INNOVATIONS... MONTAGES FIABLES... ÉTUDES DÉTAILLÉES... ASSISTANCE LECTEUR

ELECTRONIQUE

ET LOISIRS magazine

http://www.electronique-magazine.com

n°75 SEPTEMBRE 2005

CESSEZ DE FUMER GRÂCE À ÉLECTRONIQUE LM ET SON ÉLECTROPUNCTEUR



UN CONTRÔLE DE VOLUME IR

UNE UNITÉ DE MÉMOIRE À CARTE SD

UN CONTRÔLEUR DMX SUR PORT USB

UN AMPLIFICATEUR HI-FI DE 10 WRMS SUR 8 OHMS

SOMMAIRE DÉTAILLÉ PAGE 3

L'ELECTRONIQUE POUR TOUS n°75 ELECTRONIQUE

France 4,50 € - DOM 4,50 € - CE 5,00 € - Suisse 7,00 FS - MARD 50 DH - Canada 7,50 $C

Imprimé en France / Printed in France
M 04662 - 75 - F: 4,50 €
N° 75 - SEPTEMBRE 2005

elc

la qualité au sommet

Moins de **stock** et plus d'**efficacité** avec les **nouvelles** alimentations

Alimentations redressées filtrées entièrement fermées, **IP 30**, avec **transformateur torique**, protégées, entrée **230 ou 400V**, sortie 24V DC.



ALE2402R
24V 2,5A
78,94 €

ALE2405R
24V 5A
101,06 €

ALE2410R
24V 10A
134,55 €

Prix TTC

Les **avantages** du **découpage** et du **linéaire**,
résiduelle totale < à 3mV eff., stabilisées et protégées, entrée secteur 230V avec PFC si > 70W, **IP 30**.

ALF1205
12V 5A
83,72 €



ALE1205
12V 5A
81,93 €



ALE2902M
5V 4A à 29V 2A
89,70 €



ALF2902M
5V 4A à 29V 2A
94,48 €



ALF1210
12V 10A
137,54 €



ALE1210
12V 10A
125,58 €



ALE2405
24V 5A
121,99 €

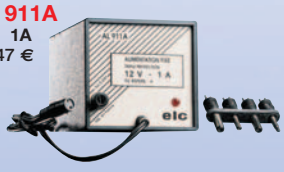


ALF2405
24V 5A
133,95 €



Alimentations **linéaires**,
stabilisées et protégées, résiduelle totale **< 1mV eff.**, secteur 230V.

AL 911A
12V 1A
39,47 €

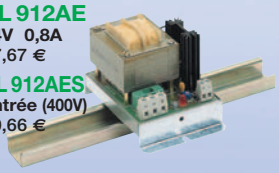


AL 911AE
12V 1A
34,68 €



AL 912AE
24V 0,8A
37,67 €

AL 912AES
entrée (400V)
40,66 €



AL 912A
24V 1A
42,46 €



AL 895A
12,5V 20A
227,24 €

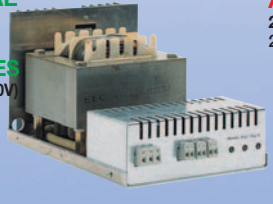


AL 895AE
12V 20A
181,79 €



AL 898AE
24V 10A
185,38 €

AL 898AES
entrée (400V)
190,16 €



AL 898A
24V 12A
215,28 €



elc

59, avenue des Romains - 74000 Annecy
Tél. 33 (0)4 50 57 30 46 - Fax 33 (0)4 50 57 45 19
En vente chez votre fournisseur de composants électroniques ou les spécialistes en appareils de mesure

Je souhaite recevoir une documentation sur :

Nom

Adresse

Ville Code postal

FRANÇOISE BAUDOUX CRÉATION GRAPHIQUE 12 - 07 - 2004

75 SOMMAIRE 75

Cessez de fumer grâce à ÉLECTRONIQUE LM (et son électropuncteur) 5



Bien que les pires malédictions soient écrites de plus en plus gros au fil des ans (comme une analogie des progrès de la tumeur qui nous envahit ?) sur chaque paquet de cigarettes (bout filtre ou sans fil...), cesser de fumer sans l'aide de contributeurs externes est plutôt difficile ! La menace ci-dessus aide à nous décider d'arrêter mais pas à nous tenir à cette décision. L'électrostimulateur, ou électropuncteur, que nous vous proposons de construire réveillera dans votre corps l'énergie nécessaire (ce que l'on appelle à tort la volonté) pour tenir bon jusqu'au sevrage et à la désintoxication définitive.

Une unité de mémoire à carte SD 11



Nous avons utilisé une carte SD pour réaliser une unité de mémoire économique gérable par protocole sériel. Ce montage peut être couplé à un PC ou bien utilisé avec des appareils «stand alone» (fonctionnant tout seuls sans ordinateur) nécessitant une capacité importante ; le programme complet des tests pour PC est disponible.

Comment programmer le module GPS Sony Ericsson GM47 ... 24

Troisième partie : programmation du microcontrôleur interne



Dans cette série d'articles, nous allons vous apprendre à programmer et à utiliser le module GSM GM47 de Sony Ericsson. Nous approfondirons la connaissance du logiciel et du matériel de ce module afin de réaliser par la suite de nombreuses applications GSM. Une grande partie de cette série sera consacrée à la programmation des microcontrôleurs présents à l'intérieur du module par des «scripts» utilisant un langage dérivé du C.

Un contrôle de volume IR 33



Ce système de réglage de volume à infrarouges concerne tout appareil audio stéréophonique. Grâce à sa distorsion harmonique très faible et à sa diaphonie de 100 dB, ce montage peut être utilisé même avec une chaîne Hi-Fi de haut de gamme.

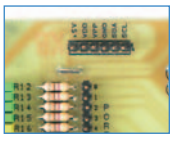
Un amplificateur Hi-Fi de 10 WRMS sur 8 ohms 38



Après que nous vous ayons récemment présenté les schémas d'une dizaine de préamplificateurs BF simples n'utilisant que deux transistors ou FET, vous avez étés nombreux à nous demander de proposer aussi un étage final de puissance simple et entièrement à transistors. Le voici. .

Comment programmer le module SitePlayer SP1 44

Quatrième partie : exemples de programmes



Dans cette série d'articles, nous allons vous apprendre à programmer et à utiliser le module SitePlayer SP1. Ce circuit intégré réalise un véritable serveur pour la Toile («Web Server»), c'est-à-dire qu'il permet d'interagir avec n'importe quel dispositif électronique à travers une page Internet normale. Nous allons donc apprendre à nous servir de ce module pour réaliser des applications nous permettant de faire communiquer sur le réseau des appareils distants en tout genre.

Un contrôleur DMX sur port USB 55



Ce système de régie lumière au protocole DMX512 comporte une interface USB et bien sûr un programme de gestion. Toutes les ressources nécessaires (DLL et routines de test) sont fournies afin que vous puissiez réaliser des programmes personnalisés en Delphi, Visual Basic.NET et C++ Builder.

Actualité 65

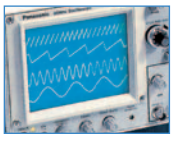
La société MICRELEC vous présente un nouveau logiciel: MULTI-PROG



Le logiciel MULTI-PROG a été créé pour faciliter la prise en main des microcontrôleurs par des élèves et des débutants en programmation. En effet, ce logiciel permet d'utiliser toutes les fonctions liées autour du microcontrôleur sans écrire une seule ligne de code. Les élèves peuvent comprendre le rôle d'un CAN sans paramétrer un seul registre: ils dessinent l'algorigramme et obtiennent le programme directement.

Apprendre l'électronique en partant de zéro 69

Quatrième partie: mesures altrenatives



Étant donné que l'oscilloscope nous permet de visualiser à l'écran n'importe quelle forme d'onde, qu'elle soit sinusoïdale, triangulaire ou carrée, il est un excellent instrument pour la mesure des tensions alternatives. Cette Leçon vous apprendra comment mesurer l'amplitude d'une onde et comment convertir les Vpp en Veff.

Les Petites Annonces 76

L'index des annonceurs se trouve page 76

Ce numéro a été envoyé à nos abonnés le 25 août 2005

Crédits Photos : Corel, Futura, Nuova, JMJ

ABONNEZ-VOUS A

ELECTRONIQUE ET LOISIRS magazine

LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS

Les projets que nous vous présentons dans ce numéro ont été développés par des bureaux d'études et contrôlés par nos soins, aussi nous vous assurons qu'ils sont tous réalisables et surtout qu'ils fonctionnent parfaitement. L'ensemble des typons des circuits imprimés ainsi que la plupart des programmes sources des microcontrôleurs utilisés sont téléchargeables sur notre site à l'adresse : www.electronique-magazine.com/ci.asp. Si vous rencontrez la moindre difficulté lors de la réalisation d'un de nos projets, vous pouvez contacter le service technique de la revue, en appelant la hot line, qui est à votre service du lundi au vendredi de 16 à 18 H au 0820 000 787 (N° INDIGO : 0,12 € / MM), ou par mail à info@electronique-magazine.com

Le bon d'abonnement ELECTRONIQUE ET LOISIRS magazine LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS se trouve page 78

LES KITS DU MOIS... LES KITS DU MOIS

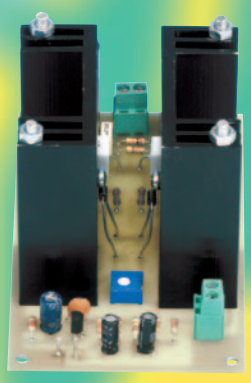
CESSEZ DE FUMER GRÂCE À ÉLECTRONIQUE LM ET SON ÉLECTROPUNCTEUR



Bien que les pires malédictions soient écrites de plus en plus gros au fil des ans (comme une analogie des progrès de la tumeur qui nous envahit ?) sur chaque paquet de cigarettes (bout filtre ou sans), cesser de fumer sans l'aide de contributeurs externes est plutôt difficile ! La menace ci-dessus aide à nous décider d'arrêter mais pas à nous tenir à cette décision. L'électrostimulateur, ou électro-puncteur, que nous vous proposons de construire réveillera dans votre corps l'énergie nécessaire (ce que l'on appelle à tort la volonté) pour tenir bon jusqu'au sevrage et à la désintoxication définitive.

EN1621.... Kit complet avec son boitier.....24,00 €

UN AMPLIFICATEUR HI-FI DE 10 WRMS / 8 OHMS



Alimentation simple: 30 à 33 V environ

Courant de repos: 70 mA

Courant à la puissance max.: 450 mA

Puissance max. sur 8 ohms: 12 WRMS

B.P à –3 dB: 10 Hz à 40 kHz

Amplitude max. signal d'entrée: 1,6 Vpp

Distorsion harmonique: 0,4% environ

EN1616.... Kit complet sans boitier.....24,00 €
EN1617 .... Kit alimentation.....28,00 €

UN CONTRÔLE DE VOLUME IR



Ce système de réglage de volume à infrarouges concerne tout appareil audio stéréophonique. Grâce à sa distorsion harmonique très faible et à sa diaphonie de 100 dB, ce montage peut être utilisé même avec une chaîne Hi-Fi de haut de gamme.

MEV164... Kit complet avec son boitier..... 19,95 €

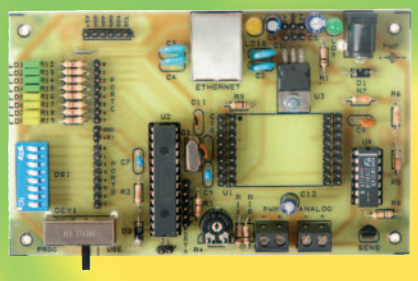
UN CONTRÔLEUR DMX SUR PORT USB



Ce système de régie lumière au protocole DMX512 comporte une interface USB et bien sûr un programme de gestion. Toutes les ressources nécessaires (DLL et routines de test) sont fournies afin que vous puissiez réaliser des programmes personnalisés en Delphi, Visual Basic.NET et C++ Builder.

EV8062.... Kit régie de lumière DMX via USB ..... 81,00 €

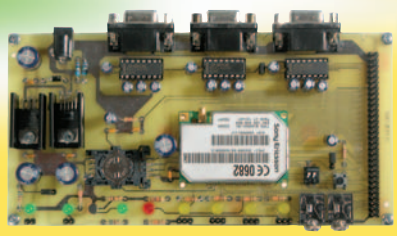
PROGRAMMATEUR / PLATINE D'EXPÉRIMENTATION POUR MODULE SP1



Associé à sa documentation, cette platine d'expérimentation pour module SP1 vous permet de mettre au point vos programmes pour votre server web SP1. Dim : 75 x 130 mm – Alimentation 12 VDC.

ET497..... Kit programmateur de SP1 sans module SP1.58,00 €
SP1..... Module SP1 seul.....58,00 €

COMMENT PROGRAMMER LE GPS SONY ERICSSON GM47



Cette platine d'expérimentation vous permet d'apprendre à programmer et à utiliser le module GSM GM47 de Sony Ericsson. Une notice approfondie du logiciel et du matériel de ce module vous permettra de réaliser de nombreuses applications GSM. Dim : 180 x 110 mm – Alimentation 12 VDC.

ET502..... Kit platine d'expérimentation sans le GM47 . 70,00 €
GM47 ..... Module GM47 seul ..... 190,00 €

COMELEC CD 908 - 13720 BELCODENE
www.comelec.fr
Tél.: 04 42 70 63 90 Fax: 04 42 70 63 95
DEMANDEZ NOTRE CATALOGUE 96 PAGES ILLUSTRÉES AVEC LES CARACTÉRISTIQUES DE TOUS LES KITS
Epéditions dans toute la France.Moins de 5 Kg : port 8,40 €. Règlement à la commande par chèque, mandat ou CB. Bons administratifs acceptés.
De nombreux kits sont disponibles, envoyez nous votre adresse et cinq timbres, nous vous ferons parvenir notre catalogue général de 96 pages.

PASSEZ VOS COMMANDES DIRECTEMENT SUR NOTRE SITE : www.comelec.fr

Photos non contractuelles. Publicité valable pour le mois de parution. Prix exprimés en euro toutes taxes comprises. Sauf erreurs typographiques ou omissions.

Comelec 09 / 2005

Cessez de fumer grâce à ÉLECTRONIQUE LM

(et son électropuncteur)

Bien que les pires malédictions soient écrites de plus en plus gros au fil des ans (comme une analogie des progrès de la tumeur qui nous envahit ?) sur chaque paquet de cigarettes (bout filtre ou sans fil…), cesser de fumer sans l'aide de contributeurs externes est plutôt difficile ! La menace ci-dessus aide à nous décider d'arrêter mais pas à nous tenir à cette décision. L'électrostimulateur, ou électropuncteur, que nous vous proposons de construire réveillera dans votre corps l'énergie nécessaire (ce que l'on appelle à tort la volonté) pour tenir bon jusqu'au sevrage et à la désintoxication définitive.



L'origine de ce petit appareil (voir figure 1) fort efficace remonte à la découverte, au milieu des années 80, par un médecin chinois, de la fréquence idéale d'électrostimulation à appliquer aux points d'acupuncture (oreilles et nez, voir figures 8 et 9) pour éliminer la dépendance tabagique : entre 5,4 et 5,6 Hz. Efficace ? On peut le dire : 80% des personnes des deux sexes cessent de fumer dans les dix à douze jours. Miracle, non : les 20% restants ont besoin de plus de deux semaines pour arrêter (le degré de motivation, l'ancienneté de la dépendance et les facteurs psychologiques personnels sont la cause de ces différences). Aussi, ne croyez pas vous arrêter après la première application (si ça vous arrive, tant mieux) et ne vous imaginez pas, en décidant d'arrêter et en commençant le traitement, que vous ferez partie des 80% : en effet, il vous faudra peut-être deux semaines ; mais entre temps vous diminuerez le nombre de cigarettes que vous fumez par jour et cela vous fera le plus grand bien, tout en vous encourageant à poursuivre.

Le schéma électrique de l'électropuncteur

Comme le montre le schéma électrique (figure 3), le circuit utilise le très fameux circuit intégré NE555 (IC1) monté en multivibrateur astable : de sa broche 3 sortent des impulsions d'onde carrée ayant une durée T/on d'environ 1,47 ms et une durée T/off d'environ 180 ms (voir figure 2). Pour connaître la fréquence F en Hz produite par ces impulsions, on utilise la formule :

F = 1 000 : (T/on + T/off),

si nous insérons les valeurs précédentes dans cette formule nous obtenons :

F = 1 000 : (1,47 + 180) = 5,51 Hz.

