf. le croquis) :

La forme générale d'une telle équation est la suivante :

$$y = f(x) = m \cdot x + b$$

ce qui se traduit, dans notre cas par :

$$v = f(U_v) = m \cdot U_v + v_b$$

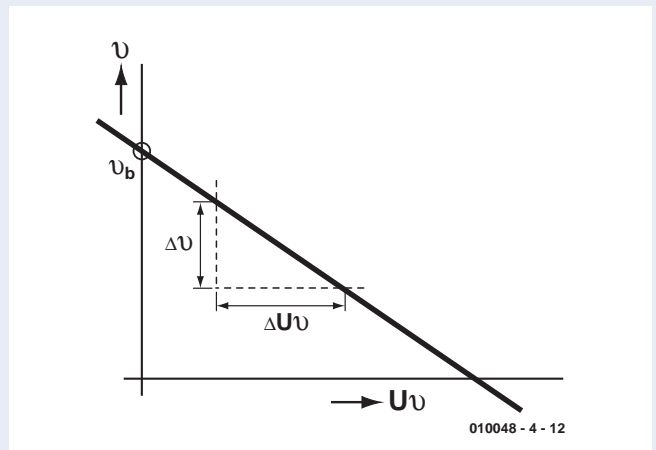
soit, si D représente la valeur de conversion du convertisseur A/N, canal N° 8, :

$$v = f(D) = m \cdot D + v_b$$

avec :

- v = température ambiante recherchée
- U_v = tension de température relevée ou résultat de conversion D du convertisseur A/N, canal N° 8,
- m = croissance de la ligne de courbe de température (droite)
- v_b = intersection avec l'axe des y, c'est-à-dire point de coupe de la ligne de courbe de température avec l'axe u.

La ligne de courbe de température est parfaitement définie à l'aide des termes m et b; ces 2 paramètres doivent ensuite être déterminés par le biais de mesures de référence (c'est-à-dire de 2 mesures de température distinctes).



Nous allons, pour cela, utiliser notre programme `adc_1.c` et, après lancement, choisissons le canal 8 (utilisation de la tension de référence interne du convertisseur A/N de 2,5 V). On fait ensuite fonctionner le ADuC812 à une température ambiante que l'on mesure également à l'aide d'un autre thermomètre précis (un thermomètre à affichage LCD par exemple). On procède, au bout d'1 heure environ, durée pendant laquelle la température ambiante doit rester constante, à la lecture des 2 valeurs de mesure :

- v_1 = température ambiante relevée à l'aide de l'instrument de référence, disons 21 °C
- D_1 = valeur de mesure résultant de la conversion numérique du canal N° 8 du convertisseur N/A, disons 1039 (décimal)

On expose ensuite le ADuC812 à une seconde température en le mettant, par exemple, dans un réfrigérateur pendant 1 heure environ. On détermine à nouveau, à l'aide du thermomètre de référence, la température régnant à l'intérieur du réfrigérateur. Ces opérations fournissent 2 valeurs de mesure :

- v_2 = température dans le frigo relevée à l'aide de l'instrument de référence, disons 6 °C
- D_2 = valeur de mesure résultant de la conversion numérique du canal N° 8 du convertisseur N/A, disons 1289 (décimal)

À partir de ces 4 valeurs de mesure il nous est possible de déterminer les 2 paramètres de la courbe recherchés m et v_b . (Moto : forme à 2 points de l'équation). On a :

$$m = (v_1 - v_2) / (D_1 - D_2) \text{ c'est-à-dire ici :}$$

$$m = (21 - 6) / (1039 - 1289) \text{ °C/digit}$$

$$m = -0,06 \text{ °C/digit}$$

$$v_b = (v_2 \cdot D_1 - v_1 \cdot D_2) / (D_1 - D_2) \text{ c'est-à-dire ici :}$$

$$v_b = (6 \cdot 1039 - 21 \cdot 1289) / (1039 - 1289)$$

$$v_b = 83,34 \text{ °C.}$$

$$v_1 - v_2$$

Il est possible ainsi de rendre la courbe de température de votre ADuC812 par l'équation suivante :

$$\vartheta = -0,06 \text{ }^{\circ}\text{C/digit} \cdot D + 83,34 \text{ }^{\circ}\text{C}.$$

équation dans laquelle D est la valeur numérique du moment fournie par le convertisseur A/N pour son canal N° 8. IL suffit par-tant de programmer cette équation, d'entrer à chaque fois la valeur relevée pour D et l'on sait instantanément à quelle température travaille son ADuC812.

Nous avons bien évidemment déjà faite le travail pour vous et vous proposons à cet effet le programme *temp812.c*.

Il faut dans ce cadre tenir compte de 3 points :

1. Normalement, le capteur de température embarqué relève la température de la puce, qui est, en règle générale, supérieure à la température ambiante. Nous avons, par le biais de nos 2 mesures de calibration, combiné la valeur numérique mesurée avec la température ambiante, de sorte que c'est cette dernière que nous mesurons.
2. Les valeurs correspondantes pour m et Jb ne valent que pour notre propre ADuC812. Il vous faudra, dans le cas de votre ADuC812 (qui fait sans aucun doute partie d'un autre lot), déterminer vos propres valeurs pour m et Jb à l'aide de la procédure décrite plus haut et les intégrer dans l'équation.
3. Il faudra, pour déterminer la température ambiante à l'aide de l'équation donnée plus haut, faire travailler votre programme à

l'aide de nombre fractionnaires (nombres flottants). Cela ne pose bien entendu pas le moindre problème sous C51. Il se veut malheureusement que les versions d'évaluation de mVision mises à votre disposition ont été amputées de l'arithmétique flottante ce qui a pour conséquence que vous vous trouvez dans l'impossibilité de convertir le programme *temp812.c*, ce qui se traduit par l'apparition de messages d'erreur.

À titre de compensation nous vous proposons, sur Internet (www.elektor.presse.fr), le fichier hexadécimal de notre programme que vous chargez dans votre ADuC812 pour le démarrer à partir de là.

Le programme commence par demander les valeurs correspondant à m et Jb, valeurs que vous pouvez entrer à ce moment-là. Vous procédez ensuite à la mesure des températures à l'aide de votre propre set de paramètres.

Il vous faudra, si vous tenez à calculer en nombres fractionnaires dans le cadre de vos programmes, soit acquérir la version complète de mVision, soit opter pour un autre compilateur C51.

Tableau pour clavier sériel connecté à une broche du port d'entrée analogique

Ce tableau donne la relation entre la touche actionnée, la tension d'entrée Uent qu'elle produit, le mot numérique converti, l'intervalle de scrutation de touche sûr et le niveau de priorité de touche (0 = niveau de priorité le plus élevé).

Touche actionnée	Tension sur la sortie analogique (en pratique) [V]	Nombre numérique [12 bits, décimal]	Intervalle de scrutation de touche [décimal]	Niveau de priorité de touche
Touche 0	24,4 mV	20	0 à 184	0
Touche 1	0,481	394	185 à 604	1
Touche 2	0,963	789	605 à 999	2
Touche 3	1,484	1216	1000 à 1426	3
Touche 4	1,996	1636	1427 à 1846	4
Touche 5	2,503	2052	1847 à 2262	5
Touche 6	3,027	2481	2263 à 2691	6
Touche 7	3,528	2892	2692 à 3102	7
Touche 8	4,026	3300	3103 à 3510	8
Touche 9	4,546	3726	3511 à 3936	9
Aucune touche	4,996	4095	> 3937	10

L'émulateur est connecté au système-cible par le biais de la broche \overline{EA} . À l'image de la broche \overline{PSEN} , cette broche remplit également plusieurs fonctions, toutes décrites tant dans la fiche de caractéristiques que dans le manuel de l'utilisateur (*User's Manual*) du MicroConverter. C'est en particulier dans le cas des dérivés à convertisseur analogique/numérique 16 et 24 bits du MicroConverter que ce type d'interconnexion au système-cible présente de gros avantages en raison de sa simplicité. L'utilisateur professionnel dispose ainsi d'un outil de développement très puissant.

Après vous avoir fait faire connais-

sance avec toute la panoplie des différents outils qui vous permettront de transférer vos programmes dans le ISAC-Cube, nous envisageons, dans le prochain article, de passer aux choses sérieuses en vous proposant un certain nombre d'applications intéressantes écrites en C51. Notons qu'Analog Devices travaille à une nouvelle version plus élaborée de ses outils de développement destinés aux MicroConverters. Il est recommandé partant de faire, de temps à autre, plus ou moins régulièrement, un petit tour sur le site d'accueil du MicroConverter (www.analog.com/microconverter).

(010048-3)

Littérature et sites Internet :

- [1] Bernd vom Berg, Peter Groppe: « Je programme en Pascal les microcontrôleurs de la famille 8051 », Publitrone, ISBN 2-86661-098-9
- [2] La patrie du MicroConverter sur Internet : www.analog.com/microconverter
- [3] Analog Devices sur Internet : www.analog.com
- [4] EDI mVision de la société Keil : www.keil.com

ALIMENTATIONS

Adaptateur auto pour portables	11	32
Alimentation à découpage « auto »	06	46
Alimentation numérique (I)	11	52
Alimentation numérique (II)	12	30
Alimentation secteur à découpage symétrique (+/-)	7/8	25
« Béquilles » pour régulateur de tension	7/8	91
Chargeur pour accu Li-ion Mark II	7/8	88
Chargeur pour accu Li-ion modèle 1	7/8	44
Commande de tension alternative à FETMOS	7/8	101
Commutatrice continu/continu	7/8	10
Convertisseur haute-tension : 90 V depuis 1,5 V	7/8	107
Convertissez votre alim de labo en alim double	7/8	11
Déchargeur pour pack d'accus	7/8	93
Indicateur de fonctionnement et de destruction de fusible	7/8	49
Pilote « High Side » double	7/8	99
Protection anti-court-circuit à FETMOS	7/8	30
Protection pour Li-ion	7/8	33
Régulateur à découpage double	7/8	39
« Tire-jus » pour piles	7/8	102
Une alimentation haute tension	01	08

APPAREILS DE MESURE & DE TEST

Ampèremètre	7/8	83
« Ampèremètre » LM 3822/24	7/8	71
Analyseur de code IR	10	12
Capteur de température I ² C	7/8	57
Chronomarqueur à petit prix	7/8	76
Émetteur traque-fil	7/8	17
Générateur de formes d'ondes à large bande	7/8	22
Générateur d'impulsions programmable	03	46
Générateur sinusoïdal 3 phases numérique	7/8	103
Jauge de niveau à haute précision	7/8	118
Mesure de niveau « alternative »	7/8	51
Mesure économique de température par microcontrôleur	7/8	53
Pilote de convertisseur N/A à basse fréquence	7/8	70
Récepteur traque-fil	7/8	28
Sélecteur d'impulsion	7/8	82
Testeur de transistor	7/8	92
Testeur de télécommande rustique	7/8	82
Thermomètre numérique à LED	04	52
Vide-accumulateur capacimètre (I)	11	26
Vide-accumulateur capacimètre (II)	12	14

AUDIO, VIDÉO & MUSIQUE

Adaptation de niveau	02	68
Ampli Hi-Fi P ³ : À tubes KT 88	05	40
Attaque de ligne vidéo	7/8	113
Audio-CAN 2000	03	08
Convertisseur rééchantillonneur 96 kHz	04	72
Convertisseur S/PDIF vers TTL	7/8	100
Convertisseur Vidéo/S-VHS	11	70
Crescendo - Millennium Edition	04	40
Crête-mètre stéréo	7/8	101
De VGA vers 3 x BNC	7/8	54
Écrêteur à tubes	11	58
Encodeur rotatif pour potentiomètre silencieux	7/8	80
Filtre passe-bas SC du 5 ^{ème} ordre de 1 Hz à 45 kHz	7/8	24

Haut-parleur actif pour PC	7/8	67
Interface port série -> MIDI	04	08
Métronome-diapason	06	16
PC-SoundAmp	01	46
Sortie symétrique	7/8	71
Switch audio/vidéo	7/8	52
« Switchbox » pour enceintes/amplificateurs	7/8	90
Transmission de données full duplex	7/8	127

CIRCUITS HF, RADIO

Audion à tube ECC86	7/8	78
Booster pour câble	7/8	60
Convertisseur Ondes Courtes pour autoradio AM numérique	7/8	112
Convertisseur S-VHS/Vidéo	09	79
Détecteur FM sans réglage	7/8	14
Filtre de parole	7/8	66
Microradio OM	11	64
Récepteur à M.A.	02	08
Récepteur radio large bande pour PC	7/8	59
Réseau d'adaptation HF universel	01	76
Source 2,5 GHz	7/8	123
Système d'identification d'appels radio	02	32
Une radio à transistors économe	7/8	63

DIVERS

Ampli piézo	7/8	99
Capacité variable intégrée	7/8	105
Circuit encodeur de clavier avec sortie sérielle	7/8	116
Clignotant alterné	7/8	125
Commutateur énergétique pour signaux analogiques	7/8	119
Démarrage progressif d'un régulateur dévolteur à découpage	7/8	72
Émetteur IR rustique	7/8	94
Générateur MLI à consigne 7 bits à GAL	7/8	114
Interrogation de touches avec peu de lignes	7/8	42
Interrupteur à thyristor	7/8	105
MIDI (de) aux convertisseurs N/A	01	70
Oscillateur à quartz pour fondamentale et harmonique	7/8	74
Petit récepteur IR	7/8	84
Régulateur à découpage abaisseur de tension	7/8	115
Sécurité thermique à Polyswitch	7/8	87
Serrure codée électronique	7/8	120

DOMESTIQUE

Commande de ventilateur à détecteur « FanSense »	7/8	117
Commutateur horaire : À 16 canaux piloté par μ C	05	14
Gare aux chiens !	7/8	81
Gradateur piloté par mC	01	42
Interrupteur activé par un claquement de mains	7/8	32
Interrupteur crépusculaire	7/8	56
LED blanches alimentées sous 1,5 V	7/8	69
Mémoire vocale intégrée	7/8	40
Moniteur pour fonction « Opentherm »	7/8	96
Polyswitch pour bobine	7/8	35
Serrure à code	7/8	12
Sonnette électronique sans problème de CEM	7/8	61
Système de protection de pompe	7/8	64

	Mois	Page
Télécommande RC5	7/8	48
Thermostat avec capteur à diodes pn déporté	7/8	31
Xéromètre : pour plantes d'intérieur	05	58

EXPÉRIMENTATION

Clignotant à LED et LDR	7/8	15
Coupure de la sortie d'un régulateur survolteur à découpage	7/8	27
Flash à LED	7/8	23
Gain et phasemètre	7/8	30
LED en photocapteur	7/8	61
Oscillateur 3 composants	7/8	28
Oscillateur à quartz commutable	7/8	75
Régulateur à découpage step-up à limitation de courant intégrée	7/8	29
Speedy Darling	7/8	36
Visualisation de flancs	7/8	73

JEUX, MODÉLISME & BRICOLAGE

Affichage modulaire à matrice de points	06	52
Démultiplexeur DMX 8 voies	12	08
ForTan : Foreuse tangentielle de platine commandée par PC (I)	03	38
ForTan (II)	04	28
ForTan (IIa)	05	28
ForTan (III) : Partie 3. Réalisation en 7 étapes	06	38
ForTan (IV) : mettons la main à la pâte	09	20
Guirlande	7/8	62
Interface DMX <-> MIDI	10	32
Interface I ² C pour servos	09	14
Jouvence pour accu	10	48
LED imprévisible	7/8	55
MIDI-Lights	03	70
QuizMaster en version monofilaire	09	28
Refroidisseur pour accu	7/8	16
Speed-Combi régulateur de vitesse modulaire pour modèle RC	01	14
Speed-Combi (II) régulateur de vitesse modulaire pour modélisme RC	02	50
Télécommande PCM miniature (I)	10	71
Télécommande PCM miniature (II)	11	14
Télécommande pour modélisme	04	20
Testeur de tension pour accu de modélisme	7/8	50
Timer pour chambre noire	05	20
Wobbel, le mini-robot « wobblant » !	10	16

MICROPROCESSEURS, MICRO-INFORMATIQUE

Adaptateur RS-232 isolé électriquement	7/8	26
Adaptateur pour connecteur Sub-D à 9 voies	7/8	58
Adaptateur simple pour la SB Live! Player 1024	7/8	86
Ampli Surround pour PC	11	46
CAN télécommandé à sortie RS-232	7/8	124
Calculateur diviseur de vitesse de bauds pour micros AVR	7/8	15
Capteurs de température à interface numérique à 1 fil	7/8	98
Carte à 32 entrées analogiques	06	26

	Mois	Page
Carte d'expérimentation PCI	02	16
Carte 89S8252 Flash	12	20
Commuteur clavier/souris	7/8	34
Compilateur graphique pour microcontrôleurs MCS-51	7/8	18
C.I. pilote de ventilateur avec sortie « température trop élevée »	7/8	65
DataSpy RS-232	10	8
DCI-PLC, simulation PLC par ordinateur	06	66
De PC à PC par 2 câbles d'imprimante	7/8	08
Émetteur-récepteur IR pour PC	12	54
GBPB - Carte de prototypage pour GameBoy	01	20
Graphiques (les) en Excel	7/8	107
HC12, microcontrôleur 16 bits	01	30
Horloge planétaire	7/8	79
Interface disque dur pour port imprimante	12	70
Interface disque dur pour mC 8 bits	03	16
Interface I2C série	11	76
Interrogation de touches avec peu de lignes	7/8	39
Inverseur de données RS-232 miniature	7/8	62
ISAC (1), contrôleur capteur/actionneur intelligent	10	40
ISAC-Cube	11	8
ISAC (3)	12	38
MCS BASIC-52 V1.3	02	45
Mini-serveur Web (I)	09	52
Mini-serveur Web (II)	10	52
Module graphique LCD sur μ P 8051	09	8
Moniteur BITBUS	7/8	47
Passage par zéro des microcontrôleurs	7/8	39
PC éteint, l'écran itou	7/8	74
Programmeur Atmel	09	38
Programmeur pour DS1621	02	40
Rendez-vous avec le BS2p de Parallax	09	60
Réseaux (les) neuraux aux commandes	06	22
Rien de plus qu'un nouveau programmeur d'EEPROM	7/8	38
SDCC (Small Device C-Compiler)	7/8	106
Simulateur d'EPROM	01	52
Télécommande pour PC (Récepteur de télécommande RC5)	06	62
UART pour USB	12	XX

VOITURE, MOTO & VÉLO

Chenillard dit « KITT-scanner »	7/8	20
---------------------------------	-----	----

INFORMATIONS

Accumulateurs Lithium-ion	09	32
Applikator :		
Étage de sortie 2 x 25 W pour automobile	12	60
NCP1200	10	20
Nouveaux circuits intégrés en oscillateurs	03	32
XICOR X9258T, 9250T	01	36
CAO (la) en Europe (I)	05	08
CAO (la) en Europe (II)	06	08
Circuits de base :		
Éclairage temporisé	02	66