Même si l'on tient compte des tolérances des condensateurs et des résistances, la fréquence sera tout de même située dans l'intervalle requis de 5,45 à 5,55 Hz. Cette fréquence, prélevée sur la broche 3 de IC1, est acheminée vers la LED DL1 laquelle, en clignotant, indique que l'électropuncteur est alimenté et que, par conséquent, la base du darlington TR2, un ZTX601, est elle-même alimentée (ce darlington pilote le primaire du transformateur de sortie TM1).

Ce dernier remplit la même fonction que la bobine HT d'une voiture à essence (qui produit les étincelles d'allumage entre les électrodes des bougies). Du secondaire de TM1 sortent de larges impulsions positives suivies d'impulsions négatives en revanche très étroites (voir figure 2), nécessaires pour éviter que ne se produise le phénomène de l'électrolyse du sang. A la sortie du primaire, le NPN TR1, un ZTX653 cette fois, fait varier l'amplitude des impulsions négatives, simplement en modifiant la polarisation de sa base au moyen du potentiomètre R3 (une molette de réglage est fixée sur son axe). Il faudra, lors de l'utilisation de l'appareil, tourner cette molette jusqu'à ressentir, au point de contact de "l'aiguille" (ou pointe de touche) avec l'oreille ou le nez, un picotement supportable.

La totalité de ce circuit, au demeurant fort simple, est alimenté par une pile (ou batterie rechargeable) 6F22 de 9 V. La consommation moyenne étant de 20 mA, une pile alcaline devrait durer quelque 500 heures. Compter pour chaque application une minute en moyenne.

La réalisation pratique de l'électropuncteur

Pour réaliser cet électropuncteur, il vous faut le petit circuit imprimé EN1621 : c'est un double face à trous métallisés dont la figure 6b-1 et 2 donne les dessins à l'échelle 1. Si vous observez bien les figures 6a, 5 et 7 et la liste des composants, vous n'aurez aucune difficulté à y monter tous les composants. Accordez, comme d'habitude beaucoup d'attention aux soudures du support du NE555 (ni court-circuit entre pistes ou pastilles ni



Figure 1 : Photo d'un des prototypes de l'électropuncteur anti-tabagique qui vous permettra de cesser de fumer.

soudure froide collée) que vous monterez en premier. Attention en outre à l'orientation (et donc à la polarité) des condensateurs électrolytiques, des diodes, des transistor et darlington, de la LED et de IC1 (dont le repère-détrompeur en U doit être tourné vers le transformateur).

Le montage dans le boîtier

Le potentiomètre est un modèle à interrupteur intégré pour circuit imprimé, il est soudé au ci par ses cinq broches et son "axe", situé à l'arrière, reçoit une molette de commande (pour son passage, il faudra pratiquer un évidement rectangulaire dans la partie inférieure du boîtier plastique). A côté, faites un trou pour le passage de la LED. "L'aiguille", ou pointe de touche, est constituée d'un morceau de fil de cuivre ou de laiton de 3 mm de diamètre et de 25 mm de long au bout arrondi (comme l'électrode d'une ancienne fiche secteur) : elle est soudée solidement au picot de sortie (ensuite elle sortira du boîtier par les évidements semi circulaires pratiqués dans les deux coques de celui-ci à l'aide d'une petite lime queue de rat). Quant au point de masse (qui devra faire un excellent contact avec votre doigt), il est formé par un "clou" de cuivre ou de laiton à tête large et arrondie : la tête restant accessible à votre doigt à l'extérieur de la coque inférieure du boîtier, la pointe devra traverser celle-ci et le trou du ci, près de R7, où elle sera

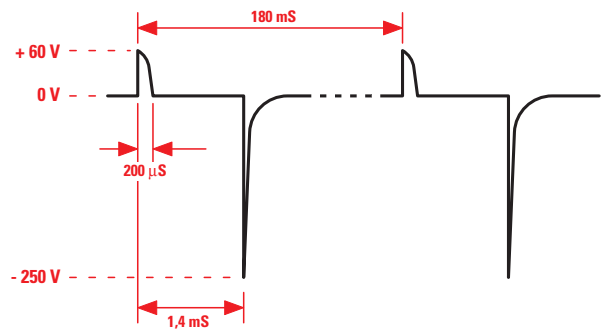


Figure 2 : Du secondaire du transformateur TM1 (voir figure 3) sortent des impulsions négatives très étroites d'une durée de 1,47 ms (temps T/on) suivi d'une pause de 180 ms (temps T/off). Les impulsions positives précédant les négatives servent à éviter d'éventuels effets d'électrolyse du sang.

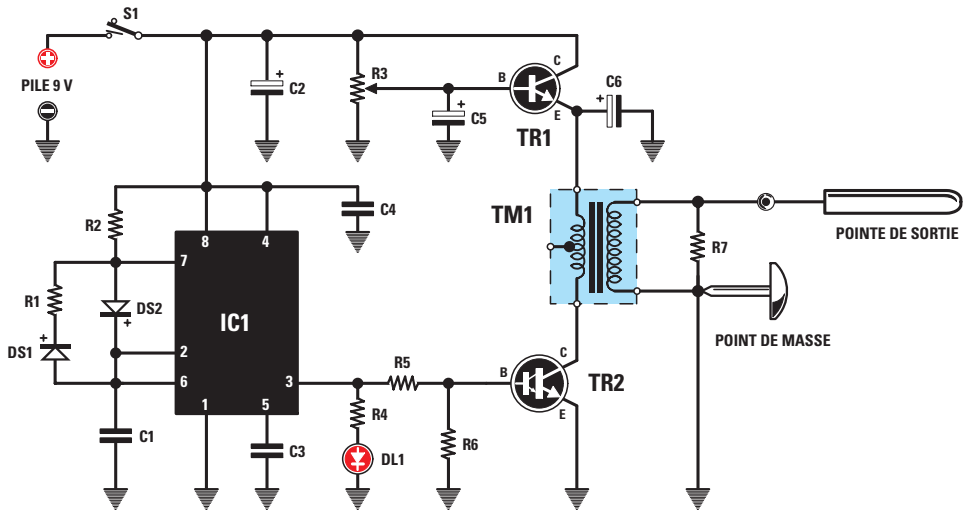


Figure 3 : Schéma électrique de l'électropuncteur. Peu de composants vous seront nécessaires (voir liste ci-après).

soudée. On pourra aussi remplacer la tête du clou par une petite plaque fine de cuivre ou de laiton (collée au dos du boîtier), sans oublier de la relier au trou de masse, près de R7, au moyen d'un morceau de fil de cuivre nu. Les figures citées ne vous laisseront aucun doute sur la manière de procéder.

Le mode d'emploi de l'électropuncteur

Pour se débarrasser de la dépendance tabagique, il faut effectuer les opérations suivantes :

- Appuyez la pointe de touche de l'électropuncteur, pendant environ une minute, sur une des zones A ou B de l'oreille ou C du nez (voir figures 8 et 9) de manière à les stimuler avec les impulsions négatives.

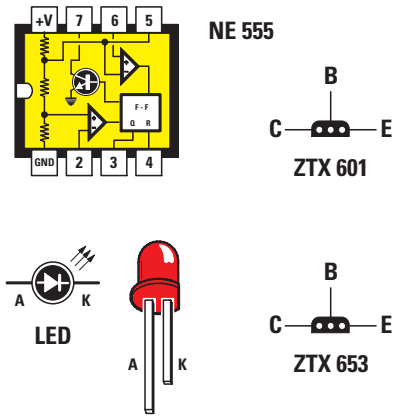


Figure 4 : Brochages du NE555 vu de dessus et des NPN ZTX653 et ZTX601 vus de dessous (la partie arrondie de ces derniers est à peine visible). Celui de la LED est vu de face en contre plongée.

Note : vous devez d'abord poser votre doigt (par exemple l'index de la main tenant l'appareil) sur le point de masse extérieur (tête du clou ou plaque) et ensuite appliquer la pointe de touche sur la zone choisie. Si, quand vous tournez la molette du potentiomètre à fond, vous ne ressentez aucun picotement, vous devez améliorer le contact du doigt avec la masse (problème de résistivité de la peau) en humectant ce doigt (avec un peu d'eau). Dans ce cas, baissez préalablement le niveau du potentiomètre, mouillez le doigt, posez-le sur le point de masse, appliquez la pointe de touche sur la zone choisie et augmentez le niveau du potentiomètre, en tournant la molette, jusqu'à l'obtention du picotement supportable.

- Vous pouvez stimuler deux des trois zones A, B, C pendant une minute environ : chacun découvrira en pratiquant laquelle, ou lesquelles, de ces zones est la, ou sont les, plus sensible(s).

- Tournez la molette du potentiomètre R3 jusqu'à ressentir un léger picotement non douloureux.

- Comme le montrent les figures 8 et 9, après une minute de traitement (oreille ou nez), éteignez l'appareil et buvez tout de suite un demi verre d'eau.

- Si vous ne parvenez pas à vous arrêter de fumer dans un délai d'une semaine, cela signifie que votre organisme est gravement intoxiqué par la nicotine : dans ce cas, traitez les trois zones A, B et C (comme on a deux oreilles, cela fait cinq zones A, B, C et A', B') pendant une durée d'environ 15-20 secondes chacune. Vous pouvez toutefois augmenter les durées sus indiquées, car cette thérapie ne présente pas de contre indication.

- Chaque fois que vous ressentez le désir d'allumer une cigarette, traitez-vous avec l'électropuncteur sur une des zones A, B, A', B' ou C puis, après une minute, éteignez l'appareil et buvez un demi verre d'eau. Cela doit vous aider à renoncer à cette cigarette.

- Sinon, stimulez une autre zone (que celle que vous venez de traiter)

Liste des composants

R1 .... 270 k
R2 .... 2,2 k
R3 .... 47 k potentiomètre lin.
R4 220
R5 .... 4,7 k
R6 .... 10 k
R7 .... 100 k
C1..... 1 µF polyester
C2..... 220 µF électrolytique
C3..... 100 nF polyester
C4..... 100 nF polyester
C5..... 10 µF électrolytique
C6..... 100 µF électrolytique
DS1.. 1N4150
DS2.. 1N4150
DL1 .. LED
TR1... NPN ZTX653
TR2... NPN ZTX601
IC1.... NE555
TM1.. transformateur mod. TM1621
S1..... interrupteur sur R3

Note : toutes les résistances sont des 1/8 de W.

pendant une minute encore puis éteignez l'appareil et buvez un demi verre d'eau.

- Quant vous ne ressentirez plus le désir d'allumer une cigarette pendant une heure environ, cela signifiera que la thérapie commence à produire des effets positifs.

- Répétez ces opérations chaque fois que vous ressentirez le désir de fumer et un jour vous constaterez que vous ne fumez plus que la moitié du nombre de cigarettes que vous fumiez auparavant. Ne vous contentez surtout pas de ce premier résultat ! Il est en effet plus facile de ne plus fumer du tout (et définitivement) que de fumer moins (car on augmente progressivement le nombre de cigarettes sans s'en rendre compte...).

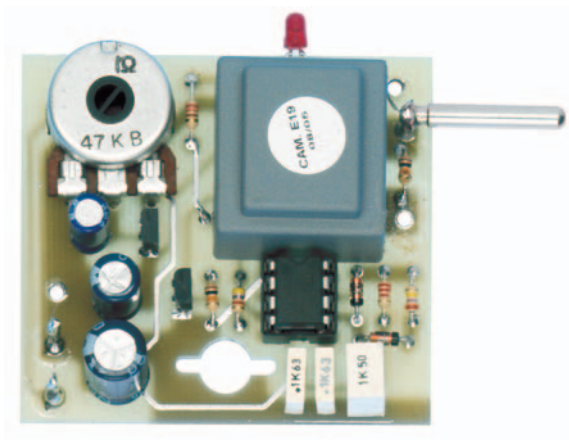


Figure 5 : Photo d'un des prototypes de la platine de l'électropuncteur prêt à être installé dans son boîtier. En haut à droite, la pointe sortant du transformateur TM1 est destinée à être mise en contact avec la zone à stimuler (voir figures 8 et 9, les points à stimuler).

- Quand vous arriverez à une ou deux cigarettes par jour, vous aurez déjà atteint un bon résultat et vous verrez qu'au cours de la semaine suivante vous fumerez encore moins.

- Vous devez maintenant vous convaincre vous-même (par un travail de raisonnement méthodique sur vous-même) que si vous avez réussi à parvenir à ne fumer plus qu'une cigarette par jour, vous pouvez éliminer cette ultime cigarette et passer toute la semaine suivante sans fumer aucune cigarette.

- Au cours de cette première semaine de liberté retrouvée (car c'est bien de cela qu'il s'agit !), afin d'éliminer les tensions nerveuses dues au sevrage de nicotine, pratiquez des activités physiques comme le jogging, ou même la marche, la promenade ou simplement une activité manuelle, etc. et répétez-vous mentalement (ou même à haute voix quand vous êtes seul) cette phrase (ou une phrase de ce type) : "puisque j'ai réussi à respirer de l'air frais et pur pendant une semaine, je peux continuer à le faire

868 pages, tout en couleurs



Envoi contre 10 timbres-poste (au tarif "lettre" en vigueur)

NOUVEAU

Catalogue **Général**

Selectronic
L'UNIVERS ELECTRONIQUE

Le CHOIX • La QUALITÉ • Le SERVICE

Connectique • Electricité
Outillage • Librairie technique
Appareils de mesure
Robotique • Etc.

Coupon à retourner à : Selectronic B.P 10050 • 59891 LILLE Cedex 9

❑ OUI, je désire recevoir le **Catalogue Général 2006 Selectronic** à l'adresse suivante (ci-joint 10 timbres-poste au tarif "LETTRE" en vigueur) : ELM

Mr. / Mme : ..... **Tél :** .....

N° : ..... **Rue :** .....

Ville : ..... **Code postal :** .....

"Conformément à la loi informatique et libertés n° 78.17 du 6 janvier 1978, Vous disposez d'un droit d'accès et de rectification aux données vous concernant"

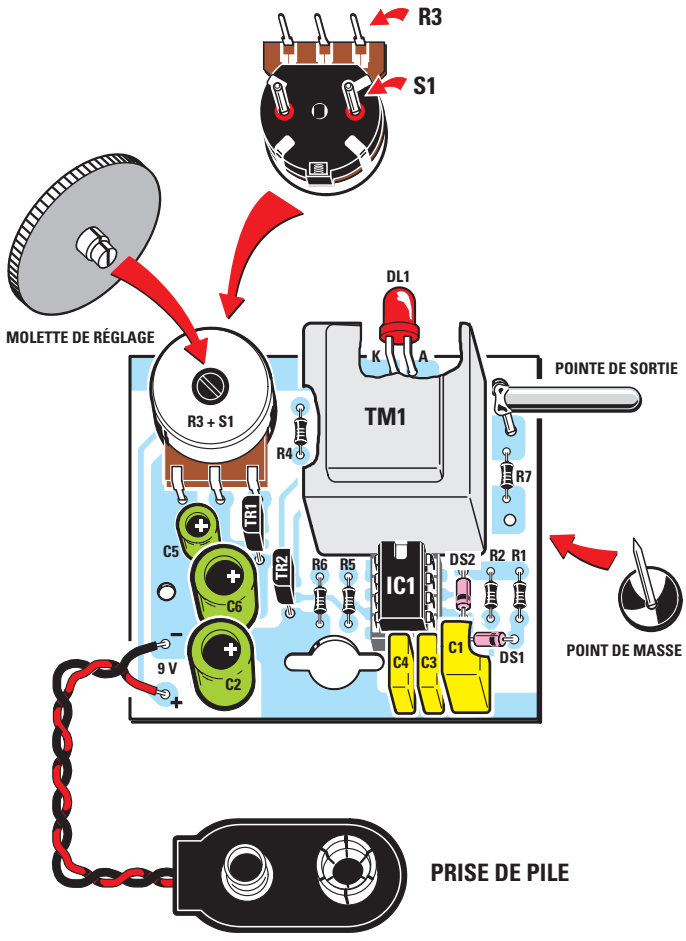


Figure 6a : Schéma d'implantation des composants de l'électropuncteur. Au moment d'insérer TR1 et le darlington TR2, orientez leur partie plate vers la droite. La pointe de sortie est à souder sur le picot auquel R7 aboutit. Le "clou" de masse est à insérer dans le trou près de R7 après avoir installé la platine dans son boîtier plastique (voir figures 7 et 1).

pendant une nouvelle semaine" (Emile Coué, pharmacien nancéen et pionnier de l'autosuggestion thérapeutique, préconisait cette méthode mentale de cure et conseillait de n'utiliser que des formules positives, donc aucune tournure négative, aucun doute non plus).