	Mois	Page
Circuits fondamentaux :		
L'étage à relais	.04	56
Modulateur en largeur d'impulsion	.01	58
Coin du lecteur :		
Qualité d'enregistrement des CD	.02	60
Electronique sur PC - USB -		
Nouveaux microcontrôleurs	.04	63
Copie des CD-ROM sur disque dur - Transfos de Tesla		
PC-audio - Soucis d'E/S numériques - Télécommande RC		
Détecteur de métaux - Télécommande par GSM	.09	50
Terminal SIMCAD - Affichage modulaire à matrice de points - 2 amplis, 1 préampli - Acrobat et le CD-ROM de l'année Elektor 2000 - Mini-PIC-programmer	.11	36
Programmeur Atmel - Site EDiTS Pro - Réalisation de circuits imprimés - Thermomètre numérique à LED		
Booster pour EDiTS - Transfo ETD-29 - LS contre HC		
Brancher une radio portable au câble ?	.12	68
Comment fonctionne ? :		
Journal lumineux avec 7 LED seulement	.03	30
Commande en ligne sur Internet	.05	76
Concept (le) Phytex	.03	37
Cours d'initiation :		
L'électronique sur PC (V) :		
Mesure de valeurs analogiques	.01	66
L'électronique sur PC (VI) :		
Mesure par capteurs	.02	62
L'électronique sur PC :		
Mesures de tension (VIIème partie & fin)	.03	62
Description de kit :		
Commande à distance bifilaire	.01	40
Électronique en ligne :		
DMX-512	.12	80
Expériences à base de magnétron	.05	66
« Gonfler » son graveur CD	.10	82
Informations semi-conducteurs	.01	80
Lecteurs MP3 autonomes	.02	14
Projets MP3 à foison	.09	46
Montages en self-service	.04	70
Panorama des piles	.03	50
Pixels abîmés	.11	20
Surcadencement de CPU	.06	76
Elekture :		
Switch=Mode Power Supply SPICE Cookbook	.10	46
En savoir plus sur le bus PCI :		
Évolution et développement, d'ISA à PCI	.03	52
Horloges (les) radiopilotées modernes	.10	24
Infocartes :		
LTC490, pilote interface RS-485/422	.12	27
LTC1451/2/3, convertisseur A/N - N/A	.03	59
LT1074/76, régulateur à découpage abaisseur	.06	59
NCP 1200 - contrôleur MLI	.11	23
PBL3717A, commande de moteur pas à pas	.05	63
Régulateurs de la série NMF	.10	29
SG3525AN, régulateur de tension	.02	57
TDA2822, double amplificateur de puissance	.01	61
TDA8444	.09	69
TLC070 à 075 (A) Amplificateurs opérationnels	.04	61

	Mois	Page
Logiciels :		
CodeVisionAVR	.02	76
Multisim + Ultiboard	.04	14
OscilloSound™	.03	79
Logique floue (I) : Considérations de base	.04	24
Logique floue (II)	.05	68
Lorsque RS-485 rencontre CAN	.11	80
Mini-serveur Web (II)		
(suite de Télécommande à l'extrême)	.06	32
Modules de mémoire (les)	.11	38
Mystères (les) du DMX512	.05	52
Où en est la maison intelligente ?	.02	26
Outils de développement :		
ARM7TDMI Evaluation Kit	.03	78
CANDIP	.06	65
PIC Logicator et PIC Emulator	.01	82
PICSTART® Plus DDTP	.04	50
Prototype (du) à la série en temps record	.05	72
Rabbit 2000 Basic Development Kit	.10	78
TOPAS CE Starter Kit	.02	15
Voice Extreme™ Toolkit	.11	18
XE8000 ProStart	.07/08	128
78K - Test it!	.09	48
Principes de base :		
Expériences		
avec des céramiques piézo-électriques (I)	.04	16
Expériences		
avec des céramiques piézo-électriques (II)	.05	24
Portage simplifié de projets sous Windows CE	.03	66
Protocoles de télécommande par IR (I)	.03	24
Protocoles de télécommande par IR (II)	.04	36
Quartz et oscillateurs	.11	62
Récepteur GPS avec sortie NMEA	.12	48
Régulateurs de tension (la pratique des)	.12	66
Sus aux produits Microchip	.12	78
Télécommande à l'extrême (I) : via Internet et TCP/IP	.05	34
Télémétrie sans contacts	.10	60
UMTS (Universal Mobile Telephone Service)	.01	24
Vers des systèmes électroniques		
à basse puissance au quotidien	.02	70

TORT D' ELEKTOR

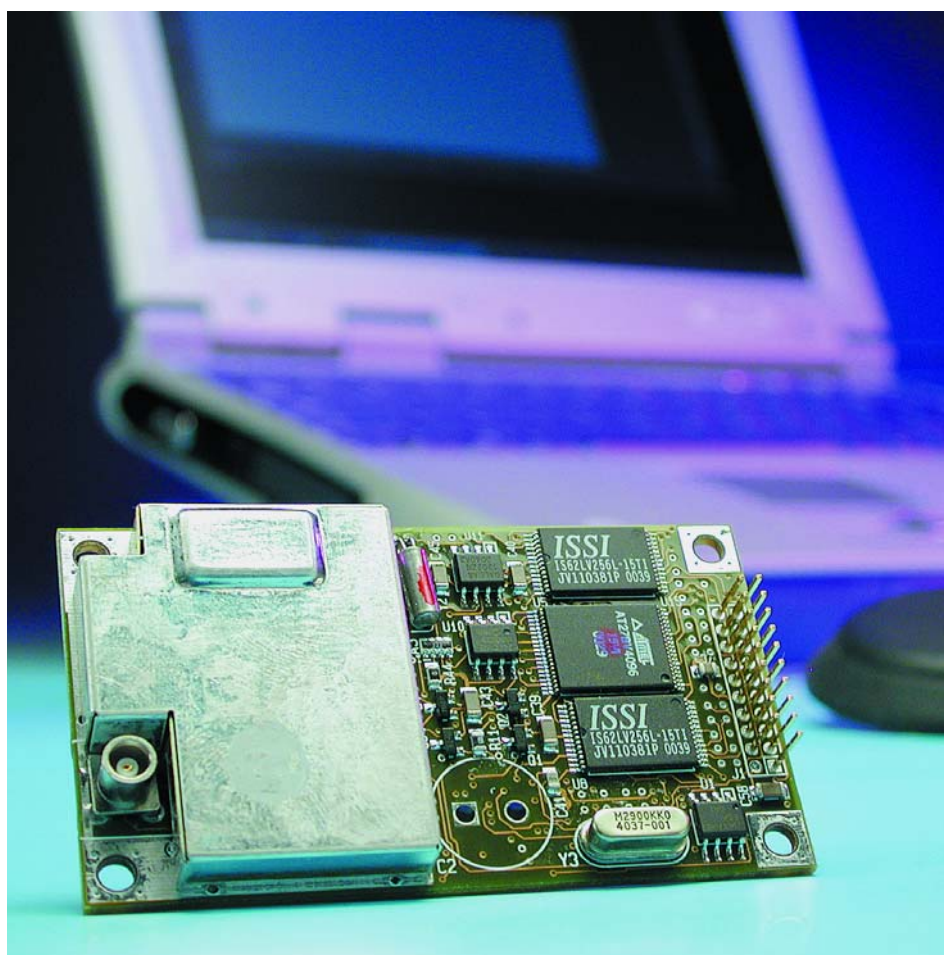
Ampli Hi-Fi P3	.09	80
Audion pour Ondes Courtes	.04	65
Commutateur horaire	.09	80
	.11	74
GBDSO GameBoy Digital Sampling Oscilloscope	.04	65
	.09	80
Indicateur de fonctionnement		
et de destruction de fusible	.11	74
Les mystères du DMX	.09	80
Mini-serveur Web (I)	.11	74
« Nonante », ampli 90 W à IGBTs	.04	65
Préamplificateur RIAA à ECL 86	.09	80
Récepteur M.A.	.09	80
Rien de plus qu'un nouveau programmeur d'EEPROM	.11	74
Simulateur d'EPROM	.04	65
Télécommande pour PC	.11	74
Timer pour chambre noire	.09	80

Récepteur GPS avec sortie NMEA

Principes de base et application

Gunther May, DL3ABQ

On trouve actuellement sur le marché des modules GPS à sortie NMEA à moins de 1 000FF. Il existe même des GPS de poche dotés de cette interface pour à peine plus cher. Le présent article a pour but de vous montrer (tout) ce que l'on peut faire à l'aide de ces appareils.



GPS (*Global Positioning System*) offre la possibilité à son utilisateur de déterminer sa position spatiale avec une précision plus qu'acceptable. Depuis la mise hors-fonction du système d'introduction d'erreur artificiel à l'intention des utilisateurs civils, la précision atteint entre 20 et 50 m, dans la pratique, l'erreur est même souvent inférieure à 10 mètres. Il est même possible, à l'aide du GPS différentiel (DGPS = *Differential GPS*), d'augmenter la précision par la prise en compte de données de correction. À l'image de nombreux appareils électroniques, les récepteurs GPS ont vu leur prix chuter sensiblement. Les modèles les plus simples sont actuellement disponibles pour de l'ordre de 1 000FF. Les appareils dotés de programmes cartographiques sophistiqués et au confort d'utilisation élaboré restent cependant d'un prix trop élevé. Il existe, si l'on ne tient pas impérativement à ce que la solution adoptée soit la plus compacte possible, une alternative abordable : acheter un module récepteur sans affichage mais doté d'une sortie données que l'on connectera à l'interface de sortie

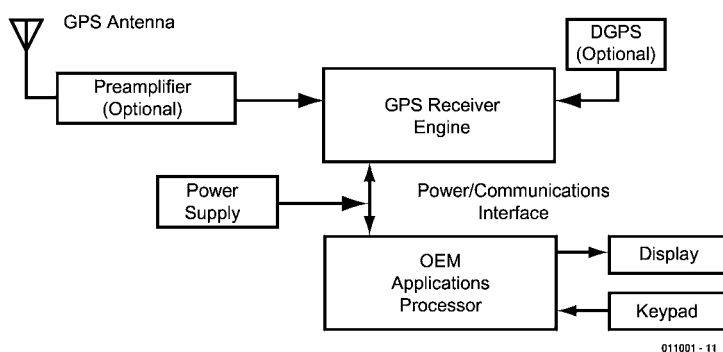


Figure 1. Synoptique d'un récepteur GPS complet.

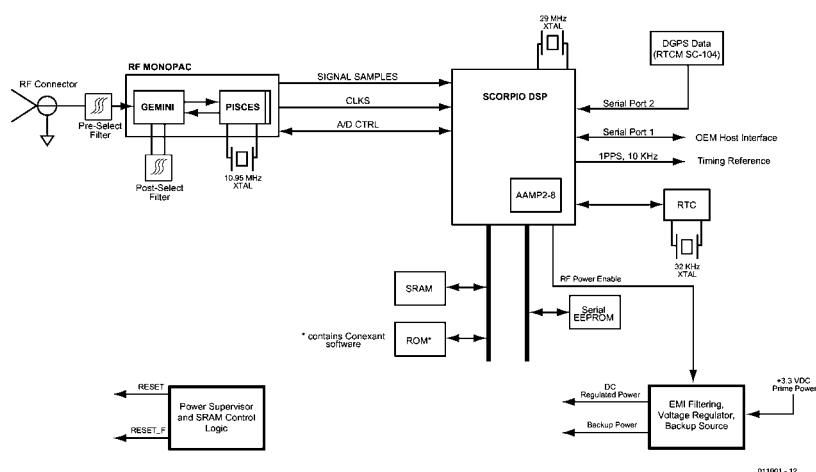


Figure 2. Synoptique du module de réception GPS.



Figure 3. Récepteur GPS haute de gamme.

d'un PC ou d'un ordinateur de poche (PDA = *Personal Digital Assistant*). Ce type de module existe pour moins de 1 000 FF, les programmes de cartographie pour PC que requiert cette approche ne sont pas trop coûteux eux non plus.

Récepteur GPS

Comme le montre le synoptique de la **figure 1**, les composants nécessaires à la réalisation d'un récepteur GPS se limitent à 2 ou 3 : une antenne GPS (qui pourra être, selon le modèle, soit intégrée soit séparée du reste de l'appareil), module souvent doté déjà d'un préamplificateur, le circuit du récepteur GPS proprement dit et une unité de commande. Cette dernière comporte dans la plupart des cas un microprocesseur pilotant un affichage et un clavier. On trouve également, en option, un récepteur DGPS chargé de recevoir, via les éthers, un signal de correction (données DGPS).

La **figure 2** vous propose un synoptique détaillé d'un récepteur GPS, schéma tiré de la fiche de caractéristiques du module de réception Jupiter LP de Conexant (ex-Rockwell). Le récepteur GPS proprement dit est constitué en fait d'un set de puces constitué de 2 circuits intégrés spécifiques au fabricant. Le bloc HF (RF-Monopac) sert à la réception du signal GPS et à son échantillonnage. Le reste du traitement, mise en forme et décodage du signal GPS ainsi que commande du bloc d'entrée (*front end*) est l'affaire d'un processeur numérique de signal (PNS plus connu sous la dénomination de DSP, *Digital Signal Processor*) spécialisé.

La complexité du matériel et du progiciel requis par la partie interface de saisie joue, dans le prix de revient d'un récepteur GPS complet, un rôle qui est loin d'être négligeable. Cela est plus particulièrement vrai en cas d'utilisation d'un écran TFT couleur de bonne taille, mais les programmes de cartographie et les banques de données intégrées grèvent eux aussi le budget de réalisation. On découvre, dans la photo de la **figure 3**, l'exemple type d'un récepteur GPS haut de gamme. Si l'on se contente d'acheter un simple récepteur et que l'on utilise un PC pour sa commande, la facture s'en voit sensiblement diminuée. Ce module « *GPS inside only* » pourra prendre la forme d'un module tel que celui de la **figure 4** ou encore de ce que l'on appelle une souris GPS (**figure 5**), module qui possède une antenne intégrée. Il existe aussi, pour à peine plus cher, des récepteurs GPS de poing bas de gamme, tel que, par exemple, le petit appareil de Garmin représenté en **figure 6**, qui, s'il ne comporte qu'une interface d'accès simple, n'en possède pas moins une interface pour données.

Il faut bien évidemment, pour pouvoir utiliser cette interface, que le programme tournant sur le PC et le récepteur puissent communiquer l'un avec l'autre par le biais d'un protocole standardisé.

NMEA

NMEA est un protocole de transmission développé pour l'échange de données entre différents instruments de mesure électroniques utilisés dans la marine (marchande). Comme le GPS a rapidement trouvé, dans le monde de la navigation maritime, un domaine de prédilection, NMEA est rapidement devenu le Standard du protocole de transmission entre récepteur GPS et périphérique en fin de ligne. Il n'est resté cependant pas moins un certain nombre de récepteurs qui connaissent leur propre protocole. Il faudra, sachant que la plupart des applications PC requièrent NMEA, ne pas opter pour un récepteur de cette catégorie ne répondant pas au standard.

Dans le présent article, lorsque nous parlons de NMEA, nous sous-entendons le protocole NMEA 0183 répandu dans ses versions 1.5 ou 2.0. On aura affaire, dans la majorité des cas, à la version 2.0, sachant que les récepteurs les plus récents n'utilisent pratiquement plus les versions plus anciennes (NMEA 0183 version 1.5, NMEA 0182 et NMEA 0180). La sortie NMEA du récepteur est connectée à l'une des interfaces sérieelles du PC. Le paramétrage RS-232 défini par le protocole est 4 800 bauds, 8 bits de donnée, 1 bit d'arrêt et sans parité. Il faudra se rappeler que certains récepteurs fournissent en sortie des niveaux TTL ou CMOS; ce type de récepteur requiert impérativement l'utilisation d'un convertisseur de niveau; l'approche la plus classique fait appel au circuit standard à base de MAX232 (figure 7).

Si le manuel mentionne une possibilité directe de connexion à un PC il n'y a aucune raison de se faire de souci à ce sujet.

Chaînes NMEA

Les informations NMEA sont subdivisées en lignes d'un maximum de 80 caractères, blocs de données baptisés chaînes (string). Il est possible d'examiner ces chaînes NMEA à



Figure 4. Exemple de module GPS.

Tableau I

Exemple de chaîne :

RMC,225446,A,4916.45,N,12311.12,W,000.5,054.7,191194,020.3,E*68

225446	Heure du point : 22:54:46 UTC (Universal Time Coordinated)
A	État du récepteur GPS (A: normal; V: Alarme)
4916.45,N	49° 16.45' Latitude Nord
12311.12,W	123° 11.12' Longitude Ouest
000.5	Vitesse
054.7	Route (cap)
191194	Date du point : 19 Novembre 1994
020.3,E	Variation magnétique 20,3° Est
*68	Somme de vérification



Figure 5. Une souris GPS intègre un récepteur GPS complet à antenne intégrée et pourra, selon sa version, être connectée directement à un port sériel ou USB de PC. La présente photo montre une NaviMouse (Unitronic) destinée à être connectée à un APN de l'écurie Palm.

l'aide d'un programme de terminal tel que HyperTerminal de Windows ou le Terminal de Norton-Commander et d'en lire le contenu sachant qu'il s'agit d'un code ASCII parfaitement lisible. Il existe cependant des alternatives au confort d'utilisation sensiblement meilleur auxquelles nous reviendrons un peu plus loin. Les chaînes débutent toujours par un « \$ », caractère suivi par un code d'identification du système (*system ID*) qui est, dans le cas des récepteurs GPS, tout simplement « GP ». Ces caractères sont suivis par les champs de données proprement dits. Le **tableau 1** propose, à titre

d'exemple, la chaîne RMS (= *Recommended Minimum Specific GPS/Transit data*) produite en fait par n'importe quel récepteur GPS. La FAQ (liste de questions/réponses) disponible à l'adresse Internet [1] donne une description complète du contenu d'une chaîne NMEA.

Logiciel pour PC

On a beau avoir le meilleur des récepteurs GPS, il ne sert pas à grand chose si le logiciel qui traite les données qu'il fournit ne le fait pas correctement. Il existe heureusement un nombre relativement

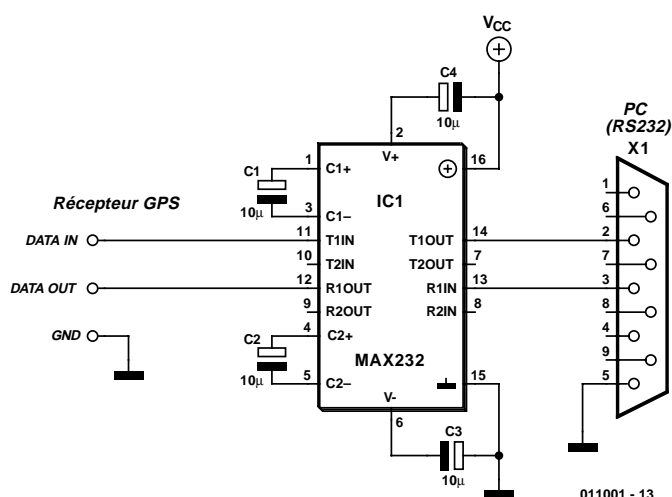


Figure 7. Nombre de récepteurs requièrent un convertisseur de niveau pour la connexion à l'interface RS-232. Le schéma standard à base de MAX232, circuit conçu spécialement à cet effet, remplit cette fonction à merveille.

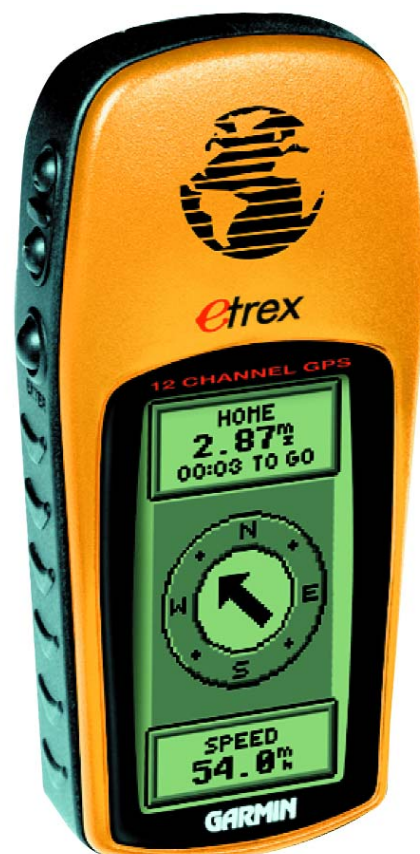


Figure 6. Les portables à GPS abordables disposent d'une interface NMEA.

important de programmes gratuits écrits à cet effet; nous vous proposons une description succincte de certains d'entre eux.

GPSS (cf. figure 8)

Ce programme freeware avec cartographie disponible à l'adresse www.gpss.co.uk/ est remarquable. Son environnement utilisateur requiert cependant une certaine période d'adaptation.

NMEA-Monitor

Autre programme freeware de visualisation de données NMEA. À télécharger à l'adresse :

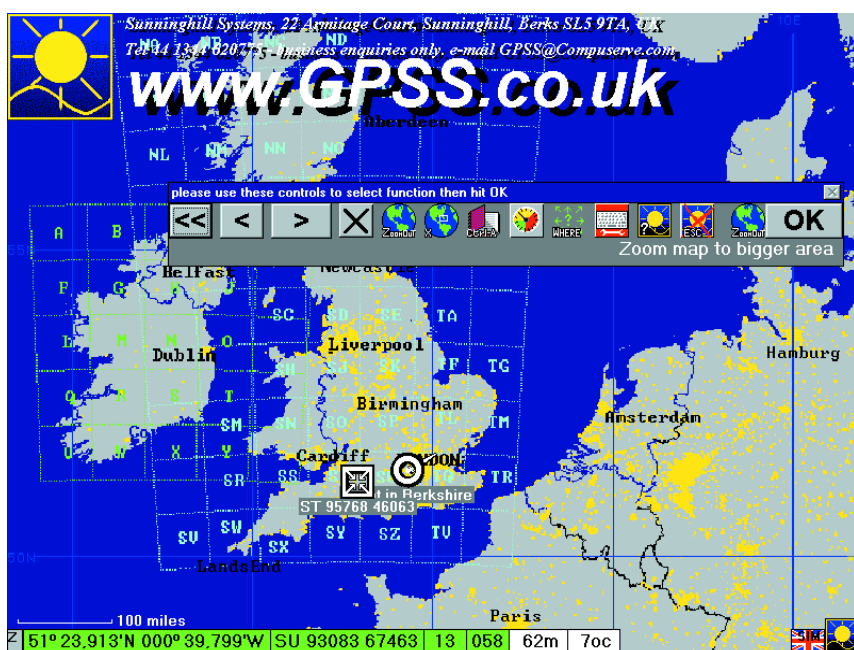
www.navtec.de/english/nmea_mon.htm

WinGPS

Un programme commercial très puissant avec cartographie dont il existe une version shareware. Son adresse de téléchargement :

www.stentec.com/

Figure 8. GPSS est un programme freeware à visualisation de cartes on ne peut plus intéressant. Mérite le détour.



Il ne s'agit là que d'un choix restreint sachant qu'il existe bien d'autres programmes de décodage de données NMEA. Notons qu'il existe également des programmes pour les ordinateurs de poing (PDA ou APN pour Assistant Personnel Numérique) tels que les Palms et autres Psions.

La plupart du temps, les programmes dotés d'une fonction de cartographie comprennent déjà un certain nombre de cartes géographiques; on pourra en importer d'autres depuis les CD-ROM prévus à cet effet ou de sites offrant des collections de cartes. Il est nécessaire, dans la plupart des cas, de calibrer ces cartes à l'aide de quelques points de repères connus.

APRS

Il s'agit ici d'un nouveau mode connu des radio-amateurs dans lequel on connecte un récepteur GPS à un émetteur/récepteur de trafic radio-amateur en utilisant le protocole NMEA évoqué plus haut. Il est possible ainsi d'échanger des informations de position entre différentes stations radio. Pour de plus amples informations on pourra faire un tour à l'adresse Internet :

www.dididahdidit.com/

Liens Internet

Il existe sur Internet nombre de sites traitant des thèmes GPS et NMEA. Donnons, en guise de conclusion à ce premier article, un florilège d'adresses :

- Pour en savoir plus sur le système GPS (TU Ilmenau) : www.emt.tu-ilmenau.de/emt/gps/www/gps/main.htm
- La page NMEA de Torsten Baumbachs avec projet logiciel : <http://pandora.inf.uni-jena.de/ttbb/nmea.html>
- Liste des modules GPS les plus abordables :

<http://rover.wiesbaden.netsurf.de/~hirschgps/gpsbord.htm>

- Site Internet d'information concernant le GPS de Joe Mehaffey et Jack Yeazel : <http://joe.mehaffey.com/>
- Le site GPS et NMEA de Peter Bennett : <http://zazu.optiva.ee/pub/nmea/>
- La « Global Positioning System (GPS) Resource Library » : www.gpsv.com/gpsinfo/
- La FAQ du Newsgroup maus.technik.gps : <http://home.wtal.de/noegs/mtg-faq.htm>

(011001-1)

Sources :

- [1] Bennett, Peter (2000): The NMEA FAQ Version 6.3 (<http://vancouver-webpages.com/peter/nmeafaq.txt>)
- [2] Schrödter, Frank (1994): GPS Satelliten-Kommunikation en allemand); Franzis-Verlag, Poing
- [3] Récepteur GPS (figures 3 et 6) : www.garmin.de
- [4] Récepteur GPS (figures 4 et 5) : www.icc-unitronic.de
- [5] Synoptiques (figures 1 et 2) : www.conexant.com

Émetteur/récepteur IR pour PC

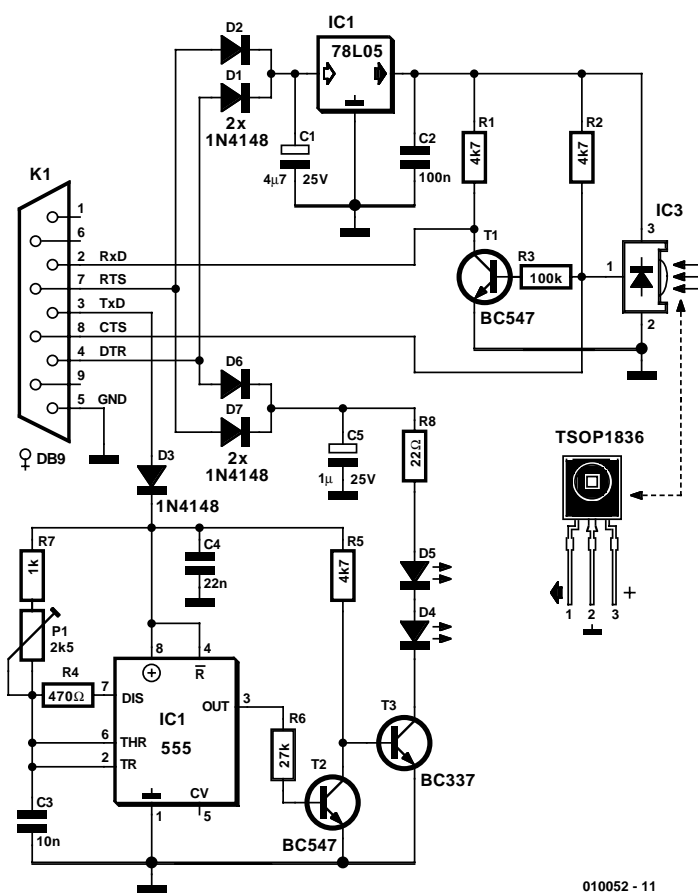
Télécommande d'appareils et liaison de données en un

Projet : Burkhard Kainka

Le présent article ne se contente pas de décrire un projet d'émetteur/récepteur IR à l'aide duquel il sera possible, à partir d'un PC, de télécommander et de transférer des données. Il constitue une suite à la série d'articles intitulée « L'électronique sur PC » et donnera également des informations additionnelles sur le transfert par le biais des IR et les interfaces utilisées, de sorte que vous pourrez utiliser ces éléments lors de la conception de projets propres.



La transmission de données par infrarouge (IR) ne cesse de voir sa part de marché croître. Si, il n'y a encore que peu de temps, les télécommandes étaient les seuls ustensiles à travailler avec cette lumière invisible (pour l'oeil humain) de plus en plus de périphériques allant de la souris aux claviers pour PC en passant par les imprimantes font appel à l'infrarouge pour communiquer. Aujourd'hui, on utilise même la lumière IR dans les faisceaux de câbles à fibre optique. Le montage proposé ici, notre « émetteur/récepteur IR pour PC » constitue, de par son aspect pratique puisqu'il fait appel à l'utilisation d'une interface de PC et son logiciel à Visual BASIC, en



010052 - 11

Figure 1. L'électronique de l'émetteur/récepteur IR à connecter à l'interface série d'un PC.

quelque sorte une suite à la série d'articles « L'électronique sur PC » (Elektor n° 267 à 273, septembre 2000 à mars 2001) et du livre « Petites expériences d'électronique avec mon PC » (avec CD-ROM) paru tout récemment.

Nombre de télécommandes IR pour téléviseurs, magnétoscopes et autres équipements grand public fonctionnent selon le standard RC5 défini par Philips. Les programmes donnés plus loin permettent de s'assurer facilement si une télécommande donnée utilise ou non le code RC5. Ce standard utilise des signaux modulés dont la fréquence se situe entre 30 et 40 kHz. La télécommande émet ce que l'on appelle des trains d'impulsions (*burst*) qui sont en fait des paquets d'impulsions d'une longueur de 0,888 ou 1,776 ms.

Dans le cas d'une fréquence de modulation de 36 kHz, un train court comporte 32 impulsions distinctes, un train long 2 fois plus, 64 partant. La durée de l'ensemble du paquet

est de 25 ms environ; sa répétition se fait toutes les 100 ms tant que se prolonge l'action sur l'une ou l'autre touche.

Il est possible, très facilement, d'utiliser une télécommande infrarouge à d'autres fins. On peut ainsi s'en servir pour piloter certaines fonctions d'un programme de son propre cru. L'une des applications typiques pourrait être le pilotage d'un diaporama. Une fois le programme démarré, il ne reste plus qu'à s'enfoncer dans son fauteuil et à effectuer la régie du spectacle à partir de là.

Émetteur/récepteur : le matériel

La réception de signaux infrarouge courants est devenue, grâce à la mise sur le marché de récepteurs intégrés, une opération relativement simple. Le SFH506, un récepteur IR très connu de Siemens existe pour différentes fréquences de modula-

Émetteur/récepteur IR

Caractéristiques techniques

- Fréquence de réception : 30, 33, 36 kHz en fonction des composants mis en place
- Fréquence d'émission : 30 à 40 kHz ajustable continûment
- Alimentation : par le biais de l'interface série
- Portées : 10 m environ
- Récepteur IR : pour télécommandes au standard RC5
- Émetteur IR : compatible RC5
- Émetteur/récepteur de données IR : données sérielles, max 2 400 bauds.

tion, 30, 33, 36 kHz et ainsi de suite. La courbe du filtre possède une bande relativement large de sorte qu'une dérive de quelques kilohertz n'entraîne qu'une détérioration relativement faible de la sensibilité.

Le TSOP1836 de Vishai/Telefunken est un circuit intégré de fonction similaire. Ces 2 composants requièrent une tension d'alimentation de 5 V seulement et se contentent d'une consommation de moins de 2 mA. Il est par conséquent possible de les alimenter directement depuis l'interface série du PC.

L'émetteur/récepteur IR proposé ici comporte, outre le récepteur, on s'en serait douté, également un émetteur IR modulé pour des fréquences de porteuse auxiliaire comprises entre 30 et 40 kHz. L'émetteur IR pourra être utilisé pour la télécommande d'appareils tel que magnétoscopes ou téléviseurs. Mais il est également possible de s'en servir pour la transmission de données par voie série entre 2 PC.

Un coup d'oeil sur le schéma de la **figure 1** nous permet de découvrir les 3 picots auxquels vient se connecter le circuit intégré du récepteur, IC3, et un régulateur de tension intégré, IC1, un 78L05. La tension d'alimentation est dérivée des sorties DTR (*DaTa Ready*) et RTS (*Ready To Send*) de l'interface RS-232, lignes interconnectées par le biais des diodes D1 et D2 en vue de leur découplage. Il est possible, en faisant tourner un programme sur le PC, de disposer à ce niveau d'une tension de l'ordre de 10 V. Ces lignes alimentent en outre, au travers des diodes D6 et D7, la partie émission du montage. Il nous a fallu prévoir un condensateur de capacité relativement importante, 4,7 μ F pour C1, en raison du courant impulsionnel relativement élevé requis lors de l'émission d'une part, et pour le lissage de la tension d'entrée du régu-

lateur de tension de l'autre.

Le récepteur IR intégré prévu pour des signaux IR modulés à 36 kHz produit à son contact central, un signal de sortie actif au niveau bas. Ce signal est appliqué directement à la ligne CTS d'où il subira un décodage logiciel. La présence d'une résistance de forçage au niveau haut (*pull-up*) additionnelle se justifie en raison de la résistance d'entrée relativement faible de la ligne CTS. Le signal subit en outre une inversion introduite par un étage à transistor, T1, avant d'être appliqué à l'entrée Rx D de l'interface série. Cette ligne sert à la réception de données série, lors de l'établissement d'une liaison de données entre 2 PC par exemple.

L'émetteur infrarouge comporte un étage de modulation constitué par IC2 et un amplificateur d'impulsions prenant la forme d'une paire de transistors, T2 et T3, le tout attaquant 2 LED IR, D4 et D5. IC2 est un temporisateur du type 555 monté en oscillateur chargé de fournir des impulsions en aiguille négatives d'une durée de quelque 2 μ s. L'ajustable P1 permet d'ajuster la fréquence entre 30 et 40 kHz. On peut ainsi, en fonction de l'application, ajuster le circuit à la fréquence requise et de ce fait obtenir la portée maximum. Le 555 tire son alimentation de la ligne Tx D qui active et désactive l'émetteur le modulant ainsi (on parle d'« échantillonnage »).

L'énergie requise par les 2 diodes IR d'émission est elle aussi dérivée de l'interface série. Les lignes DTR et RTS chargent un condensateur de 1 μ F, C5, par le biais des diodes D6 et D7. Les brèves impulsions de sortie disponibles à la sortie (broche 3) du temporisateur rendent passant l'étage de comme des LED que constituent T2 et T3. Ce processus se traduit par la naissance de courants impulsionnels d'une intensité comprise entre 200 et 300 mA. La portée est, dans ces conditions, de l'ordre de 10 mètres. Bien que le courant de charge maximum que puisse fournir l'interface série soit, avec ses 40 mA, relativement faible, il n'en est pas moins, en raison de l'intervalle séparant 2 impulsions, largement suffisant pour recharger à chaque fois le condensateur électrochimique de charge.

Nous avons dessiné une platine en vue de vous faciliter la réalisation de ce montage. Elle vous est proposée, recto-verso, en **figure 2**. Le seul point auquel il faille veiller lors de la mise en place des composants est le respect de la polarité de toutes les diodes et condensateurs électrochimiques, et de l'orientation des circuits intégrés. L'une des erreurs les plus souvent commises est l'utilisation, pour l'embase Sub-D en équerre à

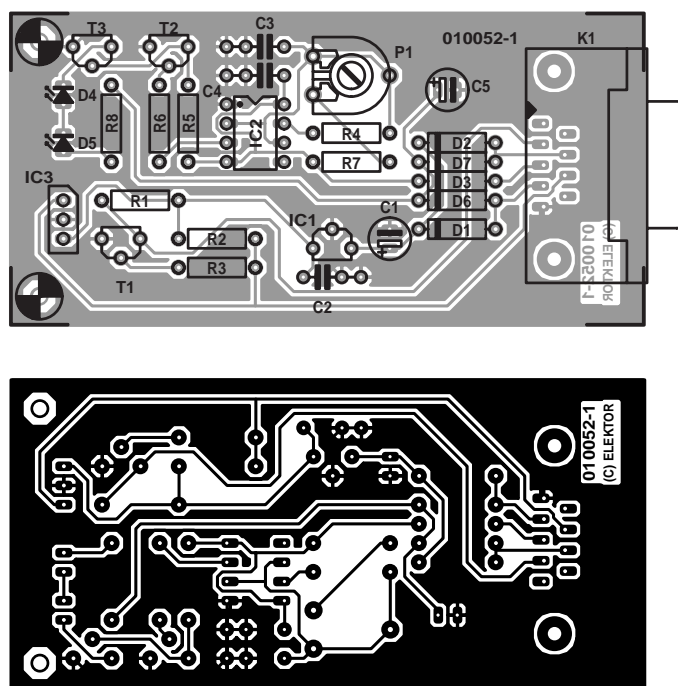


Figure 2. Dessin des pistes et sérigraphie de l'émetteur/récepteur IRI.