- Après la deuxième semaine de liberté et d'air pur, atteindre la troisième semaine sera plus facile et plus facile encore d'arriver à la quatrième. Votre objectif devient maintenant de passer tout un mois libre et respirant de l'air pur.

- Vous dépasserez ainsi un deuxième mois puis un troisième et peu à peu votre organisme se désaccoutumera de la nicotine et de l'habitude d'allumer une cigarette et d'en tirer de la fumée. Vous vous serez désintoxiqué, libéré d'une assuétude (les Anglais disent "addiction") morbide. Jusqu'à la fin de votre vie vous vous en féliciterez !

Les conséquences seront tout de suite des plus positives :

- Vous respirerez mieux, serez en bien meilleure forme et aurez bien meilleur moral (car tout cela va de pair, bien sûr) : finie la toux sèche opiniâtre résistant à tout médicament.

- Vous vous apercevrez que, certes, la cigarette vous aidait dans bien des cas (à vous concentrer, à résister à la fatigue ponctuelle, au sommeil, etc.), mais qu'elle avait finalement abaissé le niveau général de votre vitalité (votre potentiel vital) et que, ce potentiel de vie heureuse, vous êtes en train de le recouvrer et que vous n'avez plus nul besoin de porter cet appendice nauséabond à vos lèvres pour être bien dans votre peau !

- Vous ferez d'importantes économies (inutile de m'étendre sur ce chapitre, voir le prix de votre dernier paquet acheté)...

- Si vous craignez de grossir après le sevrage tabagique, rassurez-vous. D'une part dites-vous que le pire est bien de fumer et d'avoir du goudron dans les poumons et de la nicotine dans le sang. D'autre part votre méde-

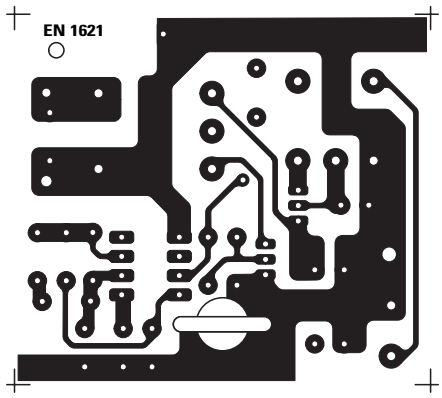


Figure 6b-1 : Dessin, à l'échelle 1, du circuit imprimé double face à trous métallisés de l'électropuncteur, côté soudures.

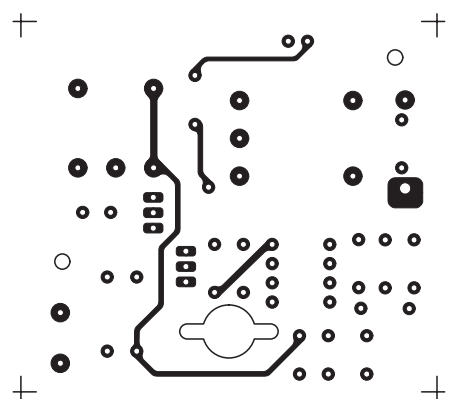


Figure 6b-2 : Dessin, à l'échelle 1, du circuit imprimé double face à trous métallisés de l'électropuncteur, côté composants.



Figure 7 : Photo d'un des prototypes de la platine de l'électropuncteur monté dans son boîtier plastique (couvercle déposé). La pile est une 6F22 de 9 V (ce peut être, bien sûr, une batterie rechargeable). Pour le passage de "l'aiguille" d'électropuncture, pratiquer dans la partie inférieure du boîtier un évidement en demi lune à l'aide d'une toute petite lime "queue de rat". Pour enfoncer le "clou" de masse (voir figure 6a), faites dans le panneau arrière de ce boîtier un tout petit trou d'un mm de diamètre. Afin d'obtenir un meilleur contact entre votre doigt et le point de masse que constitue la "tête" de ce clou, vous pouvez remplacer ce dernier par une petite plaque de laiton collée sur le panneau arrière (et bien sûr reliée à la masse du circuit imprimé par soudure).

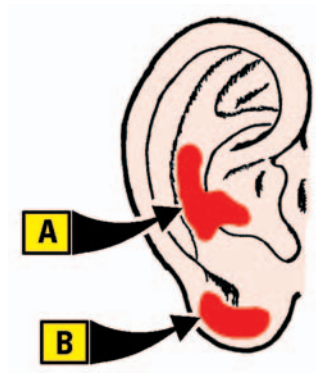


Figure 8 : Pour utiliser cet électropuncteur, appuyez le doigt sur le "clou" (ou la plaque de laiton) de masse puis mettez l'aiguille (ou pointe de touche) en contact avec une des zones de l'oreille A ou B (ici en rouge). Puis agissez sur la molette du potentiomètre R3 jusqu'à ressentir un très léger et supportable picotement. Au bout d'une minute, vous pouvez éteindre l'appareil, toujours avec la molette : buvez alors tout de suite un demi verre d'eau.

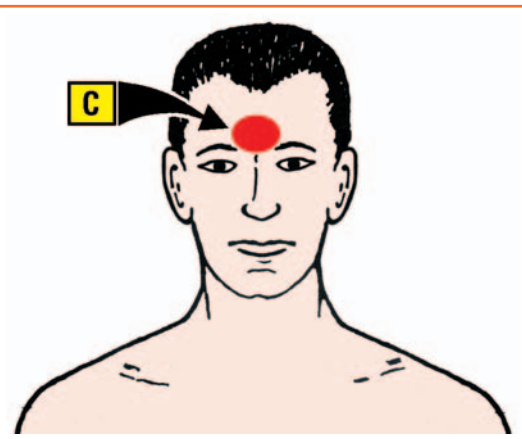


Figure 9 : L'autre point que vous pouvez stimuler également est situé à la naissance du nez et à la base du front au milieu (en rouge ici, point C). Si vous ne ressentez aucun picotement quand vous posez la pointe de l'appareil en A, B ou C, alors que le potentiomètre est tourné à fond, améliorez le contact du doigt avec la masse. Pour cela mouillez préalablement le doigt. Après chaque application, n'oubliez pas de boire un demi verre d'eau.

cin, surtout s'il est un peu orienté homéopathie (et ils sont nombreux à l'être aujourd'hui), vous prescrira un ou deux types de granules homéopathiques permettant d'éliminer ce problème (à défaut votre pharmacien lui-même pourra vous indiquer les deux tubes à prendre et la posologie, c'est bon marché et sans danger).

Etant une revue d'électronique, nous n'avons pas le droit de vous indiquer ces médicaments, même s'ils sont en vente libre sans ordonnance, mais ils sont très connus. Enfin, maintenant que vous avez découvert le bonheur de vous libérer par vous-même, par votre propre volonté (avec le petit coup de pouce de votre électropuncteur LM, que vous aurez monté vous-même), vous allez y prendre goût : alors, pourquoi ne pas continuer et vous mettre (si ce n'est déjà le cas) à une alimentation rationnelle permettant de ne pas prendre d'embonpoint ? En effet le tabagisme est souvent destiné à combler une carence physique et/ou psychologique, vous le savez. Eh bien la nourriture aussi ! Et souvent on se jette sur elle pour compenser quelque chose, pour oublier, pour résoudre absurdement un problème que l'on ne résout pourtant pas. Une alimentation rationnelle, diététique (ne parlons surtout pas de régime) permet de pallier de tels errements... et la liberté est à la clé !

Conclusion

Heureux (nous aussi) de vous avoir éventuellement (re)donné l'espoir de recouvrer une respiration libre, nous vous souhaitons une bonne réalisation et...bienvenue au club des non-fumeurs.

Comment construire ce montage ?

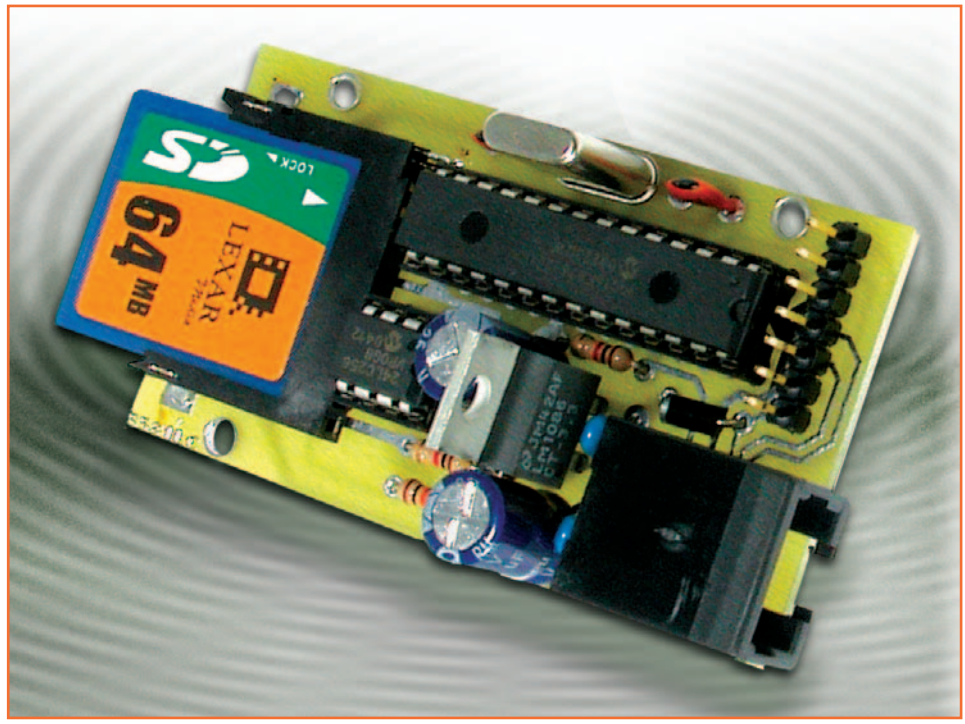
Tout le matériel nécessaire pour construire cet électropuncteur EN1621 est disponible chez certains de nos annonceurs. Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp. ◆

ET581

Une unité de mémoire à carte SD

Nous avons utilisé une carte SD pour réaliser une unité de mémoire économique gérable par protocole sériel. Ce montage peut être couplé à un PC ou bien utilisé avec des appareils “stand alone” (fonctionnant tout seuls sans ordinateur) nécessitant une capacité importante ; le programme complet des tests pour PC est disponible.



L’idée de départ de ce montage est de réaliser une interface permettant d’utiliser une SD-Card de la même manière qu’on le ferait d’une EEPROM (laquelle est bien plus limitée !). En fait, on a voulu créer un système qui permette de remplacer les dispositifs de mémorisation insérés dans les autres montages afin de les doter d’une capacité de mémoire très élevée, sans pour autant avoir à retoucher le programme résidant dans le PIC. Les possibilités d’application sont très vastes, des navigateurs GPS aux “data logger” (acquisitions de données), en passant par toutes les platines qui ont besoin pour fonctionner de manipuler des informations pour les réélaborer ensuite. La quantité d’espace utilisable d’une carte SD est tellement grande qu’elle permet d’effectuer des modifications aux montages existants simplement par insertion de données d’abord écartées par manque de capacité. Par exemple, un navigateur, en plus des données de latitude et longitude peut donner la vitesse ou l’altitude. Pensez qu’une seule carte SD de 64 Mo équivaut, en terme de capacité mémorielle (stockage de données), à environ deux mille EEPROM de type 24LC256. En outre, nous avons tout fait pour rendre le système SD facilement intégrable dans un autre appareil en utilisant un protocole de communication RS232 standard, certes aujourd’hui dépassé mais stable et facile à mettre en œuvre, même dans des montages anciens que vous doterez ainsi d’une aire de mémoire énorme et bien peu coûteuse, tout cela sans nécessairement connaître le protocole de gestion de la SD. Le résultat escompté a été atteint grâce à l’utilisation d’un PIC16F876 et de quelques rares composants périphériques. La difficulté majeure a consisté dans le développement du programme résident : comme le montre la figure 5, fournissant douze “listings” (!), ce programme traduit et exécute les commandes passées par le port sériel d’un PC. Le montage est assorti d’un logiciel en Delphi, permettant de tester les commandes de lecture et d’écriture sur le vif en utilisant la même syntaxe que celle installée dans le programme résident pour la communication

avec notre interface.

Le schéma électrique

Comme le montre la figure 1, le schéma électrique est assez simple: le PIC16F876 travaille avec une tension de 3,3 V, ce qui économise un circuit de conversion des signaux. Les cartes SD sont construites de telle manière qu'elles fonctionnent très bien entre 2,7 et 3,6 V. Pour la stabilisation de la tension, nous avons utilisé un régulateur courant et économique, le LM1086. Pour relier la carte SD au circuit, nous avons adopté la configuration permettant de mettre à profit le mode SPI pour communiquer avec le contrôleur intégré. Il s'agit d'un protocole de communication utilisant seulement trois lignes, une en entrée, une en sortie et une pour le signal d'horloge. S'y ajoute une ligne de service "Chip-Select", permettant de lancer et d'achever les transactions: nous y reviendrons dans de futurs articles et cela donnera lieu à des réalisations encore plus sophistiquées!

Vous avez sans doute noté la présence d'une FRAM FM24C64: à quoi peut bien servir un circuit de ce type? Dans les SD-Card il n'est pas possible d'écrire les données un octet à la fois, comme c'est le cas pour les EEPROM, mais seulement par groupes de 512 octets appelés "blocs". Il a fallu créer une aire de mémorisation temporaire que nous avons baptisée "aire d'échange" en tirant parti du premier secteur de la FRAM. Ainsi, nous utilisons cette FRAM comme une sorte de "buffer" (tampon) entre les diverses opérations de lecture et d'écriture. Nous avons fait en sorte que la SD soit vue de l'extérieur non comme une seule aire de 512 octets mais comme une énorme matrice d'octet. C'est au logiciel que revient de gérer la chose de manière transparente. Naturellement, les temps d'écriture d'un seul octet seront plus longs, mais cependant incomparablement plus brefs qu'avec la plupart des EEPROM du commerce. Le retard est dû au fait que le programme résident doit lire le bloc de 512 octets inséré dans la FRAM, mettre à jour l'octet correspondant et récrire le bloc sur la carte. Tout cela, bien qu'un peu complexe à organiser, permet de rendre le système pleinement compatible avec tous les dispositifs envoyant un par un les octets d'information. Nous avons prévu une ligne de sorties du PIC (PORTA RA2), mise à l'état logique haut quand le système est occupé à exécuter une commande déjà envoyée, de manière à éviter tout problème "d'overloading". De même la ligne "Errore" permet d'être informé qu'une opération n'a pas abouti. Enfin, une dernière ligne de "reset" a été prévue

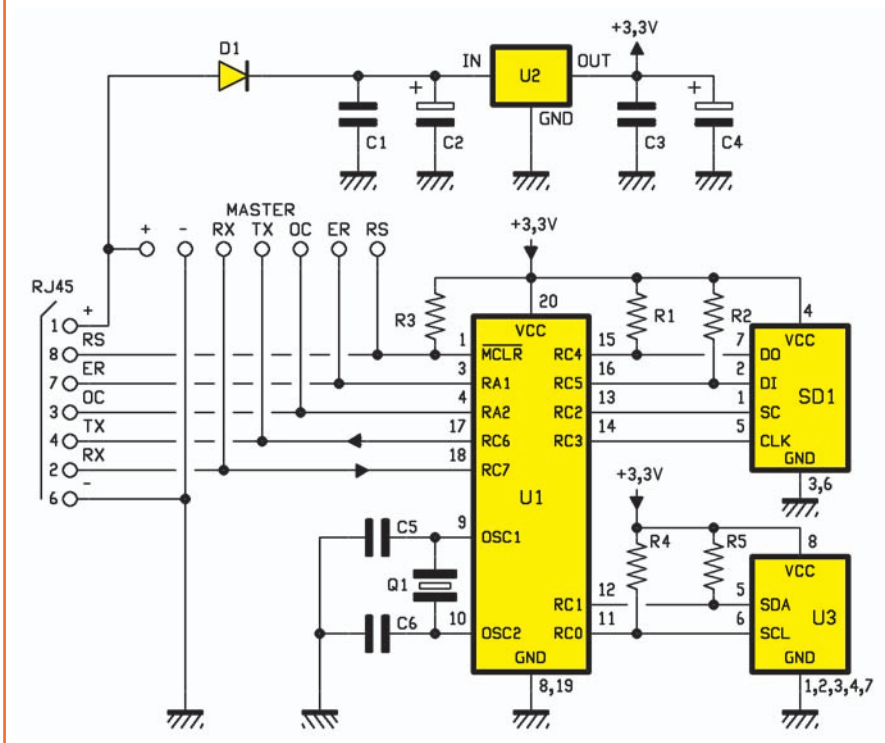


Figure 1: Schéma électrique de l'unité de mémoire.

pour pouvoir forcer un "reset" physique du circuit et donc de la SD-Card.

La réalisation pratique

La réalisation pratique de cette unité de mémoire SD ne présente pas de difficulté. La platine est constituée d'un circuit imprimé simple face, dont la figure 2b donne le dessin à l'échelle 1. Insérez et soudez tous les composants (comme le montrent les figures 2a et 3), en commençant par les supports de circuits intégrés et en terminant par les "périphériques": le porte carte SD, la barrette et le connecteur RJ45. Le régulateur U2 est monté debout sans dissipateur, semelle métallique vers C2, le quartz aussi est monté debout. Attention à l'orientation des composants polarisés: ci, diode et électrolytiques.

En ce qui concerne l'interface pour attaquer le port sériel de l'ordinateur, nous avons prévu d'utiliser la ET475: cette petite platine est constituée essentiellement d'un MAX232, très courant et assez simple à utiliser (il réalise la conversion des niveaux logiques des signaux provenant du PIC en standard RS232, +12 V / –12 V). Grâce à cela, nous pouvons relier les lignes du micro à un port du PC ou à tout autre dispositif RS232. Pour alimenter l'interface, nous réutilisons les broches 1 et 5 du RJ45 monté sur la ET475.

Le programme résident

Comme dans les autres montages avec microcontrôleur, la difficulté principale est constituée par l'écriture du programme résident conservé dans la mémoire du PIC. Nous avons essayé de rendre le "listing" (voir figure 5) le plus lisible possible en le fractionnant en douze parties ou blocs fonctionnels. Voyons tout d'abord les définitions des ports et des variables. Toutes les broches du PIC ayant une fonction de communication ou de contrôle ont été redéfinies à travers un nom significatif ("listing" 1). Les lignes d'E / S de la carte sont appelées respectivement SDI ("Serial Data Input") et SDO ("Serial Data Output"). De même celles de l'horloge deviennent SCK ("Signal Clock").

Prêtons attention aux variables CMD, IND1, IND0, LOC512, DATO. Elles représentent les champs fondamentaux utilisés pour envoyer des ordres à notre interface. Nous avons en effet décidé que chaque commande envoyée se compose de 8 octets définis dans le Tableau 1. Quand nous envoyons une séquence de ce genre, nous transmettons au PC non seulement la valeur que nous voulons écrire mais aussi le point

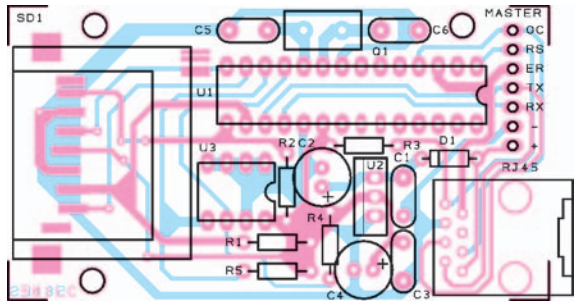


Figure 2a : Schéma d'implantation des composants de l'unité de mémoire.

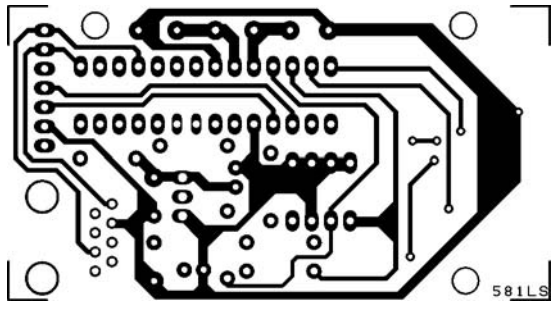


Figure 2b : Dessin, à l'échelle 1, du circuit imprimé de l'unité de mémoire.

Liste des composants

R1 ..... 1 k
R2 ..... 1 k
R3 ..... 1 k
R4 ..... 10 k
R5 ..... 10 k
C1..... 100 nF multicouche
C2..... 220 µF 25 V électrolytique
C3..... 100 nF multicouche
C4..... 220 µF 25 V électrolytique
C5..... 22 pF céramique
C6..... 22 pF céramique
D1 ..... 1N4007
U1..... PIC16F876-EF581
U2..... LM1086
U3..... FM24C64-P
Q1 ..... quartz 10 MHz
SD1 ... connecteur SD-card

Divers :

1 support 2 x 14
1 support 2 x 4
1 barrette mâle 7 pôles
1 connecteur RJ45

La barrette à 7 pôles permet la connexion à un appareil maître pouvant utiliser notre circuit comme support de mémoire.

Sauf spécification contraire, toutes les résistances sont des 1/4 W à 5 %.

exact où nous voulons qu'elle se place. Si nous considérons la triade IND1, IND0 et LOC512, nous voyons qu'elle rassemble toutes les informations nécessaires et suffisantes pour adresser l'espace entier de mémorisation de la carte. Par exemple, si nous voulons écrire la valeur 114 (72h) dans la septième adresse du troisième bloc de la carte, nous enverrons une commande (en hexadécimal) du type :

24 0000 0400 0003 72,
soit en décimal :
24 00000 01024 00003 114.

Il faut faire très attention à l'adressage des blocs car le contrôleur intégré sur la carte est intraitable sur ce point ! IND0 doit toujours être un multiple de 512 sous peine d'interruption du processus d'écriture avec message du type "out of range". Naturellement vous devrez augmenter les 128 secteurs de 1 IND1. Avec PICBasic vous ne pouvez en effet gérer une valeur de 32 bits mais seulement des mots de 16 bits ; par conséquent vous devez faire attention aux reports. Par exemple, le bloc 130 est adressé par IND1=0001h, IND0=0400h qui correspond à la valeur décimale 66560. Il est important, en utilisant une communication décimale, que vous divisiez les deux champs pour faire en sorte que le PIC puisse travailler correctement. Par conséquent vous devez utiliser 1 pour IND1 et 1024 pour IND0 ; vous ne pouvez pas envoyer directement la valeur 66560 car vous finiriez par mettre en "overflow" (dépassement de capacité) IND0, ce qui ferait échouer l'opération d'écriture. Nous vous présentons les deux points de vue car si, d'un côté, l'utilisation des codes hexadécimaux est plus proche de la logique du PIC, de l'autre l'utilisation des valeurs décimales est bien plus commode. Le mode de communication que nous entendons utiliser ayant été vu, nous allons analyser maintenant le code d'initialisation (voir "listing" 2)

Comme vous voyez, nous paramétrons toutes les broches du PORTA comme sorties et, pour le PORTC, nous maintenons comme broche d'entrée (côté PIC) la RC4 qui est justement celle reliée à la sortie de la carte. Le PORTA est défini avec des broches toutes numériques de manière à pouvoir l'utiliser pour la signalisation de dispositif occupé. Passons au programme principal en sautant les étiquettes des sous-programmes que nous verrons plus loin. Le programme principal commence avec une procédure essentielle qui consiste en l'initialisation de la carte, laquelle peut être comprise selon trois phases fondamentales :

1) RESET CARD (CMD0)
2) INITIALISATION (CMD1)
3) PRECISION LIMITES 512 octets (CMD16).

Voici comment se présente le code de la phase 1 (voir "listing" 1). Nous mettons la valeur logique haute sur la ligne correspondante afin de signaler que le dispositif se trouve en état "occupé" et que donc il ne peut recevoir aucune commande. Le "reset", on le voit, est précédé d'une séquence de cycles d'horloge à vide ("Dummy Clock"), comme les spécifications de la SD-Card l'impliquent. En particulier le CMD0 est envoyé en maintenant la ligne "Chip Select" à 0 et cette procédure permet d'entrer en mode SPI.
Le "reset" étant effectué, un message avec indication de la réponse reçue par la carte est envoyé. Dans le logiciel que nous fournissons, il est possible de surveiller ces messages directement à l'écran. Nous prévoyons une sorte de "time-out" (délai) pour faire en sorte que la procédure soit bloquée si la

Figure 5: Douze "listings" du programme résident.

"Listing" 1

```
OCC  var PORTA.2  'LIGNE OCCUPEE/LIBRE
ERR  var PORTA.1  'LIGNE ERREUR

SS   var PORTC.2  'SÉLECTION-CARD PIN1
SCK  var PORTC.3  'CLOCK-CARD PIN5
SDO  var PORTC.4  'DONNÉES-SORTIE-CARD PIN7
SDI  var PORTC.5  'DONNÉES-ENTRÉE-CARD PIN2

SCL  var PORTC.0  'CLOCK FRAM
SDA  var PORTC.1  'DONNÉES  FRAM

TX   var PORTC.6  'LIGNE SORTIE VERS PC
RX   var PORTC.7  'LIGNE ENTRÉE DU PC
BPS  con 32       'VITESSE DE COMMUNICATION 19 200 bits/s

RISP1 var byte    'RÉPONSE TYPE R1 DE CARD (8 bits)
RISP2 var word    'RÉPONSE TYPE R2 DE CARD (16 bits)

IND1 var word     'ADRESSE SDCARD WORD HAUTE bit 16-31
IND0 var word     'ADRESSE SDCARD WORD BASSE bit 0-15

INDEEP var word   'ADRESSE CELLULE FRAM
INIEEP var word   'ADRESSE INITIALE FRAM
CTL    var byte   'OCTET DE CONTROLE FRAM

CMD    var byte   'CODE COMMANDE A EXECUTER
LOC512 var word   'POINTEUR ADRESSE FRAM A METTRE A JOUR
DATO   var byte   'VALEUR A INSERER EN MEMOIRE

CAR    var byte   'CARACTERES A ECRIRE
CONTA1 var byte   'COMPTEUR
CONTA2 var byte   'COMPTEUR
QX     var byte   'ANALYSES 4 bits Réponse Données
```

Lignes de signalisation vers dispositif externe

Définitions lignes de connexion avec la SD-Card

Définitions Bus I2C pour la FRAM

Définitions lignes port série et paramètres de connexion

Définitions formées de réponse comme établi dans les spécifications pour le mode SP

Adressage blocs de 512 octets de la SD-Card

Adressage de la FRAM FM24C64 qui arrive directement à travers une valeur mot

Séquence de champs utilisée pour l'envoi de commandes à travers le port série

Ensemble de variables d'utilisation générique (compteurs pour time-out, extraction bit, etc.)

"Listing" 2

```
ADCON1=6
TRISA=%00000000
TRISC=%00010000
PORTA=%00000000
PORTC=%00000000
SEROUT2 TX,BPS,[«++++++++++++++++++++»]
PAUSE 500

GOTO INIZIO       'Saute au programme principal
```

Ce registre Analog/Digital Conversion Register permet de définir toutes les broches du PORT comme numériques pour pouvoir commander les lignes de signalisation ERR et OCC

Tableau 1

Dénomination	Longueur	Description
CMD	1 octet	Contient le code d'identification de la commande. Reflète le code utilisé par les commandes SPI des spécifications Secure Digital. On utilise par conséquent le code 24 pour l'écriture et le code 17 pour la lecture.
IND1	2 octets	C'est le mot haut de l'adresse à 32 bits utilisé pour identifier chaque bloc de 512 octets de l'espace de mémorisation de la carte. Le champ est prévu dans les spécifications SD.
IND0	2 octets	C'est le mot bas de l'adresse à 32 bits utilisé pour identifier chaque bloc de 512 octets de l'espace de mémorisation de la carte. Le champ est prévu dans les spécifications SD.
LOC512	2 octets	C'est le pointeur référé à l'élément du bloc de 512 octets que nous voulons mettre à jour. Le champ n'est pas prévu par les spécifications mais nous sert à rendre possible l'écriture d'un octet unique pour chaque transaction d'écriture. Il est clair qu'il doit avoir une valeur comprise entre 0 et 511.
DATO	1 octet	C'est la valeur que nous voulons écrire dans la carte.

“Listing” 3

```
INIZIO:
        OCC = 1  'Je mets en occupé durant initialisation
        ERR = 0
        SS=1
        FOR CONTA1 = 1 TO 10
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]  'J'envoie cycles de clock à vide
        NEXT CONTA1
        SS=0
        PAUSE 50

'*******************************************
'* CMD0 en maintenant SS a 0
'*******************************************

        SHIFTOUT SDI, SCK, MSBFIRST, [$40,$00,$00,$00,$00,$95]  'J'envoie CMD0
        SHIFTIN SDO, SCK, MSBPRE, [RISP1]  'Je lis réponse R1 sur Card
        CONTA1 = 0
        WHILE RISP1 <> 1
        SHIFTIN SDO, SCK, MSBPRE, [RISP1]  'Je lis réponse R1 sur Card
        CONTA1 = CONTA1 + 1
        IF CONTA1 >= 255 THEN  'Time-Out écoulé je sors
        GOTO ERRORE
        ENDIF
        WEND
        SS=1
        PAUSE 50
        SS=0
        SEROUT2 TX,BPS,[«RESET RISP=»,IBIN8 RISP1,10,13]
```

80 cycles de Dummy Clock comme précisé dans les spécifications SD-Card

Après la commande de reset on attend que la carte rentre en idle-state et c'est pourquoi on trouve dans la réponse le bit le moins significatif situé en 1

“Listing” 4

```
'*************************************************
'* Envoi à répétition CMD1 jusqu'à la réponse = 0
'*************************************************
        CONTA1 = 0
        RISP1 = 1
        WHILE RISP1 <> 0
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SS=0
        PAUSE 50
        SHIFTOUT SDI,SCK,MSBFIRST,[$41,$00,$00,$00,$00,$FF,$FF] 'Envoi CMD1
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = CONTA1 + 1
        IF CONTA1 >= 255 THEN  'Time-Out écoulé je sors
        GOTO ERRORE
        ENDIF
        WEND
        SEROUT2 TX,BPS,[«CARD IN MODO SPI PRONTA!»,10,13]
```

A partir du moment où la réponse est à zéro la Card sort de l'état de busy en ayant terminé la phase d'initialisation.

Au moins 8 clock à vide entre une transaction et l'autre comme conseillé dans les spécifications SPI SD-Card.

“Listing” 5

```
'*********************************************
'* Envoi CMD16
'*********************************************
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SS=0
        SHIFTOUT SDI,SCK,MSBFIRST,[$50,$00,$00,$02,$00,$FF,$FF] 'Envoi CMD16
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = 0
        WHILE RISP1<> 0
```

Dans le CMD16 nous transférons la grandeur du bloc que nous voulons lire ou écrire 0200h = 512

```
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = CONTA1 + 1
        IF CONTA1 >= 255 THEN 'Time-Out écoulé je sors
        GOTO ERRORE
        ENDIF
        WEND
        SEROUT2 TX,BPS,[«BLOCCO DATI 512 SETTATO»,10,13]
```

"Listing" 6

```
CANC:
        CTL = %10100000  'Contrôle
        FOR INDEEP = $0000 to $01FF
        I2CWRITE SDA,SCL,CTL,INDEEP,[$00]
        NEXT INDEEP
        RETURN
```

"Listing" 7

```
RICEVI:
        OCC = 0   'Je mets en libre prêt à recevoir commandes
        SERIN2 RX,BPS,[HEX2 CMD,HEX4 IND1,HEX4 IND0,HEX4 LOC512,HEX2 DATO]
        IF CMD = 24 THEN
        OCC = 1   'Je mets en occupé
        ERR = 0
        GOTO SCRIVI
        ENDIF
        IF CMD = 17 THEN
        OCC = 1
        ERR = 0
        GOTO LEGGI
        ENDIF
        GOSUB CANC 'Efface FRAM
        GOTO RICEVI
```

Séquence CODE COMMANDE + ADRESSE BLOC + POINTEUR FRAM + VALEUR DONNÉE transférée à travers port sériel.