Liste des composants

Résistances :

R1, R2, R5 = 4k Ω
R3 = 100 k Ω
R4 = 470 Ω
R6 = 27 k Ω
R7 = 1 k Ω
R8 = 22 Ω
P1 = ajustable 2k Ω 5

Condensateurs :

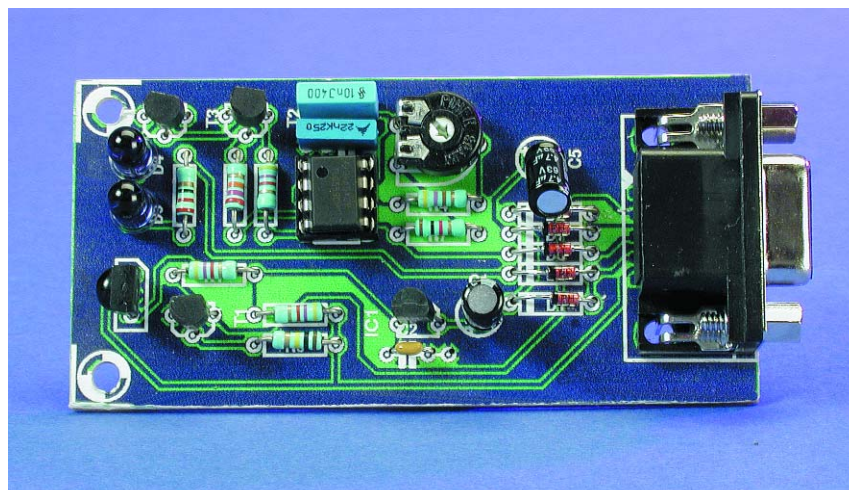
C1 = 4 μ F/25 V vertical
C2 = 100 nF
C3 = 10 nF
C4 = 22 nF
C5 = 1 μ F/25 V vertical

Semi-conducteurs :

D1 à D3, D6, D7 = 1N4148
D4, D5 = LED IR telle que, par exemple, LD271
T1, T2 = BC547
T3 = BC337
IC1 = 78L05
IC2 = 555
IC3 = TSOP1836 (Temic), SFH506-36

Divers :

K1 = embase femelle Sub-D à 9 contacts femelle en équerre encartable



Listage 1

Réception et visualisation de données RC5

```

Dim Ctr
Dim Adr
Dim Dat

Sub RC5Error()
    Ctr = -1
    Adr = -1
    Dat = -1
End Sub

Function RC5Bit() As Integer
    TIMEINITUS
    If CTS = 0 Then
        While ((TIMEREADUS < 500) And (CTS = 0))
            Wend
            If TIMEREADUS > 499 Then RC5Error
            DELAYUS 444
            If CTS = 0 Then RC5Error
            RC5Bit = 0
        Else
            While ((TIMEREADUS < 500) And (CTS = 1))
                Wend
                If TIMEREADUS > 499 Then RC5Error
                DELAYUS 444
                If CTS = 1 Then RC5Error
                RC5Bit = 1
            End If
        End Function

Sub RC5()
    Ctr = 0
    Adr = 0
    Dat = 0
    Startbit = False
    REALTIME True
    TIMEINIT
    While Not Startbit
        While (CTS = 1) And (TIMEREAD < 500)
            Wend
            Startbit = True
            DELAYUS 444
            If CTS = 1 Then Startbit = False
            DELAYUS 888
            If CTS = 0 Then Startbit = False
            DELAYUS 888
            If CTS = 1 Then Startbit = False
            Adr = Startbit
            If TIMEREAD > 499 Then Startbit = True
        Wend
        DELAYUS 888
        Ctr = RC5Bit
        Adr = 0
        For N = 1 To 5
            DELAYUS 888
            Adr = Adr * 2
            Adr = Adr + RC5Bit
        Next N
        Dat = 0
        For N = 1 To 6
            DELAYUS 888
            Dat = Dat * 2
            Dat = Dat + RC5Bit
        Next N
        REALTIME (False)
    End Sub

Private Sub Form_Load()
    OPENCOM "COM2"
    DTR 1
    RTS 1
End Sub

Private Sub Form_Unload(Cancel As Integer)
    CLOSECOM
End Sub

Private Sub Timer1_Timer()
    RC5
    Text1.Text = Str$(Ctr) + " " + Str$(Adr) + " " + Str$(Dat)
End Sub

```

9 voies K1, d'une embase mâle (9 contacts) au lieu de l'embase femelle (9 orifices) qu'il faut en fait utiliser.

Décodeur RC5 : le logiciel

La sortie de données du récepteur intégré est reliée directement à la ligne CTS. On dispose à ce niveau

des signaux produits par une télécommande dirigée vers le récepteur. Il nous faut maintenant un programme de décodage du signal qui traite les impulsions entrant de manière

Listage 2

Chargement d'images pour un diaporama

```

Private Sub Timer1_Timer()
    RC5
    If Dat <> Dat_old Then
        If Dat = 1 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast11.jpg")
        If Dat = 2 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast21.jpg")
        If Dat = 3 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast31.jpg")
        If Dat = 4 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast41.jpg")
        If Dat = 5 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast51.jpg")
        If Dat = 6 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast61.jpg")
        If Dat = 7 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast71.jpg")
        If Dat = 8 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast81.jpg")
        If Dat = 9 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast91.jpg")
    End If
    Dat_old = Dat
End Sub

```


à identifier la touche ayant été actionnée sur la télécommande.

La **figure 3** montre le signal RC5 entrant, signal émis par une télécommande RC5. Notons que le chronodiagramme a été saisi par le biais d'un analyseur logique que décrit l'ouvrage cité en début d'article.

Un programme visualise immédiatement tout changement du signal sur l'entrée CTS. Voyons ce signal d'un peu plus près. Le protocole RC5 utilise ce que l'on appelle un signal biphase. L'information proprement dite se trouve dans le changement de phase. Le niveau de signal change au plus tard après 1,776 ms. Le récepteur peut ainsi, grâce à ce changement de niveau, se resynchroniser en permanence sur le signal.

Le signal démarre toujours avec la même séquence initiale. Celle-ci est suivie de 3 trains de données, dans lesquels les changements de niveau à des intervalles de 1,776 ms constituent en fait les vrais bits de donnée. Après chacun de ces changements de niveau, le récepteur attend un peu plus de 0,888 ms avant de prendre en compte le niveau suivant, qu'il ait ou non changé. Le changement de niveau qui le suit est tout à la fois un point de synchronisation et une information de bit. Il est possible, avec ce principe, de transmettre des mots de donnée de longueur quelconque. Dans le cas du signal RC5, ces mots de donnée ont une longueur de 12 bits :

- * Le bit de commande (*Ctl*) change d'état lors de chaque action sur une touche, basculant entre 0 et 1 et inversement. Ceci permet au récepteur de faire la différence entre une action prolongée sur une touche et une action répétée sur cette touche.

- * L'adresse de l'appareil (*Adr*) comporte 5 bits, la transmission débutant par les bits de poids fort. Les adresses d'équipement

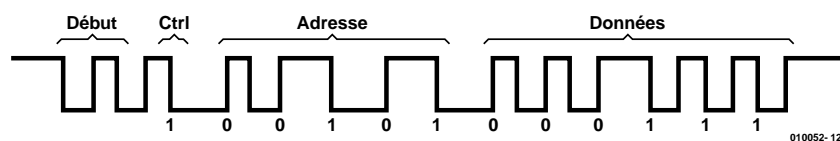


Figure 3. Exemple de signal RC5 correspondant à l'adresse 5 et à la touche numérique 7.

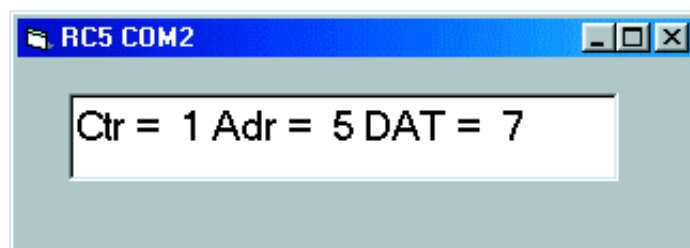


Figure 4. Visualisation de données RC5.

courantes sont 1 pour les téléviseurs et 5 pour les magnétoscopes. Cela permet d'utiliser plusieurs télécommandes dans une même pièce.

- * Le train de données (*Dat*) comporte 6 bits, ce qui permet de différencier un maximum de 64 touches différentes. Les touches numériques 0 à 9 produisent les codes correspondants, 0 à 9. Ici encore la transmission commence par les bits de poids fort.

Le **listage 1** donne le programme du décodeur logiciel écrit en Visual BASIC. La procédure RC5 reçoit les données. On utilise dans ce cadre le fichier PORT.DLL du livre « Je pilote les interfaces de mon PC » (téléchargeable gratuitement depuis le site Internet d'Elektor, adresse www.elektor.presse.fr, cf. la note Téléchargement en fin d'article) sert à tous les accès à l'interface série et assure le respect de la chronologie.

Cette tâche possède une chronométrie relativement critique et requiert la mise en oeuvre de *REALTIME True*. La procédure commence par attendre un niveau logique bas faisant office d'impulsion de début. Pour éviter qu'en l'absence de signal le PC ne se plante dans une boucle d'attente, le programme a été doté d'une condition de *Timeout*. Si, au bout de 500 ms, il n'y a toujours pas

de réception de signal, le programme s'arrête et génère un message d'erreur.

Les pilotages par infrarouge peuvent être gênés par la présence d'autres sources de lumière. Les sources de parasites sont les ampoules fluorescentes qui produisent des parasites lumineux impulsifs. La routine de réception RC5 vérifie pour cette raison la séquence de début du signal reçu pour s'assurer de son déroulement correct. Si le signal ne correspond pas au train d'impulsions attendu, il doit s'agir d'un parasite. Ce processus permet la réception correcte de signaux RC5 même dans un environnement relativement parasité.

Après la séquence de début, chacun des bits est traité dans la procédure RC5bit. En cas de lecture, au début de la procédure, d'un niveau 0, il doit s'agir d'un bit 0. On attend ensuite le changement de niveau. Au bout d'une demi-longueur d'impulsion, 444 μ s, on procède à une interrogation pour voir si le niveau présent est encore le même. Si cela n'est pas le cas, il y a eu une erreur. Toutes les informations prises en compte jusqu'à cet instant sont écrasées par une valeur -1. La procédure RC5bit se manifeste, après un peu plus de 444 μ s, avec le résultat du bit lu. La procédure RC5 mise à contribution



Figure 5. Diaporama sur l'écran d'un PC.

Littérature :

Burkhard Kainka: « Petites expériences d'électronique avec mon PC, Publitronic (ISBN 2-86661-126-8)

Burkhard Kainka: « Je programme les interfaces de mon PC sous Windows » Publitronic (ISBN 2-86661-115-2)

traite le bit et attend à nouveau 444 μ s jusqu'au nouvel appel de la procédure RC5bit.

Ce faisant, on ne tient pas compte d'un éventuel changement de niveau suivant. C'est d'ailleurs le but de la manoeuvre sachant que ce n'est que 1,776 ms après le dernier changement de niveau pris en compte que l'on attend le nouveau changement de niveau valide.

La réception des caractères est pilotée par un temporisateur (*timer*) du programme en VB. Toutes les informations reçues sont visualisées dans une fenêtre de texte. La **figure 4** montre la recopie d'écran des informations reçues en provenance d'une télécommande d'un appareil se trouvant à l'adresse d'équipement 5, un magnétoscope. La touche ayant été actionnée est la touche numérique 7. En l'absence de réception de signal, on constate, avec $Ctr = -1$ et $Adr = -1$, l'émission d'un message d'erreur. Dans le cas d'une véritable application cette information d'erreur ne serait pas visualisée mais utilisée pour faire la distinction entre les informations de touches utilisables et celles que ne le sont pas.

On pourra utiliser ce programme pour se familiariser avec les différents codes de la télécommande dont on dispose. Cela pourra servir de base pour des systèmes de commande à venir avec lesquels le PC se substituera à une télécommande. On

pourra également trouver sur Internet une liste complète des codes courants.

Télécommande de PC

Autre application intéressante, la télécommande du PC. On pourrait ainsi, par exemple, piloter chez soi un diaporama par le biais d'une télécommande. Le **listage 2** montre un extrait du programme de la procédure *Timer* modifiée. On utilise ici 9 codes de touches pour charger autant d'images stockées sur le disque dur.

Le fait qu'il ne s'agisse dans le cas présent que de 9 images pourra réjouir nombre de victimes ayant eu à subir l'une ou l'autre soirée diapos. Mais rien n'empêche une éventuelle extension du programme. On pourrait bien entendu imaginer d'utiliser, par exemple, les touches « + » et « - » pour la commande.

(100052-1)

Dans le prochain numéro nous vous proposerons le programme pour un émetteur RC5 avec le PC et la platine de l'émetteur/récepteur IR. L'ensemble permettra de piloter le magnétoscope et d'autres équipements depuis le PC ou encore transférer des données sérieelles entre 2 PC par le biais d'une liaison infrarouge.

Note : Téléchargement

Le fichier PORT.DLL mentionné dans cet article ainsi d'ailleurs que les listages des programmes et les dessins de platine sont mis à votre disposition sur le site Internet d'Elektor (www.elektor.presse.fr) pour un téléchargement gratuit. Vous pourrez y accéder par le biais du point de menu « Téléchargements » présent sur la gauche de la page d'accueil, pour ensuite choisir le numéro du magazine (mois) dans lequel l'article a été publié. Notons que les fichiers de programmes sont également disponibles sur le site de l'auteur : <http://home.t-online.de/home/B.Kainka>.

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.

Étage de sortie 2 x 25 W pour automobile

Dopé au TDA1563Q

Les fabricants de semi-conducteurs ne sont jamais à court d'idées et d'astuces pour utiliser efficacement l'énergie. Les étages finals des véhicules automobiles ne constituent pas une exception.

Les amplificateurs de puissance pour la reproduction audio dans les véhicules automobiles basés sur un étage de sortie de classe B classique atteignent une puissance de sortie dépassant légèrement 6 watts. Malgré leur faible puissance de sortie, ces étages finals sont utilisés dans la plupart des autoradios ; en voici la raison : Les étages finals de classe B atteignent le très bon rendement de 78,5 % (pour un sinus à la puissance maximale), la puissance dissipée n'atteint que de 20 % de la puissance de sortie. Cela permet d'éviter un échauffement excessif dans le véhicule ou de faire appel à un refroidisseur démesuré. La température ambiante que doit supporter l'amplificateur dans un véhicule automobile peut devenir en effet considérablement plus élevée que celle du salon, de sorte que l'autoradio doit chauffer le moins possible.

Rien à redire jusque là. Mais il est impossible de vendre actuellement une autoradio qui ne brille que par son manque de puissance de sortie (2 x 6 watts). Mais les fabricants d'appareils audio pour véhicules automobiles ne se sont pas tenus pour battus. Il importe tout d'abord de définir une puissance de sortie qui n'ait rien à voir avec la réalité (sinus de 100 Hz, tension de fonctionnement de 18 V, 10 % de THD). Une puissance apparente de 30 watts retombe en effet dans des conditions réalistes (signal audio, $U_b = 13,2$ V, THD = 0,5 %) à entre 7 et 8 watts. C'est certes inadmissible mais autorisé par la norme japonaise EIAJ.

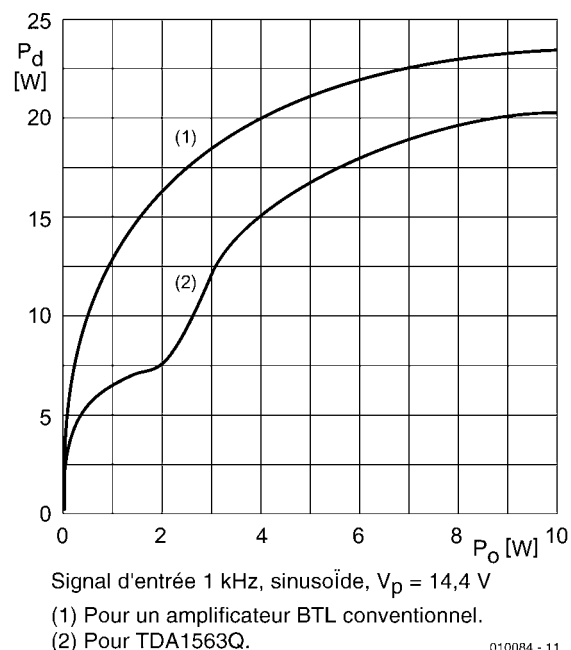


Figure 1. Comparaison d'un amplificateur en pont conventionnel et d'un TDA1563Q : la puissance dissipée par le nouveau composant de Philips est indubitablement moindre.

Une seconde possibilité (plus honnête) consiste à se servir d'une alimentation à découpage qui élèvera la tension de fonctionnement à (presque) n'importe quelle valeur. Cela permet de faire tonitruer le haut-parleur sans aucun problème.

Mais une alimentation à découpage (habituellement installée dans le boîtier contenant les étages finals) est encombrante et nécessite un refroidisseur de taille considérable qui, outre les pertes des transistors finals, doit aussi éliminer l'excédent

de chaleur de l'alimentation elle-même. Le bâti normé d'une radio n'offre en aucun cas la place nécessaire pour un engin de ce genre.

Une autre façon d'augmenter la puissance est offerte par le fonctionnement en pont. On a besoin de 2 étages finals dont l'un fonctionne comme d'habitude tandis que l'autre est excité par le signal B.F. inversé (au moyen d'un étage inverseur de phase). Le haut-parleur est commuté entre les deux sorties de l'amplificateur de sorte que l'amplitude de la tension est doublée par rapport au mode unipolaire (ou asymétrique = *single-ended*) et la puissance est (théoriquement) quadruplée. Mais, hélas, la puissance dissipée est aussi quadruplée. Cela signifie : un grand radiateur, un échauffement notable et un gaspillage de la capacité de la batterie.

Mais les fabricants d'électronique pour véhicules automobiles déploient de grands efforts pour prendre en défaut les lois de la physique en faisant appel à des recettes plus ou moins élaborées. On part du fait que la puissance normale de sortie d'un étage final de classe B efficace suffit parfaitement pour assurer une reproduction audio correcte, hormis lors de pointes de courte durée. Il suffit donc de fournir l'énergie suffisante pendant ce court laps de temps pour atteindre une puissance de sortie plus élevée.

Dans le numéro 201 d'Elektor (mars 1995, étage de puissance « auto »), nous vous avons fait part de l'existence d'un circuit intégré, le TDA1560Q, équipé d'une sorte de pompe de charge intégrée qui multiplie presque par deux la tension de fonctionnement lorsque des pics transitoires apparaissent. Cela permet d'obtenir théoriquement une puissance de sortie de 26 watts (mais en pratique cette valeur est considérablement moins élevée). Quelque génial que semble être de prime abord ce procédé, un coup d'œil aux valeurs de distorsion nous ramène à la réalité. Lors de la commutation de la tension de fonctionnement, l'étage final engendre des distorsions considérables, distinctement audibles, ce qui est très désagréable lorsque le niveau sonore est élevé et que les commutations se font d'autant plus fréquentes.