Si le CODE COMMANDE est égal à 24, écris et s'il est égal à 17, lis.

Tableau 2

Masque	Longueur	Description
BINAIRE	de 1 à 16	Reçoit la représentation ASCII de la valeur binaire
DEC	de 1 à 5	Reçoit la représentation ASCII de la valeur décimale
HEX	de 1 à 4	Reçoit la représentation ASCII de la valeur hexadécimale

"Listing" 8

```
SCRIVI:
        INIEEP = $0000
        GOSUB LEGGIDAT
        I2CWRITE SDA,SCL,CTL,LOC512,[DATO]
        INIEEP = $0000
        GOSUB SCRIVIDAT
        GOTO RICEVI  'Je reviens à la réception des commandes
```

Mise à jour de la valeur pointée à partir de l'adresse insérée en LOC512.

"Listing" 9

```
LEGGIDAT:
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SS=0
        SHIFTOUT SDI,SCK,MSBFIRST,[$51,IND1.BYTE1,IND1.BYTE0,IND0.BYTE1,IND0.BYTE0,$FF] 'J'envoie  CMD17
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = 0
        WHILE RISP1<> 0
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = CONTA1 + 1
```

Adresse à 32 bits définissant le bloc à lire. Les paramètres sont initialisés directement dans la SERIN2.

```
        IF CONTA1 >= 255 THEN 'Time-Out écoulé je sors
        GOTO ERRORE
        ENDIF
        WEND
'Réception START BLOCK
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = 0
        WHILE RISP1<> $FE
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA1 = CONTA1 + 1
        IF CONTA1 >= 255 THEN 'Time-Out écoulé je sors
        GOTO ERRORE
        ENDIF
        WEND
'Réception BLOCCO DATI

        CTL = %10100000  'Contrôle
        FOR INDEEP = INIEEP to (INIEEP+511)
        SHIFTIN SDO,SCK,MSBPRE,[CAR]
        I2CWRITE SDA,SCL,CTL,INDEEP,[CAR]
        NEXT INDEEP
'Réception CRC
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]

'Dès que la Card a fini de lire je vérifie l'état de la Card
'J'envoie le CMD13
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SS=0
        SHIFTOUT SDI,SCK,MSBFIRST,[$4D,$00,$00,$00,$00,$FF]
'Je reçois le status à 16 bits réponse format 2
        SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
        CONTA2=0
        WHILE RISP2.BYTE0 <> 0
        SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
        CONTA2 = CONTA2 + 1
        IF CONTA2 >= 255 THEN
        GOTO ERRORE2
        ENDIF
        WEND
        SEROUT2 TX,BPS,[«Lecture Bloc Adresse= «, IHEX IND1, IHEX IND0,10,13]
        RETURN
```

11111110b correspond au Start-Block envoyé par le contrôleur pour signaler qu'une séquence de 512 octets correspondant aux données requises suivra.

Suite à la demande de status register on vérifie une réponse de type 2 pour établir si l'opération est terminée avec succès ou non.

"Listing" 10

```
SCRIVIDAT:
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
'J'envoie CMD24 écriture bloc sur SDCard
        SS=0
        SHIFTOUT SDI,SCK,MSBFIRST,[$58,IND1.BYTE1,IND1.BYTE0,IND0.BYTE1,IND0.BYTE0,$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA2 = 0
'Réponse pour voir si la Card est prête à recevoir des données
        WHILE RISP1 <> 0
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA2 = CONTA2 + 1
        IF CONTA2 >= 255 THEN
        GOTO ERRORE
        ENDIF
        WEND
'J'envoie Start Block %11111110=$FE
        SHIFTOUT SDI,SCK,MSBFIRST,[$FE]
```

Adresse à 32 bits établissant le bloc à écrire.

```
'Je commence Déchargement FRAM et Ecriture sur SDCARD

        CTL = %10100000  'Controllo
        FOR INDEEP = INIEEP to (INIEEP+511)
        I2CREAD SDA,SCL,CTL,INDEEP,[CAR]
        SHIFTOUT SDI,SCK,MSBFIRST,[CAR]
        NEXT INDEEP
'J'envoie CRC
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF,$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]

        QX = %00000000
        QX = RISP1 & $0F
        IF QX <> %00000101 THEN
        GOTO ERRORE
        ENDIF

'Je reçois le bit busy de la Card alors qu'elle écrit
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA2=0
        WHILE RISP1 = 0
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        CONTA2 = CONTA2 + 1
        IF CONTA2 >= 255 THEN
        GOTO ERRORE
        ENDIF
        WEND

'Dès que la Card a fini d'écrire je vérifie l'état de la Card
'J'envoie le CMD13
        SS=1
        SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
        SHIFTIN SDO,SCK,MSBPRE,[RISP1]
        SS=0
        SHIFTOUT SDI,SCK,MSBFIRST,[$4D,$00,$00,$00,$00,$FF]
'Je reçois le status à 16 bits réponse format 2
        SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
        CONTA2=0
        WHILE RISP2.BYTE0 <> 0
        SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
        CONTA2 = CONTA2 + 1
        IF CONTA2 >= 255 THEN
        GOTO ERRORE2
        ENDIF
        WEND
        SEROUT2 TX,BPS,[«Ecriture Bloc Adresse= «, IHEX IND1, IHEX IND0,10,13]
        RETURN
```

J'extrais les 4 bits les moins significatifs de la réponse

0101 Données acceptées
1011 Données refusées pour erreur CRC
1101 Données refusées pour erreur écriture

Suite à la demande de status register on vérifie une réponse de type 2 pour établir si l'opération est terminée avec succès ou non.

"Listing" 11

```
ERRORE:
        SEROUT2 TX,BPS,[«PROCESSO INTERROTTO R1= «,IBIN8 RISP1,10,13]
        ERR=1
        GOTO RICEVI
ERRORE2:
        SEROUT2 TX,BPS,[«PROCESSO INTERROTTO R2= «,IBIN8 RISP2.BYTE1, IBIN8 RISP2.BYTE0, 10,13]
        ERR=1
        GOTO RICEVI
```

"Listing" 12

```
LEGGI:
        INIEEP = $0000
        GOSUB LEGGIDAT    'Je lis le secteur adressé par IND0 et IND1 et je le mets en FRAM
        CONTA1 = 0
        FOR INDEEP = $0000 TO $01FF   'Décharge FRAM à travers port sériel
```

```
I2CREAD SDA,SCL,CTL,INDEEP,[CAR]
SEROUT2 TX,BPS,[DEC3 CAR,32]
CONTA1 = CONTA1 + 1
IF CONTA1 = 16 THEN
SEROUT2 TX,BPS,[10,13]
CONTA1 = 0
NEXT INDEEP
GOTO RICEVI  'Je retourne à la réception des commandes
```

On transmet par le port série la représentation ASCII de la valeur décimale de l'octet sur trois chiffres suivie d'un espace pour rendre plus facile la visualisation sur console.

Tous les 16 éléments on envoie un line feed- carriage return pour diviser les 512 octets en 32 lignes.

carte refuse de se réinitialiser à cause d'un dysfonctionnement. Ensuite, on lance la procédure d'initialisation du dispositif et on attend qu'elle se termine, en vérifiant la réponse envoyée (voir "listing" 4). Rappelons qu'en mode SPI, la Card est en mesure d'envoyer exclusivement deux types de réponse : une à 8 bits (RISP1) et une à 16 bits (RISP2) ayant chacune une structure particulière de signalisation des erreurs.

À travers un compteur (CONTA1) on établit l'atteinte de la limite de répétition de CMD1 au-delà duquel on suppose que la carte refuse de répondre ou que le processus d'initialisation n'a pas abouti. Quand l'initialisation s'est terminée avec succès, un message nous informant que la carte est entrée en mode SPI et qu'elle est prête à recevoir des commandes. Nous utilisons donc le CMD16 pour communiquer à la carte le nombre d'octets constituant le bloc de lecture et d'écriture ("listing" 5). En effet, la longueur par défaut est de 512 octets, donc cette commande est plus une précaution formelle, vu que plus tard on pourra trouver dans le commerce des cartes avec des blocs de différentes dimensions. Dans les paramètres de cette commande on insère la valeur 0200h (en décimal 512).

A la fin de cette opération aussi on envoie le message de signalisation prévu. Maintenant que la carte est prête, nous devons préparer la FRAM et nous le faisons en appelant une routine nommée CANC pour effacer toutes les valeurs contenues dans les 512 premiers octets représentant notre aire d'échange.

```
GOSUB CANC 'Efface FRAM
```

Cette commande appelle la séquence du "listing" 6. Nous ne faisons d'ailleurs rien d'autre qu'écrire la valeur 0 dans les 512 premières adresses de la FRAM et nous revenons au programme principal. La définition de subroutine est essentielle chaque fois que nous devons réexécuter plusieurs fois la même séquence de code. A la fin nous aurons une compilation de dimensions inférieures et le programme sera encore plus lisible pour des insertions ultérieures. Il ne nous reste alors qu'à mettre le PIC en attente de commandes de la part du port sériel. Avant tout, mettons au niveau logique bas la ligne "occupée", puis exécutons une paire de IF qui passeront le contrôle à deux étiquettes différentes selon le code de commande reçu. Voyons-les concrètement dans le "listing" 7.

L'instruction clé de cette séquence est :

```
SERIN2          RX,BITS/S,[HEX2
CMD,HEX4
IND1,HEX4               IND0,HEX4
LOC512,HEX2
DATO]
```

C'est bien de la comprendre complètement, car elle permet d'adapter l'interface à ce que nous voulons transférer de l'extérieur. Ici nous analysons le cas hexadécimal. Rien n'empêche de tout transformer en décimal en faisant bien attention à l'adressage des blocs avec IND0 et IND1. On considère en outre qu'à travers la traduction en décimale la commande devient nettement plus longue. SERIN2 dans sa forme la plus simple a la syntaxe suivante :

```
SERIN2 DataPin, Mode, [Item…]
```

Le "datapin" se rapporte à la broche du PIC utilisée comme entrée (dans ce cas RX). "Mode", en revanche, établit la vitesse de communication, parité, etc. Ces paramètres sont définis en "modedefs.inc" que nous incluons au début du "pgm". Notre BPS = 32 correspond à une vitesse de 19 200 bits/s (pas de parité, 1 bit de stop). Il faut savoir que pour chaque champ que nous voulons recevoir, nous pouvons indiquer un format d'entrée différent. Donc si nous envoyons une séquence "11010100" et que nous utilisons le masque "BIN8", la variable sera initialisée avec la valeur décimale 212. Pour avoir le même résultat avec "DEC3" nous devons envoyer "212" alors qu'avec "HEX2" nous devons envoyer "D4". La séquence pour un octet a une longueur de 8 pour "BIN", 3 pour "DEC" et 2 pour "HEX". Eh bien, nous pouvons maintenant traduire tranquillement l'instruction précédente en décimal :

```
SERIN2 RX,BPS,[DEC2 CMD,DEC5
IND1,DEC5               IND0,DEC5
LOC512,DEC3
DATO]
```

Il faut faire attention au fait que les codes ASCII de chaque chiffre sont transférés et par conséquent, même si formellement nous avons parlé d'une commande à 8 octets, en réalité on en a transféré beaucoup plus à cause de cette représentation. De plus, en précisant la longueur du masque d'entrée, pensez que seul le nombre de chiffres établi sera reçu, pas un de plus ni de moins ! Tous les caractères ne correspondant pas à des chiffres (espaces, etc.) sont écartés. Après avoir reçu la séquence, on vérifie le code commande que nous transférons dans la variable "CMD". S'il est égal à 24, le contrôle passe à l'étiquette "SCRIVI", s'il est égal à 17, à l'étiquette "LEGGI". S'il n'est ni l'un ni l'autre, le système revient en attente de nouvelles instructions. Dans les deux cas, la ligne "occupée" est mise au niveau logique haut . Attention : au cas où une commande de lecture est transmise, les champs "LOC512" et "DATO" sont inutiles et donc ils sont complètement ignorés. Voyons ce qui arrive avec l'étiquette "SCRIVI" (voir "listing" 8).

Pour chaque opération de lecture nous avons une subroutine nommée "LEGGIDAT" qui exige en entrée trois paramètres : "IND1", "IND0", "INIEEP". Les deux premiers constituent l'adresse à 32 bits du bloc à 512 octets à lire et le troisième représente l'adresse initiale de la FRAM que nous utiliserons pour transférer les données. Pour nous il est égal à 0 car nous utiliserons toujours le premier secteur comme aire d'échange. Vous pensez peut-être que ce dernier paramètre pourrait être une constante : en fait le développement de cette routine a été conçu en prévision de futures évolutions de concept : en effet, quand nous devrons écrire sur

une carte formatée en FAT16, nous devrons utiliser diverses aires de la FRAM qui auront diverses fonctions et donc nous devrons préciser quel secteur lire. Voyons les instructions nécessaires pour l'opération de lecture (voir "listing" 9).

Le "CMD17" est envoyé à la carte : comme paramètres sont transférés les 4 octets représentant les 32 bits nécessaires pour adresser le bloc de la carte à lire. Naturellement, les variables concernées par cette commande ont été déjà valorisées à travers la "SERIN2". Après que la commande ait été acceptée, la carte envoie une séquence de bits particulière nommée "Start-Block" et la transmission en séquence de 512 octets commence. A chaque octet correspond immédiatement une écriture en FRAM, le pointeur étant repositionné au fur et à mesure sur la cellule de mémoire à écrire. A la fin est lu le registre de "Status" de la carte à travers le "CMD13" pour vérifier si l'opération s'est bien passée ou non. Si l'issue est positive, un message de confirmation avec indication de l'adresse du bloc lu en format hexadécimal est envoyé par le port sériel. Si nous revenons à la séquence initiale, concernant l'étiquette "SCRIVI", nous voyons qu'après avoir lu le bloc et l'avoir transféré dans le premier secteur de la FRAM, une instruction "I2CWRITE" est exécutée, ce qui nous permet d'écrire la valeur "DATO" dans l'adresse de FRAM pointée à travers "LOC512". Ainsi, nous ne faisons rien d'autre que mettre à jour le secteur qui nous intéresse avec la nouvelle donnée que nous voulons mémoriser.

A ce moment sur la FRAM nous avons la "photo" de la séquence de 512 octets que nous voulons reproduire à l'intérieur de la carte. Il ne reste qu'à transférer ce secteur dans le bloc adressé par "IND1" et "IND0". Nous le faisons en appelant une autre routine qui utilise les mêmes paramètres que celle utilisée en lecture mais avec une signification complémentaire. "IND0" et "IND1" composent l'adresse du bloc à écrire et "INIEEP" est l'adresse initiale de la FRAM dont on veut lire les données pour les transférer. Voyons en détail cette sous procédure nommée "SCRIVIDAT" ("listing" 10).