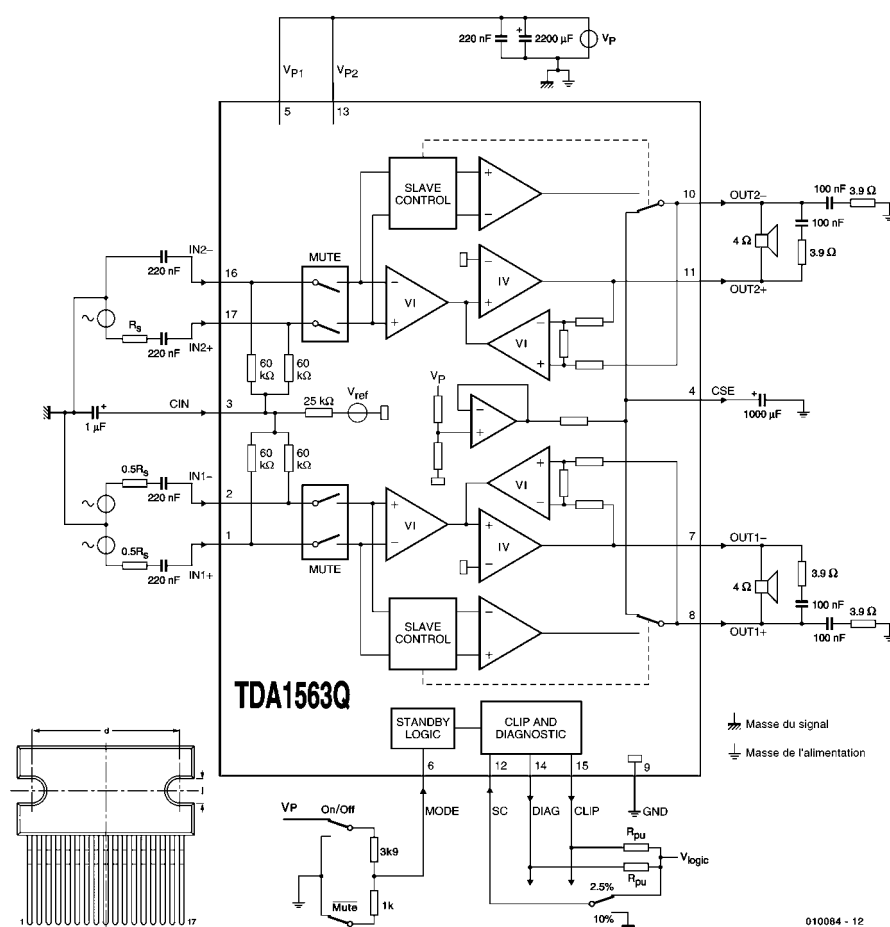


Figure 2. Circuit interne et exemple d'application du TDA1563Q.

Moins d'énergie, un radiateur plus petit

Philips vient de mettre au point un nouveau concept incarné par le TDA1563Q. L'étage final fonctionne autant que possible en classe B, donc à la moitié de la tension de fonctionnement en régime sinusoïdal. Le rendement est particulièrement élevé dans cette région. Si la valeur réelle de la tension de sortie dépasse 3 V, le circuit intégré commute en mode pont. Ce n'est certes pas aussi efficace que le fonctionnement en classe B, mais la puissance de sortie obtenue (avec une tension de fonctionnement réaliste et une distorsion supportable) peut atteindre 20 watts par canal.

Le rapport entre la puissance de sortie et la puissance dissipée est reproduit dans la **figure 1** pour un signal d'entrée sinusoïdal de 1 kHz ; la partie supérieure se rapporte à un

amplificateur en pont conventionnel, la partie inférieure au TDA1563Q. La « cassure » causée par le point de commutation apparaît clairement sur la courbe. La puissance dissipée en mode unipolaire n'atteint pas la moitié de celle d'un amplificateur en pont usuel. Si l'on suppose que l'amplitude de sortie d'un signal musical suit une distribution normale (gaussienne), la puissance dissipée d'un amplificateur en pont usuel est de 70 % supérieure.

L'amélioration du rendement présente toutefois d'autres conséquences que des économies d'énergie. Un amplificateur en pont conventionnel avec une résistance thermique totale (dissipateur thermique de la couche d'arrêt) de 1,5 K/W, une température de la puce permise de 145 °C pour une température ambiante maximale de 65 °C et une puissance de sortie prélevée de 6,5 watts par canal a besoin (dans le cas d'un signal musical) d'un dissipateur thermique de

$$R_K = (145 - 65) / 2 \cdot 6,5 - 1,5 \approx 4,6 \text{ K/W}$$

Comme le fonctionnement du TDA1563Q est

d'environ 70 % plus efficace, on obtient dans ce cas

$$R_K = 1,7 \cdot [(145-65)/2 \cdot 6,5] - 1,5 \approx 9,0 \text{ K/W}$$

Le radiateur peut être sensiblement (!) plus petit.

La **figure 2** représente le circuit interne et une application standard du TDA1563. La source du signal –symétrique ou par rapport à la masse– est connectée à IN+/- . On peut voir en haut comment raccorder une source de signal asymétrique. L'entrée avec inversion est mise à la masse par le condensateur. Le câblage de l'autre canal est prévu pour une source de signal symétrique. Les résistances indiquées par R_S symbolisent simplement la résistance interne de la source du signal. Comme cette résistance interne forme un filtre passe-haut avec le condensateur, R_S ne sert qu'au calcul de la fréquence limite inférieure.

Le signal B.F. est envoyé à 2 amplificateurs qui engendrent des signaux inverses l'un par rapport à l'autre. Une des connexion du haut-parleur est couplée de façon permanente à un étage final, l'autre est commuté par le « *Block Slave Control* » entre le deuxième étage final (pour fonctionnement en pont) et la liaison CSE (*single-ended*). La liaison CSE est maintenue à la moitié de la tension de fonctionnement pour laquelle un condensateur électrolytique de 1 000 μF fait office de tampon. Un amplificateur opérationnel supplémentaire dont les entrées sont reliées aux sorties du haut-parleur empêche la présence d'une tension offset continue aux sorties du haut-parleur.

Sécurité et diagnostic

Et voilà pour le traitement du signal. Mais le TDA1563Q dispose aussi – comme bien des étages finals pour véhicules automobiles – d'une série de fonctions de diagnostic et de sécurité.

– Température

Si la température du TDA1563Q dépasse 150 °C, la puissance de sortie est limitée automatiquement à 5 watts par canal.

– Court-circuit

L'amplificateur est à l'épreuve d'un court-circuit des sorties haut-parleur avec la masse, la tension d'alimentation positive et entre elles. Si le court-circuit avec une tension d'alimentation est permanent, les étages de sortie se déclenchent définitivement, dans le cas d'un court-circuit du haut-parleur, l'étage de sortie est périodiquement déclenché puis enclenché. La

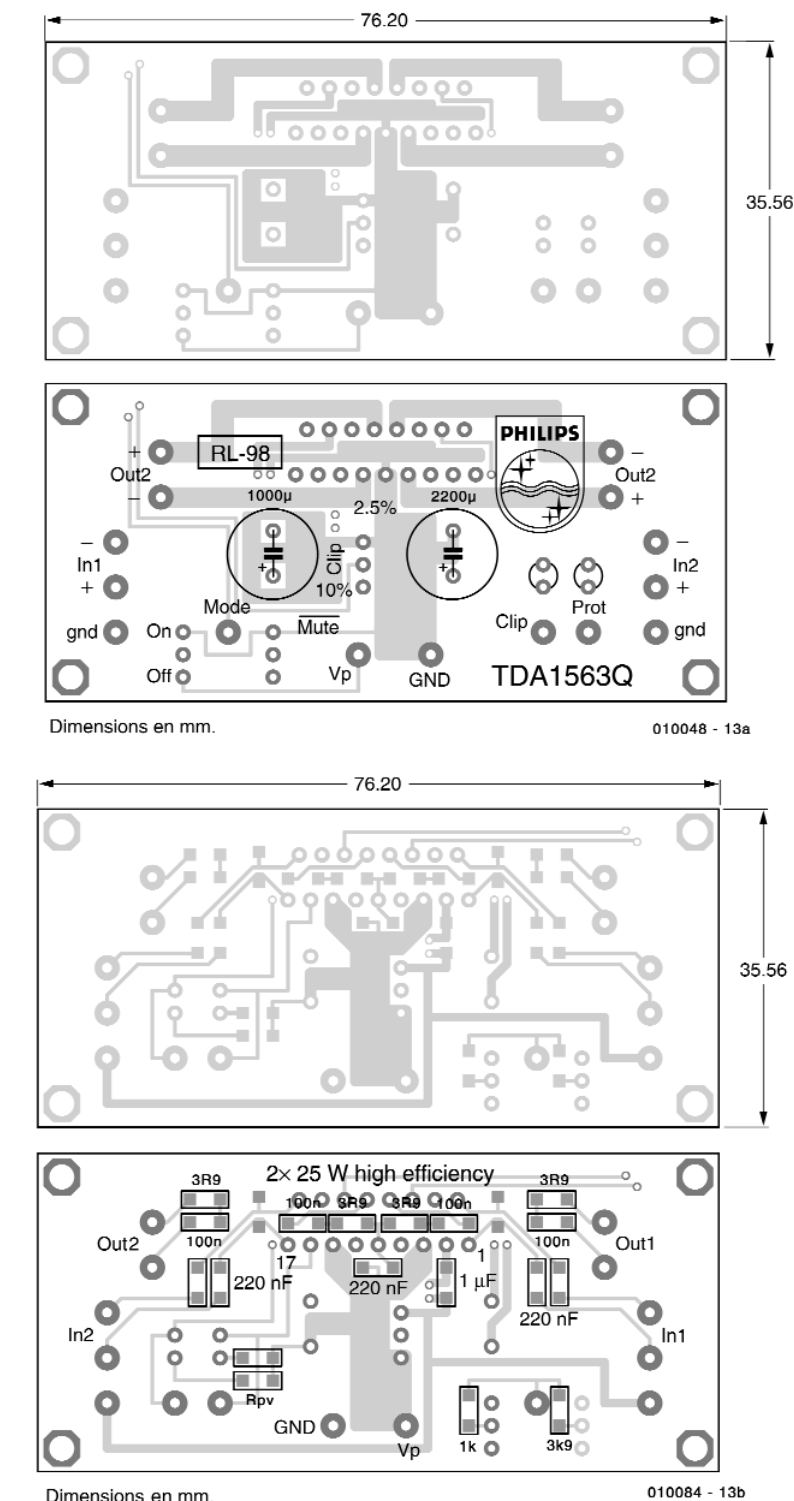


Figure 3. Une carte qui ne sert qu'à expérimenter.

phase de mise sous tension est si brève qu'un excès de puissance dissipée n'est pas à craindre.

- La puce TDA1563Q possède **3 modes de fonctionnement** : un mode d'attente (*Standby*) avec une consommation de moins de

50 μA , un mode silence (*mute*) où le signal BF à l'entrée de l'amplificateur est coupé et un mode de marche normale (*On*). La tension à la broche 6 (MODE) détermine l'état : < 1 V : Standby
2 à 3 V : Mute
> 4 V : On

Si un signal B.F. se trouve à l'entrée, un détecteur de passage par zéro intégré empêche la commutation de « Mute » à « On » ou de « On » à « Mute/Standby » à un autre moment que lors du passage par zéro. Cela évite un « plop » lors de la commutation. Si toutefois la tension descend au-dessous de 6 V (par exemple lors de la mise en marche), le circuit intégré commute directement sur « Mute ».

- La puce TDA1563Q possède **deux sorties de diagnostic**. Les deux sorties sont à collecteur ouvert et nécessitent des résistances de charge. La sortie CLIP devient Basse si la valeur THD (*Total Harmonic Distorsion* = distorsion harmonique totale) dépasse 2,5 % (entrée SC 1,5 à 18 V) ou 10 % (SC <0,5 V).

La deuxième sortie DIAG devient Basse si un court-circuit se produit, au cours de la coupure automatique du son lors de la mise en marche du circuit intégré, et enfin lorsque le seuil d'avertissement de température de 145 °C est atteint (le fonctionnement en pont est interrompu à 150 °C).

Application

La **figure 3** montre le tracé d'une platine double face pour l'application de la figure 2. Le tracé provient de la fiche de données de Philips. Hormis les 2 condensateurs électrolytique, tous les composants sont à monter en surface, non seulement pour réduire au maximum l'espace nécessaire dans le bâti de la radio, mais aussi pour réduire la vulnérabilité du circuit. Le montage ne présente

presque aucun problème. N'oubliez pas les orifices de métallisation ! Les 2 résistances RPU aux sorties de diagnostic ne sont pas dimensionnées. Les valeurs sont fonction de la tension logique et peuvent être facilement calculées lorsqu'on sait que le courant est de 2 mA à la connexion active.

Le mode est déterminé par 2 commutateurs (On/Off, $\overline{\text{Mute}}$). Le diviseur de tension est dimensionné de sorte que la tension correspondant à un mode soit appliquée à la broche 6.

La structure compacte de cette application la destine aux installations actives de véhicules automobiles dont chaque haut-parleur a besoin de son propre amplificateur de puissance. Le TDA1563Q forme avec un haut-parleur efficace (par exemple muni d'un pavillon acoustique) le couple idéal qui peut fonctionner des jours durant sur une batterie d'automobile « au repos ».

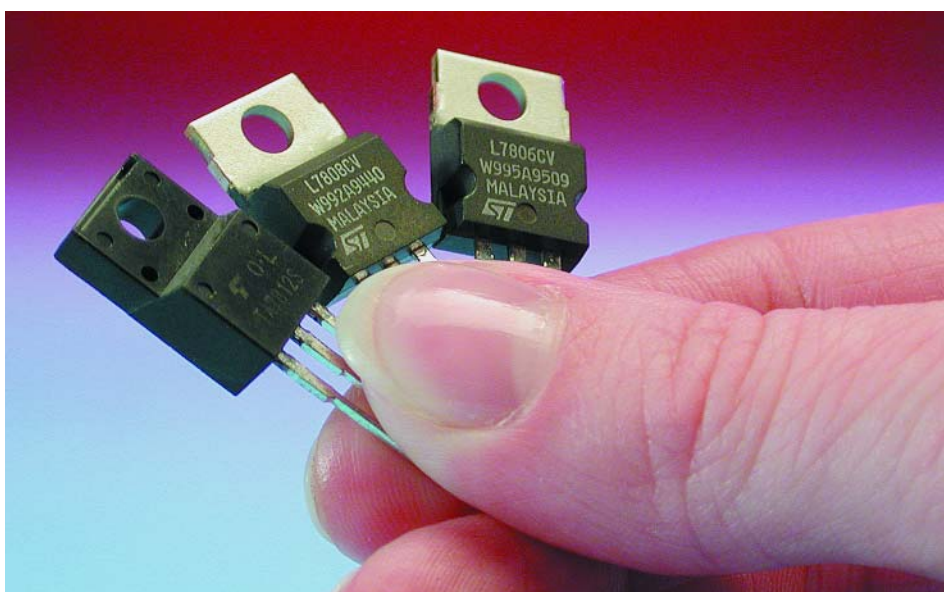
(010084)

La pratique

Des régulateurs de tension

d'après une idée de Alexander Voigt

La réputation des régulateurs de tension est d'être des composants pratiques et ne posant pas le moindre problème. Il n'en reste pas moins qu'il existe l'une ou l'autre caractéristique intéressante méritant d'être mentionnée. Comme souvent, la pratique s'écarte légèrement de la théorie.



Depuis l'apparition des régulateurs de tension tripodes, les régulateurs réalisés en technologie discrète ont quasiment disparu. Pourquoi se compliquer la vie si l'on peut se tirer d'affaire à l'aide d'un seul composant ? Ils existent pour quasiment n'importe quelle tension de sortie souhaitable et les spécifications des membres de la série 78xx, fameuse s'il en est, conviennent à la grande majorité des applications. De plus, étant donné qu'ils sont dotés d'origine d'une protection thermique et anti-surcharge efficace, il faut vraiment y mettre de la « bonne volonté » si l'on veut que les choses tournent au vinaigre.

La seule exigence que pose ce type de régulateur tripode est que la tension brute appli-

quée à l'entrée soit supérieure d'au moins 3 V à la tension requise en sortie. Si cette condition n'est pas remplie le circuit de régulation ne peut pas fonctionner correctement. Des composants on ne peut plus pratiques que ces régulateurs de la famille 78xx vu le peu de place qu'ils requièrent sur la platine et le fait qu'il ne leur faut pratiquement pas de composant externe. Dans la plupart des cas, le schéma d'un circuit de régulation basé sur un tel régulateur de tension ressemble à celui représenté en **figure 1**. Un pont de diodes redresse la tension fournie

par le transformateur, le condensateur C1 se chargeant ensuite du lissage de la tension disponible en aval du pont de redressement. C2 et C3 servent à améliorer la stabilité du régulateur de tension ainsi que sa réponse aux transitoires, le dernier condensateur, C4, faisant office de tampon local pour la charge connectée au circuit.

L'inconvénient des schémas standard est qu'ils ne rendent pas clairement les fonctions spécifiques des différents composants. Il serait de ce fait plus parlant de dessiner le schéma de la figure 1 de la façon illustrée en **figure 2**. Si son aspect est (peut-être) moins structuré et moins beau on y voit du moins immédiatement à quoi servent les différents composants qu'il comporte. Lors de la réalisation pratique c'est d'ailleurs ainsi qu'il faudra, de préférence, les disposer : le condensateur de lissage C1 le plus près possible du pont de redressement, C2 et C3 directement, respectivement, à l'entrée et à la sortie du régulateur et C4 à la distance la plus faible possible de la charge. Autre point tout aussi important d'ailleurs, veiller à ce que les lignes de masse soient ramenées à un point nodal central, le côté « froid » du condensateur électrochimique de sortie constituant, techniquement, l'endroit le plus pratique.

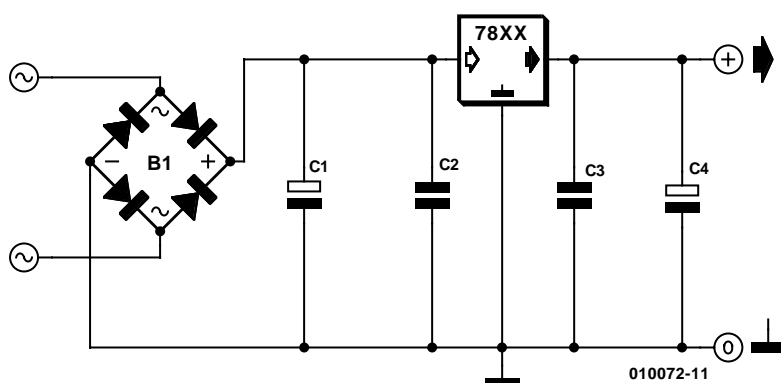


Figure 1. Schéma standard d'application d'un régulateur de tension de la série 78xx.

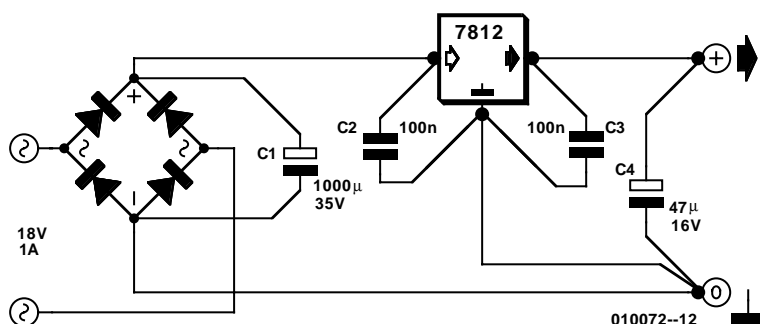


Figure 2. Il est plus parlant, d'un point de vue fonctionnel, de dessiner le schéma de la figure 1 de cette façon.