Le "CMD24" avec ses 4 octets correspondant à l'adresse du bloc à écrire est envoyé puis, comme pour la commande de lecture, il faut transmettre le "Start-Block" afin d'informer la carte que nous allons lui envoyer la séquence de 512 octets que nous

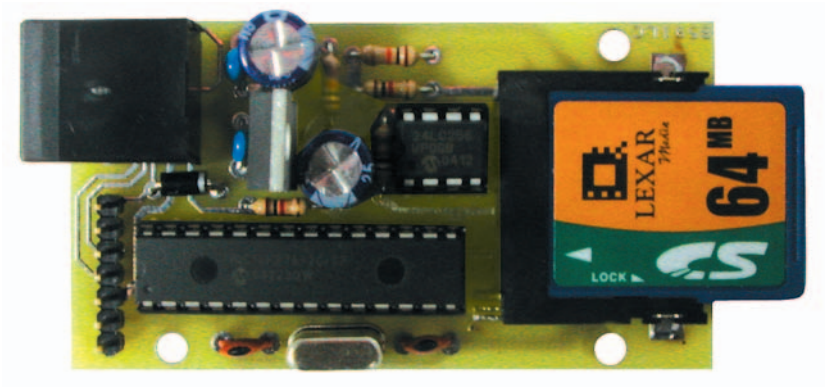


Figure 3 : Photos d'un des prototypes de l'unité de mémoire.

voulons mémoriser en elle. Là encore nous avons inséré un contrôle de "time-out" (délai) avec le même compteur, ce qui garantit une bonne fiabilité de ces opérations. De même que pour le "CMD17", nous vérifions l'état de la carte à la fin de l'opération et nous envoyons un message signalant l'imminence de l'écriture avec indication de "IND1" et "IND0" au format hexadécimal.

Dans le cycle de déchargement de la FRAM il faut noter l'opération de "AND" logique pour l'extraction du "Data Response Token" envoyé pour chaque bloc de données écrit. En cas d'erreur, c'est le rôle du PIC de bloquer la transaction à travers "CMD12" (nous la signalons en sautant à l'étiquette correspondante) : en cas d'écriture monobloc, en effet, aucune instruction de bloc de la séquence d'envoi n'est requise.

Reprenons la séquence de l'étiquette "LEGGI". Vous le voyez, à la fin de l'opération de mise à jour, un "GOTO" est fait à l'étiquette "RICEVI" ; ainsi nous remettons la ligne "occupée" au niveau logique bas et prédisposons le système pour la réception d'une nouvelle commande. En ce qui concerne la signalisation des erreurs, nous avons préparé une ligne fonctionnant comme la RC2 : un état logique haut signifie que l'opération ne s'est pas bien terminée et un état logique bas qu'il y a une erreur. Nous avons fait un compromis : après avoir signalé l'erreur, le système se met à nouveau en réception pour que le dispositif externe (celui qui commande) puisse répéter l'opération. Naturellement, on peut prévoir que, si une opération échoue plusieurs fois, un "reset" physique sera utilisé (en mettant la broche MCLR au niveau logique bas). Les routines gérant les erreurs sont au nombre de deux : en effet, en mode SPI deux types de signalisation sont prévues, type 1 long de 8 bits et type 2 long de 16 bits (voir "listing" 11).

Il ne reste qu'à analyser la partie relative au code commande 17, c'est-à-dire à la lecture d'un bloc ("listing" 12). Appelons la subroutine vue précédemment pour lire le bloc adressé par "IND1" et "IND0" et transférons les 512 octets sur le premier secteur de la FRAM. Ensuite déchargeons le mémoire temporaire en transmettant chaque octet à travers la RS232. Dans ce cas, attention : nous envoyons la représentation ASCII de la valeur décimale contenue dans la carte. Naturellement, rien n'interdit d'utiliser une représentation différente, au cas où ce serait nécessaire pour communiquer correctement avec le dispositif externe. Si, par exemple, une séquence hexadécimale (au lieu de décimale) était demandée, l'instruction deviendrait :

```
SEROUT2 TX,BPS,[HEX2 CAR]
```

Voyons maintenant le développement côté PC : il nous permettra de tester facilement notre interface.

Le logiciel

Nous avons développé un petit programme qui envoie à une interface, par le port série d'un PC, les commandes de lecture et d'écriture selon la syntaxe que nous venons d'expliquer (COMMANDE, ADRESSE 32 bits, POINTEUR, DONNÉE). Ainsi, vous pouvez expérimenter directement les fonctions que nous avons évoquées ci-dessus. Nous croyons rendre ainsi plus simple le processus de développement du programme résident, nécessaire pour commander notre interface. Nous laissons, naturellement, à l'expérimentateur avisé toute latitude pour décider du domaine d'utilisation. Le

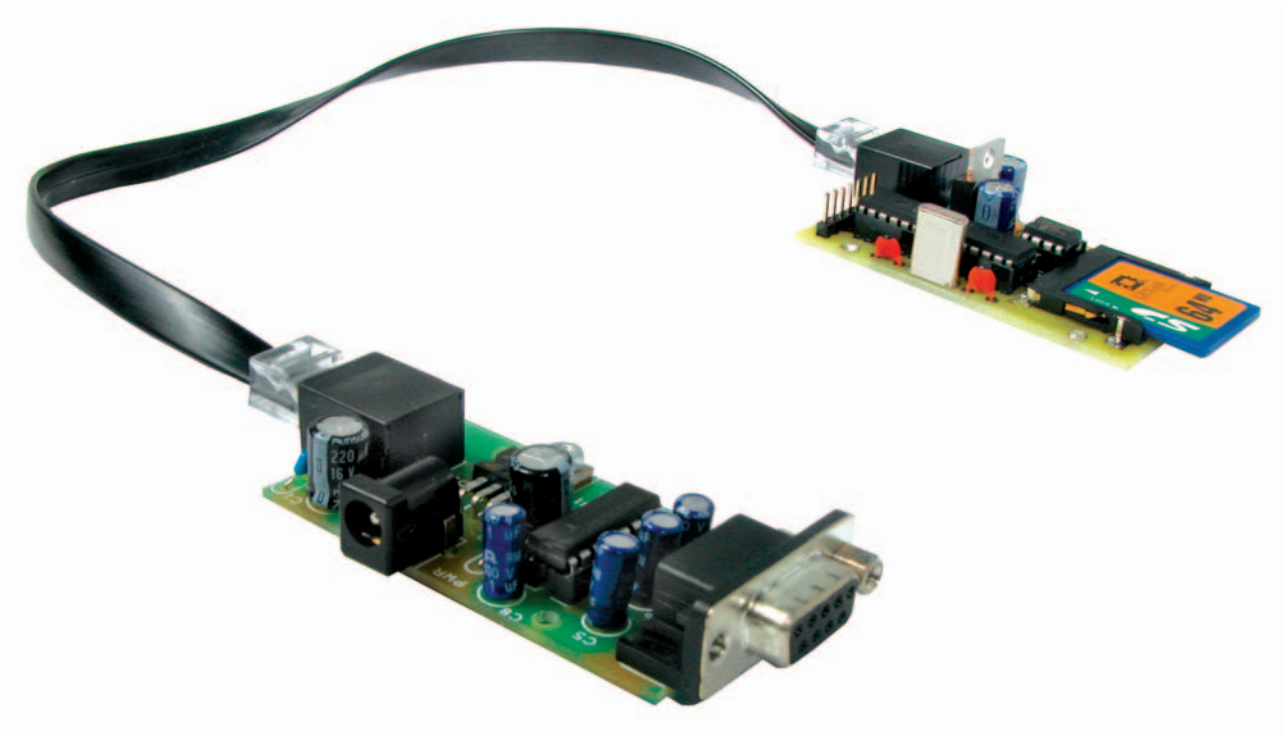


Figure 4 : Photo d'un des prototypes de l'unité de mémoire utilisant, pour son couplage à l'ordinateur, l'interface RS232 ET475.

panneau de contrôle est très intuitif. Il se compose d'un écran unique regroupant dans sa partie haute les fenêtres de configuration du port sériel et celles permettant à l'usager de valoriser les divers champs dont se compose chaque commande à envoyer à l'interface. A droite, nous trouvons trois poussoirs permettant de se connecter au port (Ouvre Port), d'envoyer la commande que l'on vient de composer (Envoi) et d'effacer le contenu de la fenêtre de log (Effacer). Au lancement nous trouvons le poussoir "Envoi" déshabilité. Pour l'activer on doit d'abord préciser le port auquel nous relions l'interface et cliquer sur le poussoir "Ouvre Port". Il est alors possible de valoriser les divers champs constituant la commande.

L'opération de lecture et d'écriture est établie à travers le premier radiogroupe. Selon l'option choisie, le programme insère le mot de deux caractères 17 ou 24 de la séquence d'octets à envoyer. Naturellement il n'est pas possible de modifier le port de connexion sinon après avoir cliqué sur "Ferme Port". En ce qui concerne les champs "LOC512" on doit se cantonner dans une gamme allant de 0 à 511 ; "DATO" peut être valorisé seulement de 0 à 255. L'adresse du bloc correspond à la valeur à 32 bits utilisée pour les "CMD17" et "CMD24". Le logiciel s'occupe de valoriser correctement "IND0" et "IND1" à travers une division entière. A l'écran, il est possible de contrôler les divers processus exécutés. A chaque envoi, par exemple, est visualisée la séquence des caractères émis ; en outre, sur le même écran, sont visualisés les messages de service produits par l'interface. Par exemple, si nous lançons le programme, si nous ouvrons le port correspondant et si nous alimentons le circuit, nous voyons se dérouler l'indication des phases d'initialisation de la carte, du "reset" jusqu'à la détermination de la longueur du bloc à lire ou écrire. Ensuite, si nous envoyons une commande d'écriture, nous verrons la séquence d'envoi et les réponses de service de l'interface avec l'indication des adresses hexadécimales relatives aux champs "IND0" et "IND1". Durant une opération d'écriture, on verra se dérouler d'abord le message de lecture qui transfère les données de la carte à la FRAM, puis celle d'écriture de la FRAM à la SD-Card après mise à jour de l'octet indiquée à travers le pointeur "LOC512".

Conclusion

Ce petit montage, malgré sa simplicité, permet de rendre une carte SD pleinement compatible avec n'importe quel système doté d'une interface série. Il pourra être intégré dans d'autres appareils nécessitant une grande capacité de mémoire aisément gérable. Nous avons d'ailleurs l'intention de vous soumettre une prochaine fois un point de vue différent : au lieu de sauvegarder nos données dans des adresses de mémoire, nous pensons utiliser une carte SD comme un véritable disque dur produisant un fichier directement lisible sur notre ordinateur. Bien sûr, le problème n'a pas de solution banale, car il s'agit de bien connaître les structures que créent les SE Microsoft sur chaque appareil formaté. Certains d'avoir excité votre imagination, nous vous donnons rendez-vous dans de prochains articles d'approfondissement où nous prendrons ce taureau par les cornes ! En effet, nous effectuerons une sorte de concept à rebours de cette interface afin de la rendre capable de produire des fichiers visualisés sur PC comme s'ils étaient de vulgaires fichiers de texte.

Comment construire ce montage ?

Tout le matériel nécessaire pour construire cette unité de mémoire ET581, ainsi que l'interface RS232 ET475, est disponible chez certains de nos annonceurs. Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp.

Les composants programmés sont disponibles sur www.electronique-magazine.com/mc.asp. ◆

ELECTRONIQUE ET LOISIRS magazine

Tous les sommaires des anciens numéros disponibles sur internet : http://www.electronique-magazine.com



Au sommaire : Un VFO VHF programmable de 50 à 180 MHz avec microcontrôleur ST7, seconde partie : le logiciel - Une alarme téléphonique GSM à deux entrées pour voiture ou maison L'heure, la date et la température sur un afficheur géant Un inductancemètre numérique de 0,1 µH à 300 mH - Un antivol pour moto et scooter - Une régie lumières commandée par PC, première partie : le matériel - Un compte-tours numérique - Leçon 6 suite et fin : Comment utiliser le programme inDART-ST7 - Un appel téléphonique GSM à synthèse vocale pour alarme Apprendre l'électronique en partant de zéro : Les amplificateurs en classe A, B ou C (seconde partie et fin).

5,50 € port inclus



Au sommaire : Un contrôleur «pluie et vent» à affichage numérique (Idéal pour stores, parasols et autres Velux motorisés) - Un programmateur de PIC universel - Un fréquencemètre numérique à dix chiffres 2,2 GHz, première partie : la théorie - Un clavier à écran tactile personnalisable - Un terminal RS485 avec afficheur LCD et clavier - Un amplificateur MOSFET mono ou stéréo de 600 W RMS (2 x 300 W RMS) - Un contrôleur LAN / Internet à 16 entrées et 16 sorties numériques - Une régie lumières commandée par PC - seconde partie et fin : le logiciel et la liaison radio TX/RX Apprendre l'électronique en partant de zéro : les FLIP-FLOP.

5,50 € port inclus



Au sommaire : Passez des appels GSM avec votre téléphone fixe - Un pont réflectométrique pour analyseur de spectre - Un impédancemètre d'antenne Un ROSmètre VHF/UHF simple à lignes imprimées - Un ROSmètre à tores de ferrite de 1 à 170 MHz - Un fréquencemètre numérique à dix chiffres 2,2 GHz, seconde et dernière partie : la réalisation pratique Un testeur de bobinages - Un détecteur de fils secteur - Un générateur sinusoïdal 1 kHz - Une station météo directement sur Internet Un détecteur de vibrations Un détecteur de fuites pour micro-ondes Un thermostat numérique LCD - Un générateur BF-VHF piloté par ordinateur -Un thermomètre –50 à +150 °C à pont de Wheatstone - Etc.

5,50 € port inclus



Au sommaire : Un amplificateur 4 x 55 W pour voiture (Fonction "standby" et "mute") - Un potentiomètre électronique monolithique - Deux clignotants basse tension - Un variateur de vitesse pour moteur à courant continu (Technologie MOSFET) Un enregistreur/reproducteur de huit minutes avec les fonctions REC, PLAY et STOP - Un amplificateur à lampes de 60 W RMS en classe A - Un micro-espion GSM (à module GSM GR47) - Un clavier de six touches à effleurement - Un contrôleur pour moteurs pas à pas - Sur l'Internet - Apprendre l'électronique en partant de zéro : Un fréquencemètre numérique à 5 chiffres 10 MHz (première partie).

5,50 € port inclus



Au sommaire : Un nouveau programmateur / duplicateur d'EPROM pour port parallèle, première partie : le matériel - Un programmateur de PIC première partie : le matériel - Un système émetteur et son récepteur infrarouge à deux canaux (Portée de 15 mètres environ) - Une minuterie multiple à ST7 - Un panneau lumineux multifonction : heure/date/ température avec six chiffres de sept segments à led - Une interface USB pour PC (avec son logiciel) première partie : le matériel - Sur l'Internet - Apprendre l'électronique en partant de zéro : Un fréquencemètre numérique à 5 chiffres 10 MHz (seconde partie et fin).

5,50 € port inclus



Au sommaire : Visualiser les SMS reçus sur PC via le port série - Un radar de recul à ultrasons de 0 jusqu'à 1,5 m Un amplificateur stéréo 2 x 30 W. Un programmateur d'EPROM pour port parallèle seconde partie et fin: le logiciel Un programmateur de PIC seconde partie et fin - Une interface USB pour PC seconde partie et fin : le logiciel - Un fréquencemètre à neuf chiffres LCD 550 MHz avec la possibilité de soustraire ou d'ajouter la valeur de la MF d'un récepteur - Un détecteur pendulaire pour sismographe permettant via une interface de visualiser sur un PC tout tremblement de terre - Apprendre l'électronique en partant de zéro Le compteur CD40103 à 8 bits

5,50 € port inclus



Au sommaire : Un variateur de puissance au standard DMX512 - Un appareil de lecture et d'analyse de cartes magnétiques avec acquisition des données par le port série : Première partie : Le logiciel et l'interface de contrôle - Dix schémas simples de préamplificateurs BF à transistors - Un gestionnaire de sonneries mélodiques de GSM - Un contrôle à distance à 10 canaux par deux fils - Un moteur à courant continu piloté par ordinateur - Un variateur à effleurement pour ampoule - Un mélangeur DMX 8 canaux pour régie de lumière - Sur l'Internet - Apprendre l'électronique en partant de zéro : Les nombres Binaires er Hexadécimaux

6,50 € port inclus



sommaire : DMX512, protocoles et applications - Un variateur DMX à huit canaux pour régie lumière Première partie : l'unité de contrôle et les unités d'extension - Un appareil de lecture et d'analyse de cartes magnétiques : Seconde partie et fin: le programme de l'interface et la liaison GSM - Deux émetteurs infrarouges à 15 canaux Un récepteur infrarouge à 15 canaux Un contrôle à distance DTMF GSM. Un moteur à courant continu piloté par ordinateur Seconde partie et fin: le logiciel - Un anémomètre programmable simple - Cours sur le SitePlayer SP1 Apprendre l'électronique en partant de zéro : Le PUT ou Transistor Unijonction Programmable.