Si l'on opte pour la disposition de la figure 2, la stabilité, la réjection du ronflement et le comportement impulsionnel du régulateur de tension sont sensiblement meilleurs que ceux obtenus lors d'une disposition des composants au petit bonheur la chance, surtout lorsque, sur la platine, de par les circonstances, la distance entre le pont de redres-

sement et la charge est relativement importante.

Un petit mot au sujet de la valeur des condensateurs. En pratique, une valeur de 100 nF pour C2 et C3 convient parfaitement en toutes circonstances. La valeur du condensateur électrochimique C4 n'est pas très critique et pourra varier, en général, entre, disons 10 et 47 µF,

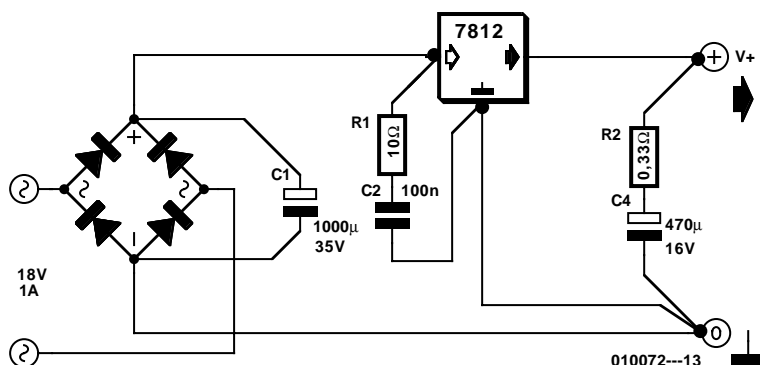


Figure 3. Une paire de résistances-série permet, dans certains cas, d'améliorer la réponse impulsionnelle.

ceci en fonction du courant de sortie.

En ce qui concerne le condensateur de lissage C1, la règle classique consiste à lui attribuer une capacité, exprimée en µF, au minimum égale, voire de préférence 2 fois supérieure au courant exprimé en milliampères. Ainsi, dans le cas du courant de 1 A (1 000 mA) du schéma de la figure 2, nous opterons pour une valeur de 1 000 ou, mieux encore, de 2 200 µF.

2 résistances de plus

Bien que nous n'ayons pas réussi, au laboratoire d'Elektor, à reproduire de façon indiscutable les constatations de l'auteur, ce dernier affirme qu'il est possible, à l'aide d'une petite modification, d'améliorer sensiblement un circuit à base de 78xx standard. Cette amélioration se traduit, en particulier, par une meilleure réponse impulsionnelle dans le cas de variations de charge rapides.

La modification se résume en fait à la mise en place, au niveau de C2 et C4, de 2 résistances-série. On peut supprimer le condensateur C3 sans que cela n'ait d'effet sur la qualité de fonctionnement du circuit et augmenter la valeur de C4. Cela nous donne alors le schéma modifié représenté en figure 3.

Il est difficile de fournir une explication scientifique de l'amélioration des caractéristiques obtenue. L'une des théories possible est que, dans certaines conditions, les condensateurs et les lignes de liaison se mettent à constituer des réseaux LC, ce qui se traduit par des oscillations qui empêcheraient le régulateur de tension de réagir de façon optimale. Une résistance-série de faible valeur pourrait avoir un effet amortisseur favorable. Il se peut que la marque (l'origine partant) du régulateur de tension joue elle aussi un certain rôle.

Bien qu'il soit difficile d'affirmer que les résistances-série aient, quel que soit le cas de figure, un effet favorable, il nous paraît valoir la peine de tenter l'expérience. Le coût des composants additionnels est négligeable et si le comportement dynamique du régulateur s'en voit effectivement amélioré, cela sera toujours cela de pris.

Il est difficile de donner des valeurs concrètes pour R1 et R2. Sur le circuit expérimental à base de 7812 de l'auteur, les valeurs données en figure 3 ont donné les meilleurs résultats. Après moult expériences, il apparaît que la valeur de R2 répond au principe : plus la capacité du condensateur électrochimique C4 est importante, plus la valeur de cette résistance pourra être diminuée. La valeur de R1 est moins critique. Expérimenter consiste, à notre avis, la meilleure solution pour en découvrir la valeur à utiliser en pratique.

(010072)

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 regrettons 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.

Programmeur Atmel

Le fonctionnement du programme grillé dans le programmeur Atmel décrit dans le numéro de septembre 2001 (010005) n'est pas parfait, ce qui se traduit par une programmation incorrecte du dernier octet du processeur programmé. Cela ne pose de problème que dans le cas où l'on voudrait griller dans le composant un programme utilisant jusqu'au dernier octet disponible. Les lecteurs qui auraient un problème peuvent nous renvoyer leur disquette originale (010005-11, ne pas oublier de mentionner l'adresse) que nous remplacerons par une version corrigée du programme.

Transfo ETD-29

Dans un ancien numéro d'Elektor vous aviez proposé un convertisseur TL 12 V. J'envisageai de le réaliser parce que je ne trouve nulle part ailleurs un montage similaire capable de travailler avec ces ampoules de 8 watts. J'ai cependant des problèmes pour me procurer le transformateur du type EB29 de Block à réaliser soi-même. Personne n'est en mesure de m'aider. De plus, comme vous avez utilisé exactement le même transformateur dans votre alimentation haute tension publiée dans le numéro de janvier 2001, je reviens à la charge. Auriez-vous une adresse ou le site Internet d'un fabricant ou d'un distributeur ?

Luuk Wendels

Nous avons eu plusieurs questions quant à la disponibilité de ce transformateur à réaliser soi-même. Étonnant, car ce transformateur est toujours encore en production chez Block et certains distributeurs tels que Farnell (www.farnell.com) l'on en stock. Redonnons une fois encore les composants dont il est constitué :

2 x demi-coquille

ETD29 (n° 305-6375)

1 x bobine ETD29 (n°178-506) et

2 x clips acier ETD29
(n°178-507).

Site EDiTS Pro

Outre le Forum de langue française consacré à EDiTS Pro sis à l'adresse :

<http://fr.groups.yahoo.com/group/editspro>

Signalons également l'existence du site de l'auteur proposé en 2 langues, le néerlandais et l'allemand, ce qui ne l'empêche pas d'être intéressant par les nombreuses illustrations et les liens qu'il propose. Si EDiTS Pro vous intéresse, faites donc un tour à : www.gironet.nl/home/editspro

Booster pour EDiTS

Dans l'article consacré à la nouvelle mouture de booster pour EDiTS (numéro de juillet/août 1999), il est indiqué dans la liste des composants un transformateur de 2 x 25 V, alors que dans toutes les autres publications relatives à ce booster on parle d'un transformateur de 2 x 18 V. Quelle est la puissance correcte ?

Ad van Iersel

La liste des composants en question est malheureusement erronée. La tension redressée doit être de 2 x 25 V, de sorte que le transformateur pourra bien être du type 2 x 18 V. Nos excuses pour cette erreur.

Réalisation de circuits imprimés

Bonjour,
Désirant réaliser le "Switch vidéo/audio" ainsi que l'« Émetteur IR rustique » et le « Petit récepteur IR » parus dans le numéro Hors gabarit 2001, j'ai commandé les circuits imprimés : 014119-1, 014120-1, 014121-1. J'ai appris que ces circuits imprimés n'étaient pas commercialisés ??? J'ai donc téléchargé les fichiers zip de ces circuits. Je les ai ensuite imprimés à l'aide d'une imprimante laser. C'est

beau. Ma question est :

Comment réaliser le négatif ? car je ne connais pas l'échelle exacte de ces dessins.

Merci d'avance pour le renseignement

M. RENOUE par E-mail

Nous ne pouvons malheureusement pas commercialiser toutes les platines dont nous proposons les dessins dans les différents numéros d'Elektor. Ceci est d'ailleurs l'une des raisons pour lesquelles nous proposons les dessins des platines au téléchargement depuis notre site. Une note générale pour commencer. Sauf mention contraire, les dessins proposés le sont à l'échelle 1:1. Il suffit partant de faire faire un film par un photographe et/ou d'utiliser une platine photosensible sur laquelle sera posé ou le film ou la photocopie du dessin des pistes réalisé sur un support transparent.

Il reste ensuite à développer et à graver la platine. Un ensemble d'opérations plutôt délicat si on n'en a pas l'habitude. Mais c'est en forgeant que l'on devient forgeron. Notons qu'il existe encore une autre possibilité, lire les petites annonces où de temps à autre un lecteur d'Elektor propose de réaliser les platines non commercialisées ou épuisées d'Elektor (cf. numéro de septembre 2001).

Thermomètre numérique à LED

Je me suis lancé dans la réalisation du thermomètre numérique à LED décrit dans le numéro d'avril 2001 d'Elektor et j'ai découvert une erreur dans la liste des composants. Chez Farnell ils ne connaissent pas le code 760-704. Le code correct pour le DS1621 est 670-704. Une petite inversion aux conséquences importantes.

Rik Deketelaere

Merci de cette correction que nous rendons publique.

LS contre HC

Est-il possible, dans l'affichage modulaire à matrice de points décrit dans le numéro de juin 2001, de remplacer les 74 LS138 et 74LS164 par leur version HC ou cela est-il risqué sans modification ?

Tom Marchand

D'un point de vue électronique les 2 familles que vous évoquez sont identiques de sorte qu'il ne devrait pas y avoir, à ce niveau, de problème de substitution. Dans le cas du projet en question il n'est cependant pas permis d'effectuer ce remplacement, pour la bonne et simple raison que les circuits LS sont capables de supporter un courant plus important, caractéristique dont il a été ici tiré profit.

Brancher une radio portable au câble ?

Dans mon appartement d'étudiant j'utilise une radio-cassette qui ne possède qu'une antenne télescopique et pas d'entrée antenne. Il m'est partant impossible de capter les stations qui n'émettent que par le câble. Existe-t-il une solution à mon problème ou dois-je me résoudre à acheter un autre récepteur radio doté d'une entrée antenne ?

Wyben

Il n'est pas nécessaire que vous fassiez les frais de l'achat d'un nouveau récepteur radio, car il existe peut-être une solution qui si elle n'est pas élégante, a l'avantage d'être simple. Il faudra rentrer l'antenne télescopique à fond et la relier par le biais d'une petite pince crocodile à l'âme (conducteur central) du câble coaxial du système d'antenne centrale. Il n'est pas nécessaire de relier le manchon externe qui sert de blindage.

Cette façon de procéder permet dans bien des cas une excellente réception des signaux véhiculés par le câble.

Interface disque dur pour port imprimante

IDE2LPT - branchez un DD sur le port Centronics

Projet : Andrew Buckin (idée : Leonid Slobodchikov)

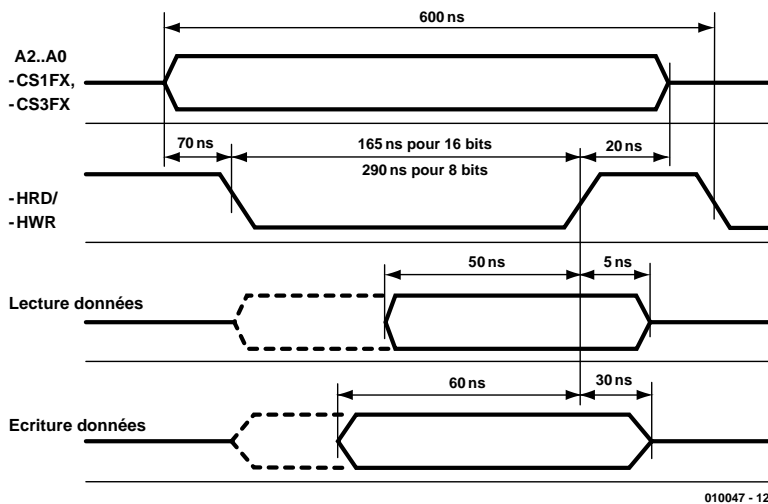
La présente interface permet de connecter n'importe quel disque dur de type IDE au port imprimante d'un PC, ce qui permet, par exemple, de transporter facilement des quantités importantes de données. L'interface IDE2LPT convient tout aussi bien aux interfaces parallèles de type classique qu'aux versions les plus récentes travaillant en mode ECP et EPP. L'utilisation d'une EPLD d'Altera permet de réaliser un montage joliment compact.



Qui pourrait, de nos jours, imaginer un PC sans disque dur. Si, voici 2 lustres à peine, le disque dur brillait par sa taille et son prix, on

en trouve aujourd'hui des modèles très compacts et de capacité impressionnante à des prix très abordables

(si tant est que l'on puisse dire que 150 € pour 40 Goctets soit abordable). Le standard le plus courant



010047 - 12

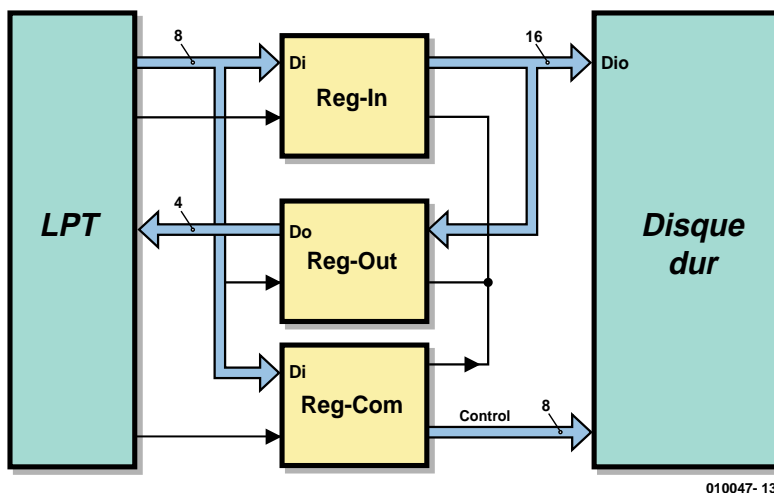
Figure 1. Le chronodiagramme visualise l'évolution des signaux sur le bus IDE lors de processus de lecture et d'écriture.

dans le monde des disques durs reste de nos jours l'interface IDE. Depuis qu'il existe des ordinateurs le problème de sauvegarder et de transporter des données sur des médias de supports de données est un problème connu. Lorsqu'il s'agit du transport de quantités de données importantes (et comme tout le monde le sait de nos jours les quantités de données concernées croissent de façon exponentielle) il peut être intéressant d'envisager d'utiliser un disque dur à interface IDE compact qui deviennent de plus en plus bon marché aujourd'hui. Le seul problème auquel se trouve confronté cette approche est qu'il n'existe pas de PC

doté d'une interface IDE externe. Au contraire, chaque PC dispose d'au moins un port imprimante parallèle. Si l'on prend la peine de comparer la technologie du port imprimante à celle de l'interface IDE, on ne manquera pas de constater quelques similitudes, qui ont encouragé l'auteur à développer une interface IDE vers LPT. L'interface présentée ici est une des différentes versions ayant vu le jour, au coeur de laquelle bat une CPLD d'Altera.

LPT vers IDE

Dans le numéro de mars 2001 d'Elektor nous vous avons proposé une



010047- 13

Figure 2. Synoptique de principe du convertisseur LPT/IDE.

En résumé :

Capacité du disque dur : illimitée
(ne dépend que du type de DD utilisé)

Nombre de disques durs connectables : 1

Transfert de données : jusqu'à 100 Koctets/s

Alimentation : 5 V / 100 mA

Tableau I Brochage de l'interface ATA

Broche	Désignation	Description
1	HRESET	Remise à zéro (reset)
2	GND	Masse (GND)
3	HD7	Bus de données bit 7
4	HD8	Bus de données bit 8
5	HD6	Bus de données bit 6
6	HD9	Bus de données bit 9
7	HD5	Bus de données bit 5
8	HD10	Bus de données bit 10
9	HD4	Bus de données bit 4
10	HD11	Bus de données bit 11
11	HD3	Bus de données bit 3
12	HD12	Bus de données bit 12
13	HD2	Bus de données bit 2
14	HD13	Bus de données bit 13
15	HD1	Bus de données bit 1
16	HD14	Bus de données bit 14
17	HD0	Bus de données bit 0
18	HD15	Bus de données bit 15
19	GND	Masse (GND)
20	N/C	Broche de détrompage
21	DMARQ	Demande d'accès direct à la mémoire (DMA request)
22	GND	Masse (GND)
23	HWR	Écriture E/S (I/O Write)
24	GND	Masse (GND)
25	HRD	Lecture E/S (I/O Read)
26	GND	Masse (GND)
27	IORDY	Canal d'E/S paré (I/O Channel Ready)
28	SPSYNC:CSEL	SPINDLE SYNC ou CABLE SELECT
29	DMACK	Acquittement DMA (DMA Acknowledge)
30	GND	Masse (GND)
31	INTRQ	Demande d'interruption (Interrupt Request)
32	IOCS16	E/S 16 bits (16 bit I/O)
33	HAI	Bus d'adresse bit 1
34	PDIAG	Diagnostics OK (Passed Diagnostics)
35	HA0	Bus d'adresse bit 0
36	HA2	Bus d'adresse bit 2
37	CS1FX	Sélection de circuit 0 (Chip Select 0)
38	CS3FX	Sélection de circuit 1 (Chip Select 1)
39	DASP	Lecteur actif/lecteur 1 présent (Drive active/drive 1 present)
40	GND	Masse (GND)

interface similaire permettant l'utilisation d'un disque dur IDE sur le bus 8 bits de microcontrôleurs. Dans le cadre du dit article nous avons décrit dans le détail l'interface IDE développée par Compaq pour une connexion facile de disques durs au bus du (AT-)PC, de sorte que nous ne passerons en revue dans le présent article que les signaux IDE les plus importants (pour ses spécifications complètes on pourra faire un tour sur le site Internet sis à l'adresse www.t13.org). Le **tableau 1** donne le brochage de la liaison IDE à 40 lignes. L'interface LPT2IDE utilise, en ce qui la concerne, les signaux suivants :

- HD15 à HD0** Lignes de données (largeur de bus de 8 ou 16 bits, bidirectionnel)
- CS1FX** (Chip Select 0) Sert au décodage du bloc du registre de commande (*command register*)
- CS3FX** (Chip Select 1) Sert au décodage du bloc du registre d'état (*status register*)
- HRD** (I/O-read) Signal de lecture des adresses de port d'E/S. Lorsque cette ligne est au niveau bas le disque dur place des données sur le bus (HD0 à HD7 voire HD0 à HD15). Lors du flanc montant de ce signal l'hôte verrouille les données dans son registre.
- HWR** (I/O-write) Signal d'écriture pour les