5,50 € port inclus



Au sommaire : Un mesureur de champ 433,92 MHz - Un variateur DMX à huit canaux pour régie lumière Seconde partie: l'unité de puissance et les nouveaux modules variateurs à microcontrôleur. Un préamplificateur Hi-Fi avec contrôle de tonalité - Une alarme vidéo à distance avec Siemens C65 - Comment programmer le module GPS Sony Ericsson GM47 Première partie : construction du programmateur Un testeur LPT pour port parallèle Un temporisateur électronique Cours sur le SitePlayer SP1: Deuxième partie : construction du programmateur Apprendre l'électronique en partant de zéro Comment utiliser l'oscilloscope Première partie: présentation de l'instrument (fonctions et commandes)

5,50 € port inclus



Au sommaire : Un émetteur FM - Un préamplificateur mono universel - Une alimentation 1 A - Un millivoltmètre numérique - Une sirène de police synthétisée - Un temporisateur avec commandes M/A - Un allumage électronique - Un avertisseur de risque de verglas - Un thermostat LCD - Un antivol auto - Une base de temps à quartz - Un amplificateur mono 7 W - Un amplificateur stéréo 2 x 30 W - Un clignotant stroboscopique à tube au xénon - Comment programmer le GPS Sony Ericsson GM47 (2éme partie) - Une platine de puissance à relais - Une platine de puissance à quatre triacs - Un compte-tours auto - Une protection pour haut-parleurs - etc.

6,50 € port inclus

Frais de port pour la CEE les DOM-TOM et l'étranger : Nous consulter.

Renseignements sur les disponibilités des revues depuis le numéro 1

Tél. : 0820 820 534 du lundi au vendredi de 9h à 12h

JMJ Editions B.P. 20025 - 13720 LA BOUILLADISSE

CD-ROM ENTIÈREMENT IMPRIMABLE

LISEZ ET IMPRIMEZ VOTRE REVUE SUR VOTRE ORDINATEUR PC OU MACINTOSH

NOUVEAU



SURVEILLANCE & SÉCURITÉ
5,50€
(Port inclus en France)

ABONNÉS 12 OU 24 NUMÉROS
- 50 %
SUR TOUS LES CD



LE CD 6 NUMÉROS
24€
(Port inclus en France)



LE CD 12 NUMÉROS
43€
(Port inclus en France)



SOMMAIRE INTERACTIF



ENTIÈREMENT IMPRIMABLE

34 € Les CD niveau 1 et 2
du Cours d'Électronique
en Partant de Zéro
(Port inclus en France)

SPÉCIAL 45 MONTAGES
5,50€
(Port inclus en France)



FRAIS DE PORT POUR LA CEE LES DOM-TOM ET AUTRES PAYS: NOUS CONSULTER.

adressez votre commande à :

JMJ/ELECTRONIQUE - B.P. 20025 - 13720 LA BOUILLADISSE *avec un règlement par Chèque à l'ordre de* ***JMJ***

Par téléphone : 0820 820 534 ou par fax : 0820 820 722 *avec un règlement par Carte Bancaire*

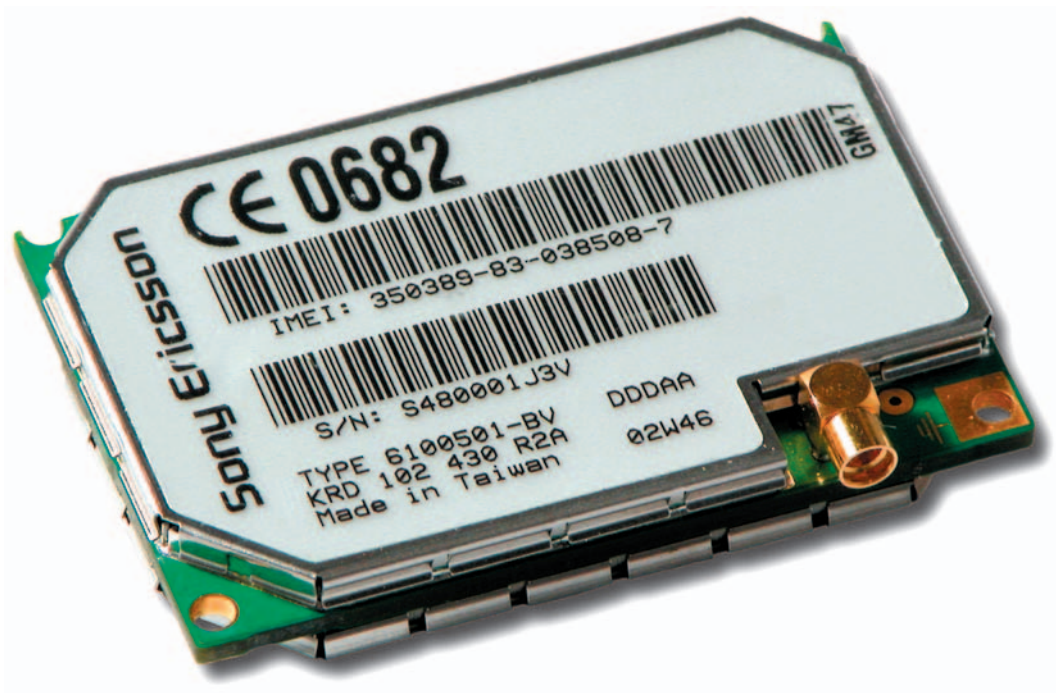
Vous pouvez également commander par l'Internet : www.electronique-magazine.com/anc_num.asp

Comment programmer

le module GPS Sony Ericsson GM47

Troisième partie

Dans cette série d'articles, nous vous apprenons à programmer et à utiliser le module GSM GM47 de Sony Ericsson. Nous approfondissons la connaissance du logiciel et du matériel de ce module afin de réaliser par la suite de nombreuses applications GSM. Une grande partie de cette série sera consacrée à la programmation du microcontrôleur présent à l'intérieur du module par des "scripts" utilisant un langage dérivé du C.



Après avoir consacré la première partie de cette série d'articles au module Sony Ericsson GM47 et à la platine d'expérimentation / programmateur à utiliser pour le développement et la programmation de ces modules, dans la deuxième nous avons commencé à apprendre (au moyen de "scripts" rédigés dans un langage dérivé du C) à programmer le microcontrôleur interne, auquel de nombreuses ressources correspondent. Rappelons à ce propos que Sony Ericsson fournit le pack M2mpower constituant un environnement de développement IDE complet : grâce à ce logiciel, il est possible d'écrire, tester et déboguer les "listings" avec un simple PC. Ensuite, à l'aide d'une connexion série entre le PC (utilisé pour développer le programme résident) et la platine d'expérimentation, on peut charger les "scripts" dans la mémoire du GM47. Une fois les "listings" insérés dans le module, ils sont lus ligne par ligne et exécutés en temps réel par le micro.

Les premiers exemples de "script" que nous avons analysés étaient destinés à la gestion du matériel (le module) : nous avons vu en effet comment le configurer, lire et écrire (à travers l'instruction io()) les quatre broches d'E / S numériques dont dispose le GM47. Puis nous avons analysé la gestion du port UART3 : nous avons vu l'instruction utc() qui permet de configurer (ouvrir et fermer) l'interface série ; l'instruction uts() qui transmet un flux et la utr() qui permet en revanche

la lecture d'un ou plusieurs caractères par le "buffer" (tampon) de réception du port. Nous avons en outre appris (en analysant des "listings", ce que nous allons faire à nouveau dans cette troisième partie) une foule d'autres instructions de portée générale.

Par exemple la prtf() utilisée pour envoyer des informations de débogage à travers l'UART2, la dlys() qui exécute des pauses de quelques secondes dans l'exécution des "scripts", des procédures de gestion des flux, etc. Dans les "scripts" jusqu'alors proposés, le GM47 a toujours été utilisé comme un microcontrôleur normal. A partir de maintenant, nous commençons au contraire à analyser et à tirer profit toutes les fonctions spécifiques, les caractéristiques et les services GSM que le module est en mesure de nous offrir.

Script numéro 5 : connexion GSM et attente d'appel

Dans cet exemple nous allons voir comment il est possible de vérifier, à l'intérieur d'un "script", si le module est connecté au réseau GSM et si un appel se présente (dans ce cas l'ID de l'appelant est extraite). Rappelons que les fonctions GSM sont réalisées directement par le GM47 ; par conséquent, il n'est pas nécessaire d'écrire dans les "scripts" les codes qui les appelle : ainsi, le programmateur voit sa tâche allégée de la gestion des procédures GSM dont la réalisation serait plutôt complexe.

Le module offre aux "scripts" la possibilité de s'interfacer avec les fonctions GSM, de réclamer l'exécution d'opérations et d'en recevoir des signalisations ou des informations.

En particulier, pour l'échange bilatéral d'informations entre le "script" et le système qui s'occupe des fonctions GSM, des adresses de mémoire (nommées "system status bytes") ont été prévues : elles contiennent des données codées. Le logiciel gérant les fonctions GSM écrit et met à jour ces valeurs ; les "scripts" peuvent en revanche les lire et, en fonction des données lues, agir comme il se doit.

L'instruction utilisée pour la lecture d'un octet est int gtb(int cod) qui restitue la valeur prise par l'octet adressé par le paramètre cod. Par exemple, pour lire l'octet identifiant l'état de la connexion au réseau GSM, il est nécessaire d'utiliser l'instruction gtb() en lui donnant comme paramètre la valeur 3 ; pour lire l'octet identifiant l'état des appels entrants, il faut utiliser gtb(5), etc.

La signification de la valeur restituée change selon l'octet lu : par exemple, pour l'instruction gtb(3), si le 0 est restitué, cela signifie que le module n'est pas en mesure de se connecter au réseau GSM, 1 que l'enregistrement au réseau a été effectué correctement, 2 que la connexion n'est pas encore réalisée mais que le GM47 termine la recherche, 3 que l'accès au réseau est annulé, 4 que le réseau est inconnu, 5 qu'une connexion en "roaming" est réalisée, etc. En revanche, dans le cas gtb(5), 0 signifie appel non actif, 1 appel entrant ("ringing"), etc.

L'instruction gtb() étant analysée, passons au "listing" proprement dit et à un exemple pratique de son utilisation. La première opération exécutée par le "script" est la définition des variables entières contenant les codes à attribuer comme paramètres à la gtb() et les valeurs possibles que peut restituer l'instruction.

Ensuite, à l'intérieur d'un cycle for, l'état de la connexion au réseau GSM est testé trente fois. Pour cela, c'est l'instruction gtb() qui est utilisée : sa valeur restituée est comparée avec les codes possibles et des informations de débogage sont envoyées.

En outre, si la connexion au réseau GSM a été faite, le cycle se termine par un break. Quand on est sorti du cycle for, si la connexion a eu lieu, le logiciel reste en attente d'un appel. A l'intérieur d'un deuxième cycle infini (instruction while(1)) et toujours à travers l'instruction gtb(), l'octet identifiant l'état des appels est lu : si cette valeur est égale à 1 (variable RINGING), l'ID de l'appelant est lue, puis cette information est transmise par l'UART2. Pour extraire le numéro de l'appelant on se sert de l'instruction void clip(char *num, int len) qui mémorise l'ID dans la séquence num (le second paramètre len indique le nombre de caractères que l'instruction copiera ; notez que la taille maximale est 32 caractères).

Si l'information relative à l'ID de l'appelant n'est pas valide (par exemple parce qu'aucun appel n'a lieu), la fonction restitue une séquence vide. Précisons que l'ID est fournie avec le préfixe international mais sans le symbole '+'.

Script numéro 6 : reconnaissance du numéro de l'appelant

Ce "script" étend les fonctions du précédent. En fait, le logiciel attend l'arrivée d'un appel, en extrait l'ID de l'appelant et le compare aux quatre numéros précédemment spécifiés.

Au cas où l'ID est identique au premier numéro, la LED 1 clignote ; si l'ID est identique au deuxième numéro, la LED 2 clignote et ainsi de suite pour les numéros 3 et 4.

Considérons maintenant le "listing" lui-même: après avoir déclaré les variables utilisées pour vérifier l'état de la connexion GSM et la présence d'un appel entrant, quatre séquences sont définies (num1÷num4) contenant les quatre numéros de téléphone avec lesquels effectuer les comparaisons et une séquence (numero) utilisée en revanche pour mémoriser l'ID de l'appelant.

Alors, après avoir désactivé la transmission des infos relatives au réseau GSM sur l'UART2 et initialisé les quatre broches d'E / S comme sorties, le logiciel attend que le GM47 se connecte au réseau.

Pour cela on utilise une boucle for à l'intérieur de laquelle est lu l'octet indiquant l'état du réseau (instruction gtb()) et, dans le cas où il est égal à 1, le cycle s'achève.

Quand le module est connecté au réseau GSM, le logiciel entre dans un deuxième cycle infini au cours duquel l'état des appels est continûment testé. Si un appel entrant est détecté, à travers l'instruction clip() l'ID de l'appelant est extraite puis comparée aux quatre numéros définis au début du "listing".

Si la comparaison aboutit, l'état de la broche d'E / S correspondante est inversé (et la LED s'allume) ; si en revanche elle n'aboutit pas, le cycle infini est réitéré.

Afin d'exécuter les comparaisons entre l'ID de l'appelant et les quatre mots contenant les numéros, on se sert de l'instruction int scmp(char *testo1, char *testo2) qui compare les deux séquences de paramètres et restitue 0 si les deux séquences sont identiques ; un entier supérieur à 0 si testo1 est inférieur à testo2 ou bien un entier inférieur à 0 si testo1 est supérieur à testo2.

Figure 1: “Listing” du “script” 5.

```
/* Script 05: Exemple d'utilisation
   de l'instruction gtb() pour vérifier
   l'état de la connexion au réseau
   GSM et l'arrivée d'un appel */

main() {

      /* Définitions variables de commodité */
      int STATO_RETE=3;
      int NOT_REG_NOT_SEARCHING=0;
      int NET_REGISTERED=1;
      int NET_NOT_REG_SEARCHING=2;
      int NET_REGISTRATION_DENIED=3;
      int NET_UNKNOWN=4;
      int NET_REGISTERED_ROAMING=5;

      int STATO_CHIAMATA=5;
      int RINGING=1;

      int val=0;
      int i=0;
      char numero[33];

      /* Désactive la transmission des
      informations sur le réseau GSM */
      prs(0);

      prtf(«\nConnexion au réseau GSM...\n»);

      for(i=0;i<30;i++) {
            /* Lit l'octet représentant
            l'état de la connexion GSM */
            val=gtb(STATO_RETE);

            /* Vérifie les 5 cas possibles */
            if (val==NOT_REG_NOT_SEARCHING) {
                  prtf(«\nConnexion impossible»);
            }
            if (val==NET_REGISTERED) {
                  prtf(«\nConnecté au réseau GSM\n”);
                  break;
            }
            if (val==NET_NOT_REG_SEARCHING) {
                  prtf(«\nRecherche en cours...»);
            }
            if (val==NET_REGISTRATION_DENIED) {
                  prtf(«\nAccès au réseau annulé»);
            }
            if (val==NET_UNKNOWN) {
                  prtf(«\nRéseau inconnu»);
            }
            if (val==NET_REGISTERED_ROAMING) {
                  prtf(«\nConnexion en roaming»);
            }
            dlys(1);
      }

      /* Si la connexion au réseau GSM
      a été réalisée */
      if (val==NET_REGISTERED) {

            prtf(«\nAttend un appel...\n»);

            /* Attente d'un appel*/
```

```
            while(1) {
                  /* Lit l'octet contenant
                  l'état de l'appel en input */
                  val=gtb(STATO_CHIAMATA);

                  /* Si appel présent */
                  if (val==RINGING) {
                        /* Lit l'ID de l'appelant et le
                        sauvegarde en numéro */
                        clip(numero,32);

                        /* Visualise message et ID
                        de l'appelant */
                        prtf(«\nAppel en entrée
                        du numéro %s\n»,numero);

                        dlys(1);
                  }
            }
      }
      dlys(10);

      /* Réactive la transmission des
      informations sur le réseau GSM */
      prs(1);
}
```