Tableau 2 Signaux et registres : Interface LPT et convertisseur LPT/IDE						
Adresses LPT/LPT2	Description					
378h/278h	Registre de données - DATA7 à DATA0 (écriture seule)					
379h/279h	Registre d'état - (écriture seule)					
	Bit	LPT Nom	IDE2LPT Nom	État	État en lecture	Fonction
	7	BUSY	LI3	I	0	DATA3 du convertisseur
	6	SLCT	LI2	I	I	DATA2 du convertisseur
	5	PE	LI1	I	I	DATA1 du convertisseur
	4	ACK	LI0	I	I	DATA0 du convertisseur
	3	ERROR	—	I	I	—
	2	—	—	—	0	—
	1	—	—	—	0	—
	0	—	—	—	0	—
37ah/27ah	Registre de commande					
	Bit	LPT Nom	IDE2LPT Nom	État	État en lecture	Fonction
	7	—	—	—	0	—
	6	—	—	—	0	—
	5	—	—	—	0	—
	4	—	—	—	0	—
	3	SLCT_IN	RCWR	0	I	Signal d'écriture pour Reg_COM
	2	INIT	RLWR	0	0	Signal d'écriture pour Reg_IN de DATA7 à DATA0 vers HD7 à HD0
	1	AUTO_FD	RHWR	0	I	Signal d'écriture pour Reg_IN de DATA7 à DATA0 vers HD15 à HD8
	0	STROBE	HRESET	0	I	RAZ (Reset) HDD

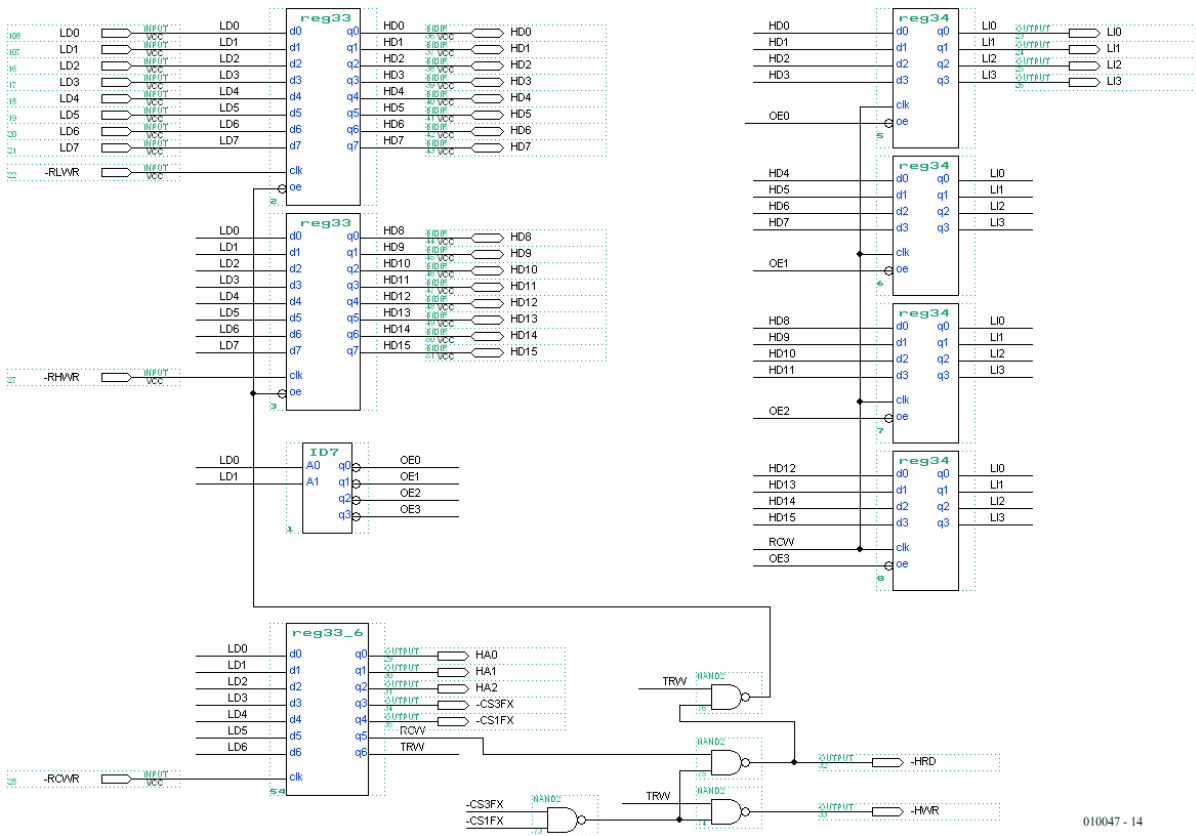


Figure 3. L'électronique produite en AHDL et créée par programmation de la CPLD.

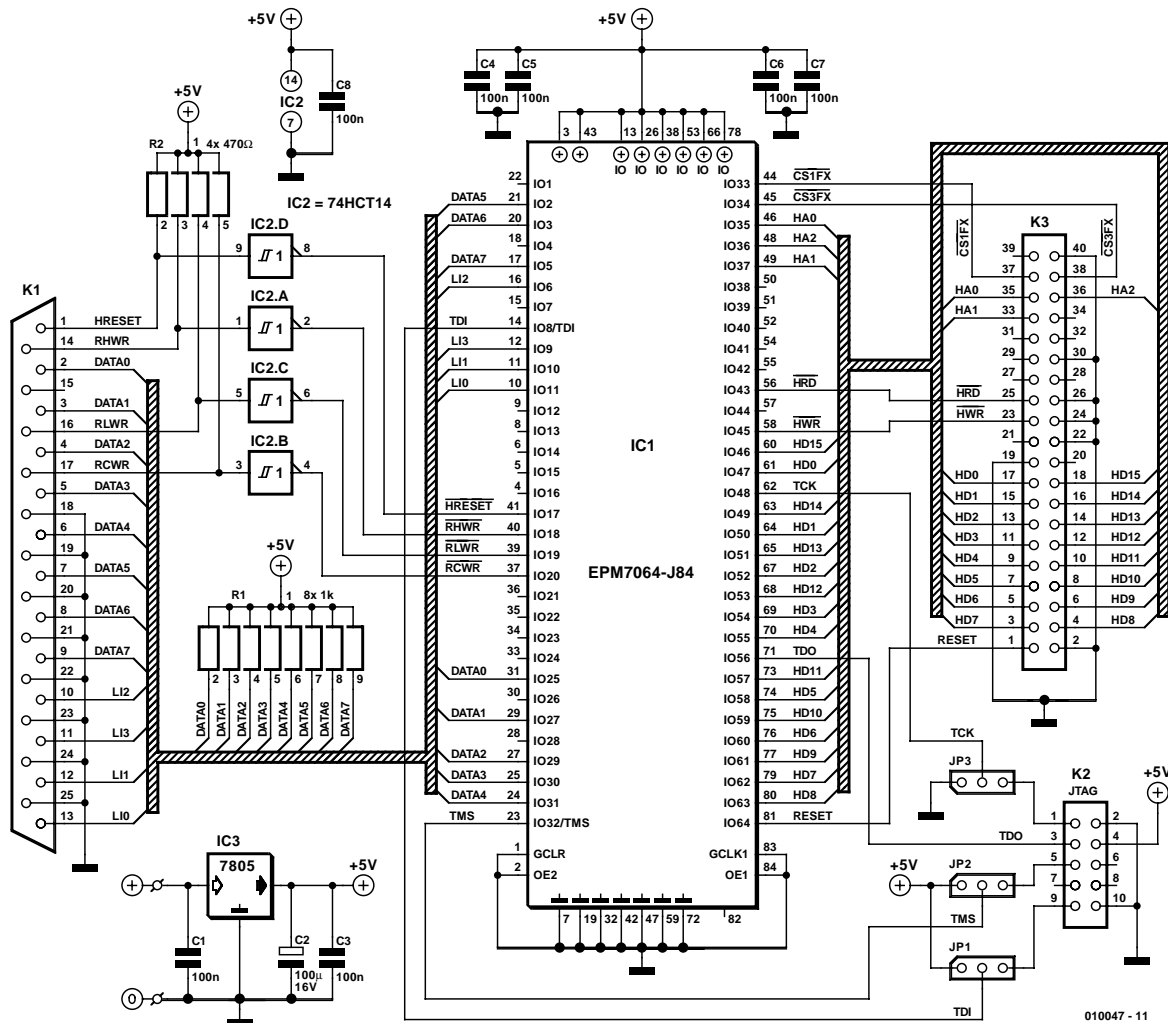


Figure 4. Schéma du convertisseur LPT/IDE. Le connecteur JTAG sert à la programmation en site (in-circuit programming) de la CPLD.

Tableau 3 Signaux du registre de commande dans le convertisseur LPT/IDE

Bit	Nom	Description
7	—	—
6	TRW	Signal d'écriture HWR sur le bus IDE Le niveau actif est « 1 »
5	RCW	Signal de lecture HRD sur le bus IDE Le niveau actif est « 1 »
4	-CS1FX	Sélection de circuit (Chip Select) 0 Le niveau actif est « 0 »
3	-CS3FX	Sélection de circuit (Chip Select) 1 Le niveau actif est « 0 »
2 à 0	HA2 à HA0	Bus d'adresse

adresses de port d'E/S. Lors du flanc descendant de ce signal les données (HD0 à HD7 ou HD0 à HD15) présentes sur le bus sont transférées dans le registre du disque dur.

HA0 à HA2 (bit 0 et bit 2 du bus d'adresse) La sélection des registres ou des ports du disque dur se fait par le biais de ces lignes d'adresses.

HRESET (Reset) Un niveau bas sur cette ligne initialise le disque dur.

Le diagramme impulsif du **figure 1** visualise les signaux véhiculés par le bus IDE en cours de lecture et d'écriture.

L'autre côté de l'interface LPT2IDE se voit connectée, par l'intermédiaire de son connecteur Sub-D à 25 contacts, au port imprimante (LPT) du PC. Cette interface parallèle possède un bus de données d'une largeur de 8 bits, comporte 6 lignes de signal utilisés pour le pilotage de l'imprimante ainsi que 5 lignes de

signalisation retour de l'état de l'imprimante (lignes d'état).

Dans le cas du port parallèle numéro 1 dit LPT1, ces lignes sont associées au registre de donnée à l'adresse 378_{HEX} (écriture seule), au registre de commande à l'adresse 37A_{HEX} (écriture seule) et au registre d'état à l'adresse 379_{HEX} (lecture uniquement); (cf. le **tableau 2** qui donne en outre les adresses de registre du port LPT2). Bien que la vitesse du port imprimante ne dépasse pas 150 Koctets, cette interface n'en présente pas moins l'avantage de permettre une commande logicielle totale de tous les signaux. Cette réalité permet un transfert de données sans complication et la création sans difficulté de l'électronique d'interfaçage LPT2IDE.

L'électronique

Il n'est pas nécessaire, sachant que le disque dur traite à son niveau toutes les instructions de l'interface IDE, de mettre en oeuvre la tota-

lité des possibilités de l'interface IDE du disque dur telles que DMA-Request, le transfert de données sur 16 bits, la mise en veille (*standby*) et le diagnostic. De ce fait, le convertisseur assurant le passage de LPT vers IDE se résume, en principe, à un tampon intercalé entre l'interface LPT et l'interface IDE du disque dur. Comme le montre le schéma de la **figure 2**, la connexion vers le disque dur se fait au travers d'un registre de données bidirectionnel et d'un registre d'instructions unidirectionnel. Les données sur 8 bits en provenance du PC subissent un stockage temporaire dans le registre de données du convertisseur pour ensuite être lues, à la sortie, sous une largeur de 16 bits en direction du disque dur. Lors de la prise en compte de données en provenance du disque dur le processus se fait dans le sens inverse. Les données sur 16 bits fournies par le disque dur sont stockées temporairement dans le convertisseur, pour ensuite être transférées,

Liste des composants

Résistances :

R1 = réseau de 8 résistances de 1 k Ω
R2 = réseau de 4 résistances de 470 Ω

Condensateurs :

C1, C3, C8 = 100 nF
C2 = 100 μ F/16 V vertical

Semi-conducteurs :

IC1 = Altera 7064LC84-15 (**programmé
EPS 010047-31**)
IC2 = 74HCT14N
IC3 = 7805

Divers :

JP1 à JP3 = embase à 1 rangée de 3 contacts
+ cavalier de court-circuit
K1 = embase Sub-D mâle à 25 contacts
encartable
K2 = embase mâle à 2 rangées de
5 contacts avec guide (HE-10)
K3 = embase mâle à 2 rangées de
20 contacts avec guide (HE-10)
2 picots
câble prolongateur pour alimentation de PC
(cf. texte)
câble prolongateur pour imprimante (1:1)
câble IDE

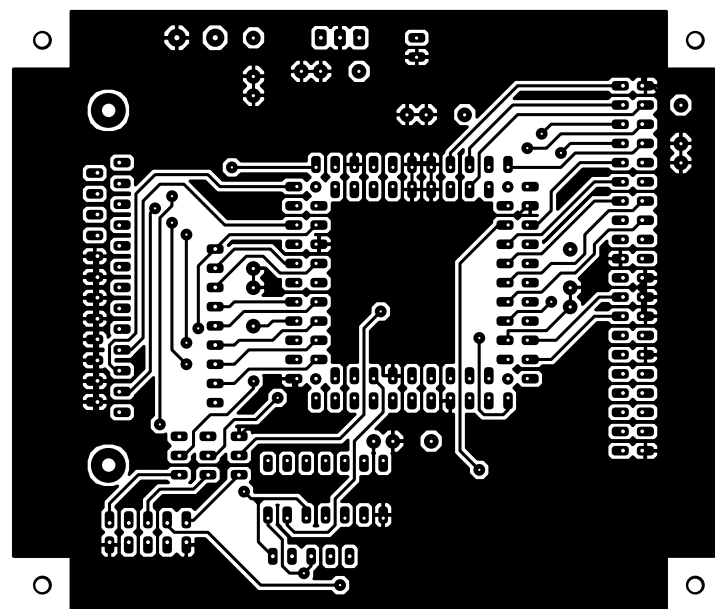
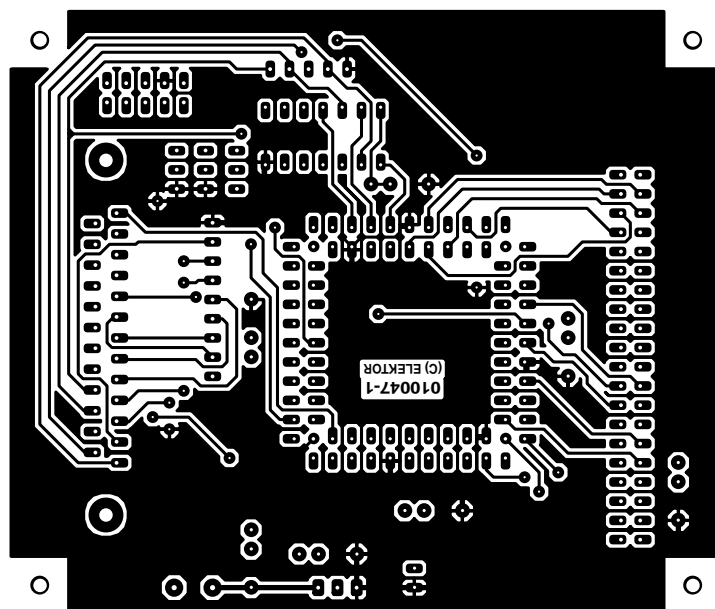
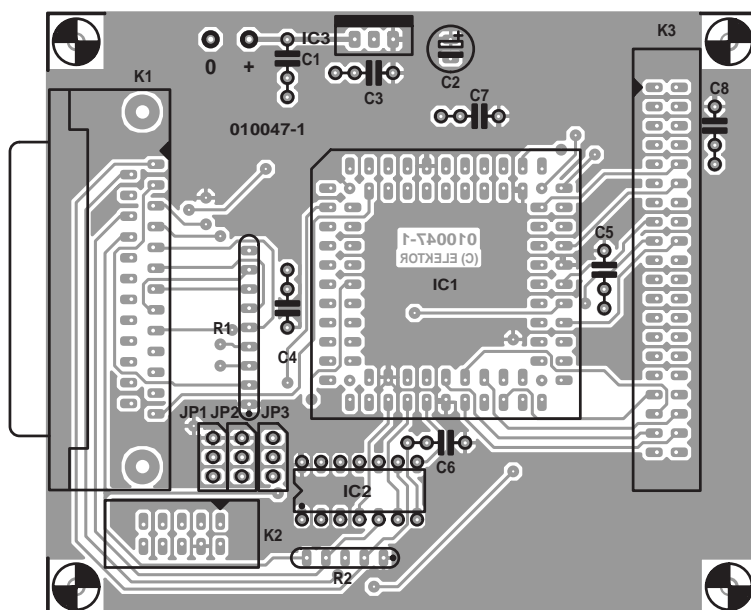
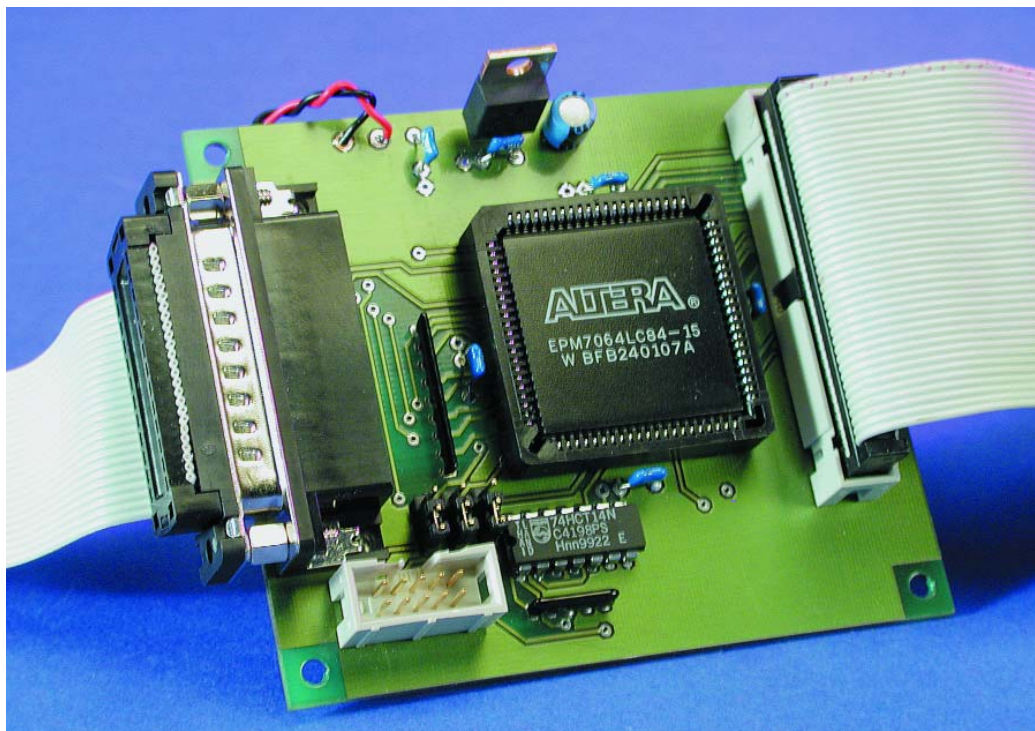


Figure 5. La sérigraphie de l'implantation des composants de la platine double face à trous métallisés rend bien la fonction de cette électronique : IC1 assure l'interconnexion entre une embase LPT d'une part et une embase IDE de l'autre.

sous une largeur de 4 bits seulement alors, vers l'interface imprimante. L'ordinateur peut à son tour lire ces données depuis le registre d'état de l'interface imprimante. C'est l'ordinateur, par le biais du registre de commande (*control register*, cf. **tableau 3**), qui se charge du pilotage du transfert des données. On dispose en outre d'une logique de commande qui interdit toute situation illicite au niveau du bus IDE (cf. **tableau 4**).

En vue de l'intégration du convertisseur LPT/IDE dans un circuit CPLD les fonctions requises (registres et logique) doivent être décrites en AHDL, un langage de programmation. Le listage des définitions est disponible au téléchargement sur le site Elektor (www.elektor.presse.fr)

parmi les autres téléchargements de ce numéro. Le schéma créé en ADHL résultant de ces définitions est proposé en **figure 3**. Ce schéma est réalisé par la programmation de la CPLD du type EPMM 7064 d'Altera. Ce circuit intégré prend à son compte une bonne partie de l'électronique du convertisseur LPT/IDE dont le schéma est donné en **figure 4**. Les 2 autres circuits intégrés requis par cette réalisation sont des périphériques : IC2 est un 75HCT14, qui fait office, par le biais de ses triggers de Schmitt, de metteur en forme d'impulsions pour les signaux \overline{RHWR} , \overline{RLWR} , \overline{RCWR} et \overline{HRESET} , ceci en vue d'améliorer l'insensibilité aux parasites. IC3 est l'indispensable régulateur de tension chargé de fournir le 5 V requis. La 3^{ème} embase du montage (outre K1 pour la connexion à l'interface LPT et K3 pour le branchement du disque dur) est ce que l'on appelle l'interface JTAG de la CPLD. Il est possible, par le biais de cette interface -reliée ici à l'embase K2- de déboguer et de programmer IC1. Il faut cependant, pour cela, ajouter une électronique de programmation externe et disposer du logiciel de débogage et de programmation correspondant. Le logiciel de programmation « MAX2Plus » est disponible



gratuitement sur le site Internet d'Altera (www.altera.com), le programmeur correspondant, baptisé BitBlaster qui lui, s'il n'est pas gratuit, n'est cependant pas très cher non plus. Seuls les CPLD de la famille MAX7000 en version S d'Altera (la EPM7064S dans le cas présent) possèdent l'interface JTAG. Les 3 signaux de programmation TCK, TMS et TDI se trouvent normalement forcés à la masse par le biais de 3 cavaliers de court-circuit, JP1 à JP3; ce n'est que pour une programmation qu'ils sont transférés aux broches correspondantes de K2 par la mise en position adéquate des dits cavaliers. Il n'est pas nécessaire, si l'on se contente de réaliser l'interface LPT/IDE, d'acheter un BitBlaster : IC1 est en effet disponible programmé auprès des adresses habituelles.

La platine dessinée à l'intention de ce montage et dont on retrouve le dessin des pistes et la sérigraphie de l'implantation des composants en **figure 5**, est elle aussi disponible à ces adresses. Si vous voulez fabriquer vous-même votre platine double face vous trouverez sur le site Internet d'Elektor les dessins de platine à télécharger (de même que le listage comme mentionné plus haut).

Le logiciel

Il existe, pour le convertisseur LPT/IDE « LPT2IDE » un pilote **et** pour DOS **et** pour Windows 9x. Le pilote DOS « *ide2lpt.exe* » tourne sous MS-DOS à partir de la version 3.3 et ne se différencie guère des autres pilotes IDE. Il n'est partant pas nécessaire d'écrire un pilote spécifique à partir de zéro; on pourra adapter un pilote IDE existant pour l'écriture et la lecture par le biais de l'interface parallèle. Il faut cependant lui ajouter quelques fonctions telles que des fonctions test et de recherche. L'auteur remercie Ewgenij Kuleschow pour le code-source du pilote utilisé ici.

Il faudra, pour installer le pilote, ajouter une ligne au fichier config.sys :

```
device=[path]\ide2lpt.exe [options]
```

normalement :

```
device=c:\lpt2ide\i2l4.exe.
```

On pourra, pour le paramétrage et les tests initiaux, lancer le pilote directement depuis l'annonce DOS (prompt) et ajouter un paramètre à l'instruction, tel que l'option **/T (I2L4.EXE/T)** pour le test.

Autres options disponibles :

/H Aide (Help)

/L:LPT L'adresse de la sortie LPT (378_{HEX} pour LPT1 et 278_{HEX} pour LPT2)

/G:SEC:HEAD Définit les paramètres du disque dur (sert aux disques dur travaillant en mode LBA) : SEC donne le nombre de secteurs, HEAD le nombre de têtes. L'entrée de ces paramètres se fait en mode décimal.

/L mode LBA

/V 0 Interrogation des paramètres du disque dur.

Le pilote DOS travaille avec les partitions du type FAT12, FAT16, BIG et étendue (*extended*). Il ne supporte pas FAT32. Ce mode de partition est lui reconnu par le pilote Windows qui s'installe de la manière suivante (Windows 95) :

Ouvrir « Démarrer » -> « Paramètres » -> « anneau de configuration » -> « Ajout de périphérique ».

Dans l'« Assistant de nouveau matériel » faire « Suivant ».

Cliquer, dans la fenêtre « Assistant de ajout de nouveau matériel » l'option « Non » puis cliquer sur « Suivant ».

Dans la liste proposée qui s'ouvre alors choisir le « Contrôleurs SCSI » et « Suivant ».

Cliquer la « Disquette fournie ».

Cliquer sur « Parcourir ». Rechercher le répertoire (disquette ou disque dur si on y a recopié ces fichiers) avec les fichiers pilote et sélectionner les fichiers *ide2lpt/inf* et *ide2lpt.mpd* et fermer la fenêtre par un clic sur le bouton « OK ». L'installation des pilotes se fait ensuite automatiquement.

Une fois l'installation des pilotes terminée, fermer les Assistants Matériel et redémarrer Windows.

Il est recommandé, pendant l'utilisation du pilote LPT2IDE de mettre la sortie LPT hors-fonction, sachant que sinon on risque des problèmes en cas de tentative d'accès à la sortie LPT sous Windows.

Connexion

Le plan de câblage de la **figure 6** représente, schématiquement l'interconnexion du PC, de la platine du convertisseur et de l'interface disque dur. L'interconnexion du convertisseur au PC se fait à l'aide d'un câble prolongateur pour imprimante, voire un câble (en multi-brins) de caractéristiques similaires que l'on aura fait soi-même, la connexion vers le disque dur se fera à l'aide d'un câble IDE classique. Il se peut que l'utilisation d'un câble trop long entre le port imprimante et son embase sur la platine du convertisseur, K1, puisse entraîner des problèmes. On pourra, le cas échéant, découpler en HF les lignes de signal DAT 0 à DATA 7 par la mise

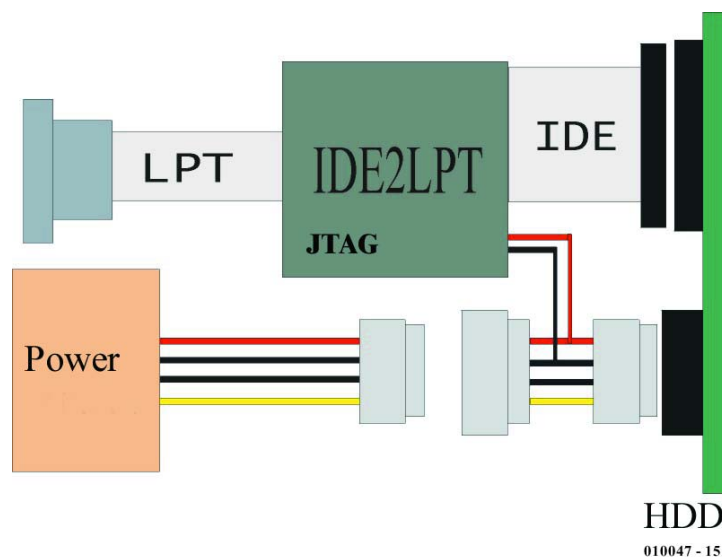


Figure 6. Le plan de câblage montre également comment effectuer l'alimentation de ce montage à l'aide d'une alimentation de PC récupérée.

en place à cet endroit de condensateurs céramique de 12 à 22 pF.

Il ne faudra bien entendu pas oublier de connecter la tension d'alimentation. Comme l'illustre la figure 6, on pourra faire appel, pour l'alimentation du disque dur et de la platine du convertisseur, à une alimentation de PC additionnelle qui fournit toutes les tensions requises et ce à une intensité plus que suffisante et qui ne coûte pas très cher, surtout si on la récupère sur un PC mis au rebut. La liaison entre cette alimentation et le disque dur se fera de préférence à l'aide d'un câble d'extension normalement utilisé à cet effet dans un PC. Depuis le connecteur d'extrémité de ce câble on prolonge 2 des lignes, +12 et 0 V vers les points de connexion de l'alimentation sur la platine du convertisseur LPT/IDE. Les 2 picots concernés disposés à proximité de C1, sont identifiés par les mentions + et 0.

Perspectives

L'auteur a prévu d'autres variantes de ce montage, l'une d'entre elles étant dotée d'un oscillateur à quartz, ce qui le rend, en ce qui concerne la chronologie, indépendant des circuits d'entrées/sorties de la carte-mère. Avec cette version future, le taux de transmission devrait passer à quelque 1,5 Moctets/s. Ceux d'entre nos lecteurs que cette variante inté-

resse peuvent s'en informer sur le site Internet de l'auteur et le contacter par le biais de son adresse E-mail.

(010047)

AndrewBuckin@aol.com
ipm_grp@yahoo.com
ipm_grp@mail.com
www.geocities.com/ipm_grp
<http://members.aol.com/andrewbuckin>
<http://hometown.aol.com/andrewbuckin>

Information de dernière minute :

L'interface a été réalisée dans les laboratoires d'Elektor (cf. photo) et testée sur différents PC avec divers DD. Ces tests ont été conclus sans le moindre problème. Tout juste avant publication, l'auteur nous a cependant fait savoir qu'il n'était pas exclu qu'il puisse y avoir des problèmes de chronologie (timing) avec l'un ou l'autre type de PC. D'après l'auteur, ce problème est cependant très peu fréquent et se manifeste dans la plupart des cas avec des cartes-mères à Pentium I (en relation avec le type de circuit d'E/S présent sur cette dernière).

DMX-512

protocole, logiciels et interfaces

Harry Baggen

Nous avons, ces derniers temps, consacré à DMX-512 un certain nombre d'articles publiés dans Elektor. Nous avons décrit plusieurs projets et n'en resterons sans doute pas là. On trouve, sur Internet, un nombre impressionnant de sociétés fabriquant ou vendant des produits DMS professionnels. Le nombre de projets réalisables par l'amateur est encore très limité. Il n'en reste pas moins qu'il est possible de découvrir quelques projets intéressants et de consulter quelques sites très informatifs.

La raison d'être du standard DMX-512 est, à l'origine, la commande d'éclairages dans le monde du théâtre et des spectacles. Depuis lors, les possibilités et les appareils disponibles ont énormément évolué au point qu'il est devenu possible de faire bien plus avec un système DMX-512 qu'uniquement télécommander des lampes. Cela le lecteur assidu d'Elektor le sait depuis belle lurette, après la lecture de l'article publié à ce sujet dans le numéro de mai dernier (« Les mystères du DMX-512 »). Il y a bien évidemment bien plus de choses à dire au sujet du protocole DMX-512 que ce que nous pouvons concentrer sur quelques pages de magazine. Le standard DMX complet est publié par l'USITT. Ce standard est disponible sous forme imprimé, la version disponible sur Internet n'étant pas complète. Il n'en reste pas moins que le site de **USITT** (*United States Institute for Theatre Technology*) [1] offre des infos et décrit les développements les plus récents dans ce domaine. Si le standard DMX vous intéresse et que vous voulez le commander vous pouvez, en Europe, vous adresser à **PLASA** (*Professional Lighting and Sound Association*) [2].

Il existe, hormis ces sources officielles, nombre d'autres sites tenus par des utilisateurs/amateurs de DMX qui donnent toute l'information nécessaire au sujet de ce protocole. Aucun d'entre ne peut cependant se targuer d'être complet, mais cela ne devrait guère poser de problème pour la majorité des utilisateurs. Nous avons découvert quelques sites intéressants publiés en diverses langues : **Ressources Techniques** [3] est en

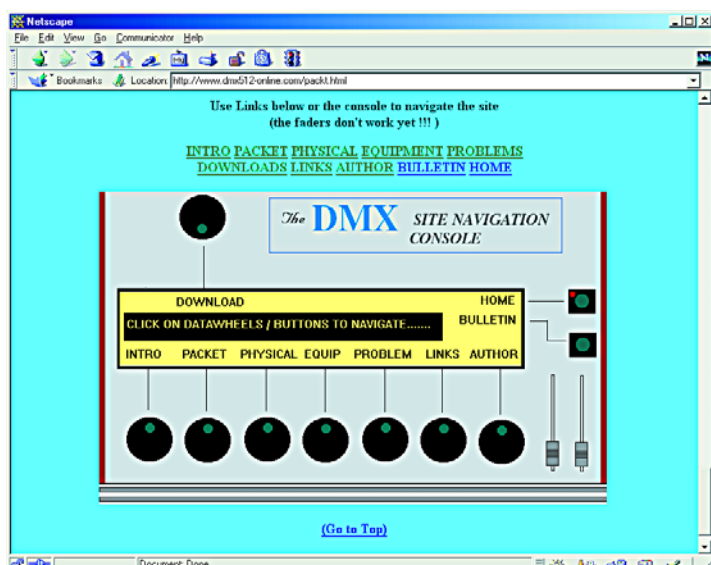


français, **DMX512.COM** [4] et **The anatomy of DMX512** [5] en anglais, **Soundlight** [5] étant en allemand.

Le site **Ujjal's DMX512 Website** [7] est un site s'intéressant à tous les aspects de DMS-512. Ce site très développé est structuré intelligemment et traite également de problèmes rencontrés avec DMX. Il propose aussi une collection importante de liens ayant trait à DMX, au nombre desquels tous les fabricants d'appareils DMX connus.

On y trouve en outre une section de

téléchargement de logiciels et matériels relatifs à DMX. Autre section intéressante, les FAQ où l'on est surpris de constater que la plupart sont relativement anciennes. Nous avons ainsi découvert 3 adresses proposant la même liste de FAQ mais avec des dates de révision différentes. Elles avaient toutes près de 5 ans. Nous vous proposons, à titre d'exemple, **DMX512 mini-FAQ** [8], dont le contenu est, en dépit de son âge, encore très utilisable (elle traite principalement de sujets de base).



S'il est dans vos intentions de vous lancer dans le développement de logiciels et/ou de matériels DMX, il vous faudra non seulement savoir ce que recouvre le protocole DMX mais également connaître quels sont les codes auxquels réagissent les appareils des différents fabricants. Nous avons trouvé 2 sites donnant un panorama intéressant à ce sujet : ESTA propose une page avec **DMX Usage Data** [9] et Interactive Technologies met à disposition une **DMX Protocol Library** [10].

On ne trouve sur Internet que très peu de développements tant au point de vue matériel que logiciel, mais les choses risquent de changer dès lors que le standard DMX devient de plus en plus populaire auprès des amateurs.

La combinaison PC et système DMX-

512 semble attirer de plus en plus de concepteurs. Il existe ainsi **AVR DMX-512 dongle** [11], un dongle AVR DMX-512 pour PC. Il s'agit en l'occurrence d'une mini-interface reposant sur un microcontrôleur de type AVR et un circuit émetteur/récepteur RS-485. Comme le laisse sous-entendre sa dénomination, ce montage se connecte au port imprimante du PC. Son logiciel d'exploitation est écrit en C. L'auteur propose sur son site tant le matériel que le logiciel au téléchargement.

Sur le site **Kristof's Webpage** [12] on trouve une autre interface pour port imprimante. Le schéma et les codes hexadécimaux sont proposés gratuitement au téléchargement.

On trouve en outre à cet endroit le schéma d'un booster DMX ainsi que le schéma et le fichier hex d'un fla-

sher DMX à 12 boutons.

Les pages **Electronic Projects** [13] de Christian Mohr proposent plusieurs projets d'électronique attrayants. Les 2 projets DMX en cours de mise au point sont une interface DMX-PC et un récepteur DMX.

Sur le site d'un autre amateur de DMX-512, Adam Davis, nous avons découvert la description d'un petit système de commande DMX réalisé à l'aide d'un PIC [14]. Ce projet aussi est encore à l'état expérimental, mais il nous paraît intéressant de le suivre pour voir comment les choses évoluent.

LPR2DMX [15] est une minuscule interface imprimante pour le PC permettant la communication DMX-512. L'interface se laisse piloter par le biais d'un pilote Linux. Le dit pilote, **DMX4Linux** est disponible à l'adresse [16]. Il supporte également, outre LPR2DMX d'autres interfaces DMX (commerciales).

(015101)

Adresses Internet :

- [1] USITT:
www.usitt.org/DMX/DMX512.htm
- [2] PLASA:
www.plasa.org
- [3] DMX512.COM:
www.dmx512.com/web/light/dmx512/index.htm
- [4] The anatomy of DMX512:
www.euro-pa.be/dmx.html
- [5] Soundlight:
www.soundlight.de/techtips/dmx512/dmx512.htm
- [6] Ressources Techniques:
http://ogloton.free.fr/dmx_512/presentation.html
- [7] Ujjal's DMX512 Website:
www.geocities.com/dmxpage/
- [8] DMX512 mini-FAQ:
www.lightresource.com/dmx512fq.html
- [9] DMX Usage Data:
www.esta.org/tsc_dmx_data/dmx-cat.htm
- [10] DMX Protocol Library:
www.interactive-online.com/library/
- [11] AVR DMX-512 dongle for PC's:
www.ele.tut.fi/~viikari
- [12] Kristof's Webpage:
<http://users.skynet.be/kristofnys/>
- [13] Electronic Projects:
www.hb.se/cmoproject/default.htm
- [14] Adam Davis:
www.ubasics.com/adam/electronics/dmx512.shtml
- [15] LPR2DMX:
www.lighting-solutions.de/products/LPR2DMX/index.html
- [16] DMX4Linux:
<http://llg.cubic.org/dmx4linux/>