Figure 2 : "Listing" du "script" 6.

```
/* Script 06: Vérification de l'ID de
   l'appel en entrée et clignottement de
   la LED correspondante */

main() {

      /* Définition variables de commodité */
      int STATO_RETE=3;
      int NET_REGISTERED=1;

      int STATO_CHIAMATA=5;
      int RINGING=1;

      int val=0;

      /* Définition des 4 numéros
      utilisés pour les comparaisons*/
      char num1[33]=»393407125506»;
      char num2[33]=»393365678543»;
      char num3[33]=»393472535626»;
      char num4[33]=»393485289402»;

      /* Variables utilisées pour lire
      l'ID de l'appelant */
      char numero[33];

      /* Désactive la transmission des
      informations sur le réseau GSM */
      prs(0);

      /* Initialise 4 E / S comme sorties */
      val = io(2,0,1);
      if (val) prtf(«\nIniz. I/O1 OK»);
      else prtf(«\nIniz. I/O1 non OK»);
      val = io(2,1,1);
      if (val) prtf(«\nIniz. I/O2 OK»);
```

```
      else prtf(«\nInizi. I/O2 non OK»);
      val = io(2,2,1);
      if (val) prtf(«\nIniz. I/O3 OK»);
      else prtf(«\nIniz. I/O3 non OK»);
      val = io(2,3,1);
      if (val) prtf(«\nIniz. I/O4 OK»);
      else prtf(«\nIniz. I/O4 non OK»);

      prtf(«\nConnexion au réseau GSM...\n»);

      /* Attend connexion au réseau GSM */
      for(;;) {
            /* Lit l'octet représentant
            l'état de la connexion GSM */
            val=gtb(STATO_RETE);

            if (val==NET_REGISTERED) {
                  prtf(«\nConnecté au réseau GSM\n");
                  break;
            }
      }

      prtf(«\nAttend appel...\n»);
      for(;;) {

            /* Lit l'octet contenant
            l'état de l'appel en input */
            val=gtb(STATO_CHIAMATA);

            /* Si appel présent */
            if (val==RINGING) {
                  /* Lit son ID */
                  clip(numero,32);

                  prtf(«\nAppel entrant du
numéro %s\n»,numero);

                  /* Compare avec les numéros
                  mémorisés */
                  if (scmp(num1,numero)==0) {
                        /* Lit état de I/O 1 */
                        val = io(0,0,0);
                        /* Inverse état de I/O 1 */
                        if (val) io(1,0,0);
                        else io(1,0,1);
                        dlys(1);
                  }

                  if (scmp(num2,numero)==0) {
                        /* Lit état de I/O 2 */
                        val = io(0,1,0);
                        /* Inverse état de I/O 2 */
                        if (val) io(1,1,0);
                        else io(1,1,1);
                        dlys(1);
                  }

                  if (scmp(num3,numero)==0) {
                        /* Lit état de I/O 3 */
                        val = io(0,2,0);
                        /* Inverse état de I/O 3 */
                        if (val) io(1,2,0);
                        else io(1,2,1);
                        dlys(1);
                  }

                  if (scmp(num4,numero)==0) {
                        /* Lit état de I/O 4 */
```

```
                        val = io(0,3,0);
                        /* Inverse état de I/O 4 */
                        if (val) io(1,3,0);
                        else io(1,3,1);
                        dlys(1);
                  }
            }
      }

      /* Réactive la transmission des
      informations sur le réseau GSM */
      prs(1);

}
```

Figure 3: “Listing” du “script” 7

```
/* Script 07: Exemple d'utilisation
   de l'instruction gtf() pour vérifier
   la réception d'un nouveau SMS */

main() {
      /* Définition des variables de commodité */
      int NUOVOSMS=29;

      int STATO_RETE=3;
      int NET_REGISTERED=1;

      int val;

      /* Désactive la transmission des
      informations sur le réseau GSM */
  prs(0);

      prtf(«\nConnexion au réseau GSM...\n»);

      for(;;) {
            /* Lit l'octet représentant
            l'état de la connexion GSM */
            val=gtb(STATO_RETE);
            /* Vérifie connexion au réseau */
            if (val==NET_REGISTERED) {
                  prtf(«\nConnecté au réseau GSM\n”);
                  break;
            }
      }

      prtf(«\nAttend arrivée d'un SMS...\n»);

      while(1) {

            /* Teste le flag représentant
            l'état de l'arrivée de nouveaux SMS */
            if(gtf(NUOVOSMS))
                  prtf(«\nNouveau SMS arrivé\n»);

      }

      /* /* Réactive la transmission des
      informations sur le réseau GSM */
  prs(1);

}
```

Figure 4: "Listing" du "script" 8.

```
/* Script 08: Exemple d'envoi de SMS */

main() {
      /* Définition variables de commodité */
      int STATO_RETE=3;
      int NET_REGISTERED=1;

      int i=0;
      int val=0;
      int aterr;
      int smserr;

      /* Définition du destinataire
      dell'SMS */
      char numero[15]=»393407125506»;

      /* Définition du texte du SMS */
      char testo[160]=»Testo dell'SMS»;

      /* Désactive la transmission des
      informations sur le réseau GSM */
   prs(0);

      prtf(«\nConnexion au réseau GSM...\n»);

      /* Attend connexion au réseau GSM */
      for(i=0;i<20;i++) {

            /* Lit l'octet représentant
            l'état de la connexion GSM */
            val=gtb(STATO_RETE);

            /* Vérifie la connexion au réseau */
            if (val==NET_REGISTERED) {
                  prtf(«\nConnecté au réseau GSM\n");
                  break;
            }
            dlys(1);

      }

      /* Si le module est connecté au
      réseau GSM tente l'envoi du SMS */
      if (val==NET_REGISTERED) {
            prtf(«\nEnvoi le SMS.\n»);

            /* Crée un canal pour l'envoi
            des commandes AT */
            aterr = atcrt();
            /* Si la longueur du SMS est
            inférieure ou égale à 160 caractères */
            if (slen(testo)<161) {
                  /* Visualise informations SMS */
                  prtf(«\nDestinataire: %s»,numero);
                  prtf(«\nTexte: %s»,testo);
                  prtf(«\nNombre caractères du SMS:
%d»,slen(testo));

                  /* Envoi SMS */
                  smserr=smss(numero,testo,145,
slen(numero),slen(testo));

                  /* Contrôle si SMS envoyé
                  correctement ou non */
                  if (smserr==0) {
                        prtf («\nEnvoi SMS OK\n");
                  }
                  else {
                        prtf(«\nErreur envoi SMS\n»);
                  }
```

Script numéro 7 : attente SMS

Dans le "script" numéro 5 nous avons introduit la fonction gtb() et le concept des "system status bytes" utilisés pour vérifier l'état des fonctions GSM. En dehors de ces cellules de mémoire, on trouve également des "flags" (nommés "system status flags") utilisables eux aussi pour vérifier l'état des sections du GM47. Comme pour les "system status bytes", les valeurs prises par les "flags" sont réglées automatiquement par le logiciel principal gérant le module ; les "scripts" peuvent seulement lire les "flags" et, une fois leur état vérifié, agir comme ils le doivent.

La différence entre les octets et les "flags" réside seulement dans la mémoire utilisée : en effet, les octets sont composés de huit bits et peuvent prendre plusieurs valeurs.

Les "flags" sont en revanche composés d'un bit seulement et ils ne peuvent donc prendre que deux états : true ou bien false. L'instruction permettant de lire un "flag" est la int.gtf(int cod) où le paramètre cod indique quel "flag" lire. La fonction restitue 1 si le "flag" vaut true ; 0 si le "flag" est paramétré comme false.

Le paramètre cod peut prendre différentes valeurs : les plus intéressantes sont celles comprises entre 0 et 3 et la 29. Les quatre premières permettent d'aller vérifier comment ont été configurées les broches d'E / S.

Par exemple, l'instruction gtf(0) renvoie 0 si la broche IO1 a été programmée comme entrée ; 1 si IO1 est une sortie. Les broches restantes peuvent être vérifiées en attribuant les codes 1, 2 et 3 comme paramètres.

En revanche le "flag" 29 peut être utilisé pour vérifier si un nouveau SMS a été reçu. L'instruction gtf(29) renvoie 1 si un nouveau message est présent, sinon 0. Notez que l'exécution de gtf(29) cause le "reset" du "flag" correspondant ; par conséquent, si un nouveau SMS est reçu, la fonction restitue 1 seulement la première fois qu'elle est appelée. Dans tous les autres cas, elle restitue 0 jusqu'à ce qu'un second message soit reçu.

L'instruction gtf() étant vue, passons au "script" suivant. Il s'agit d'un logiciel tout simple permettant au micro de vérifier si un nouveau SMS a été reçu. Les premières opérations exécutées

```
                    dlys(3);

              }

              /* Signale texte SMS trop long */
              else {
                    prtf(«\nLe texte ne peut'
dépasser 160 caractères\n»);
              }

              /* Ferme le canal AT */
              aterr = atdst();
       }

       /* Signale SMS non envoyé */
       else {
              prtf(«\nImpossible envoyer l'SMS»);
       }

       dlys(5);

       /* Réactive la transmission des
       informations sur le réseau GSM */
    prs(1);

}
```

sont la déclaration des variables de commodité contenant les codes à communiquer à la gtf() et les codes des valeurs restituées.

Notez que l'utilisation de ces variables n'est pas obligatoire : en effet, il est possible de communiquer aux fonctions gtf() et gtb() directement les codes de l'octet ou du "flag" que l'on veut lire et comparer les valeurs restituées aux codes correspondants.

L'utilisation des variables rend donc le "listing" plus lisible et plus compréhensible en vue d'éventuelles modifications ultérieures. Le "script" attend que le GM47 se connecte au réseau GSM, puis il se met en attente d'un SMS entrant.

La situation est gérée à l'intérieur d'un cycle infini où est exécutée l'instruction gtf(), passant comme paramètre la valeur 29 (contenue dans la variable NUOVOSMS) et où est vérifiée la valeur restituée.

Script numéro 8 : envoi SMS

Au sein de ce nouvel exemple nous verrons comment il est possible de gérer l'envoi d'un SMS en utilisant un "script". Avant d'analyser la méthode à utiliser, il est cependant nécessaire de faire un bref rappel sur les canaux AT. En effet, les "scripts" pour demander au GM47 l'exécution de fonctions GSM particulières, utilisent des canaux dans lesquels ils envoient des commandes AT adéquates et par lesquels ils reçoivent des réponses correspondant à ces commandes.

La communication à l'intérieur des canaux est gérée directement par les instructions utilisées ; on demande simplement au programmateur d'ouvrir un canal avant de l'utiliser (en appelant la fonction adéquate) et de le fermer quand on n'en a plus besoin.

L'instruction utilisée pour ouvrir un canal AT est la int.atcrt() qui ne prévoit pas de paramètres et restitue 0 si le canal a été ouvert.
En revanche, la fonction qui ferme un canal est la int atdst() ; elle ne prévoit pas non plus de paramètres mais restitue 0 en cas de succès de la fermeture du canal.

La gestion des canaux AT étant expliquée, passons à l'analyse de l'envoi d'un SMS directement avec un "script". L'instruction à utiliser est la int smss(char *d, char *t, int tipo, int lend, int lent) où le mot d indique le numéro de téléphone du destinataire du message, le mot t le texte du SMS et l'entier tipo le type du numéro destinataire (pour nous il est égal à 145).

Enfin les deux entiers lend et lent indiquent le nombre de caractères respectivement du numéro destinataire et du texte. La fonction smss() restitue 0 si l'opération d'envoi du SMS a abouti ; sinon une valeur différente de 0.

Mais penchons-nous sur le "listing" : après le définition des variables d'utilité, la séquence numero, contenant le destinataire du SMS et la séquence texte, contenant le message, sont initialisées (comme pour le "script" numéro 6, dans ce cas aussi, le numéro est exprimé avec le préfixe international mais sans le symbole '+').

Puis le "script" attend que le module se connecte au réseau GSM ; en cas d'issue positive, il crée un canal AT, vérifie que le texte du message se compose bien d'un nombre inférieur ou égal à 160 caractères (l'instruction utilisée est la slen(), elle sera analysée dans la quatrième partie) et, après avoir envoyé sur l'UART2 les données relatives au SMS (destinataire, texte et longueur du texte), il tente d'envoyer le message (le résultat de la tentative est envoyé sur l'UART2 pour débogage).

Les lignes de codes restantes gèrent l'envoi des informations de débogage concernant le cas où le texte du message est trop long et au cas où il est impossible d'envoyer le SMS parce que le module ne parvient pas à se connecter au réseau GSM.

Notez en outre la présence de l'instruction atdst() qui, dans le cas où le canal AT a été ouvert correctement, le ferme avant de conclure l'exécution du "script". Par commodité nous n'avons pas prévu le cas où il y aurait des problèmes durant l'ouverture et la fermeture du canal AT ; le programme est en effet fort simple et un seul canal est utilisé. Néanmoins, dans un projet plus complexe, il faudrait gérer aussi cette situation afin d'éviter que des erreurs ne se produisent pendant l'exécution des "scripts".

Comment construire ce montage ?

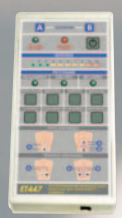
Tout le matériel nécessaire pour construire la platine d'expérimentation ET502 pour le module GSM GM47, ainsi que le module GM47, l'antenne plate FME et l'alimentation secteur, sont disponibles chez certains de nos annonceurs. Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/les_circuits_imprimés.asp.

Les composants programmés sont disponibles sur www.electronique-magazine.com/mc.asp. ◆

L'ÉTÉ EN FORME

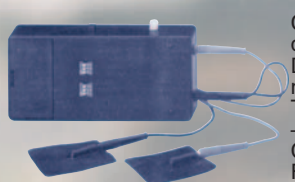
UN ÉLECTROSTIMULATEUR BIPHASIQUE ABDOMINAL



Cet électrostimulateur neuromusculaire a été conçu spécialement pour faire travailler les **abdominaux** en entraînement passif (allongé sur son lit !) ou en mixte (en faisant du footing… ou la cuisine !) puisqu'il est portatif. Il comporte quatre programmes correspondant à quatre traitements : idéal pour se maintenir en forme ou pour entretenir son esthétique quand on n'a pas trop de temps.

ET447 ..... Kit complet avec batterie et électrodes ..... 120,00 €

STIMULATEUR ANALGESIQUE



Cet appareil permet de soulager des douleurs tels l'**arthrose** et les **céphalées**. De faible encombrement, ce kit est alimenté par piles incorporées de 9 volts.
Tension électrode maximum :
–30 V - +100 V.
Courant électrode maximum : 10 mA.
Fréquences : 2 à 130 Hz.

EN1003 .... Kit complet avec boitier ..... 36,30 €

MAGNETOTHERAPIE BF (DIFFUSEUR MP90) A HAUT RENDENDEMENT



Très complet, ce kit permet d'apporter tous les "bienfaits" de la magnétothérapie BF. Par exemple, il apporte de **l'oxygène** aux cellules de l'organisme, élimine la **cellulite**, les toxines, les **états inflammatoires**, principales causes de douleurs musculaires et osseuses.
Fréquences sélectionnables :
6.25 - 12.5 - 25 - 50 - 100 Hz.
Puissance du champ magnétique :
20 - 30 - 40 Gauss.
Alimentation : 220 VAC.

EN1146 .... Kit complet avec boîtier et diffuseur ..... 165,60 €
MP90 ..... Diffuseur supplémentaire ..... 22,15 €

ELECTROSTIMULATEUR NEUROMUSCULAIRE



Cet appareil, moderne et d'une grande diversité d'emplois, répond aux attentes des athlètes, aux exigences des professionnels de la remise en forme comme aux espoirs de tous ceux qui souhaitent améliorer leur aspect physique.
Il propose plusieurs programmes de musculation, d'amincissement, de tonification, de préparation et de soin des athlètes.

ET480..... Kit complet avec boîtier, batterie et électrodes ... 545,00 €

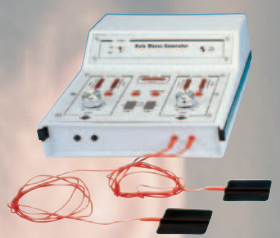
LA IONOTHERAPIE: TRAITER ELECTRONIQUEMENT LES AFFECTIONS DE LA PEAU



Pour combattre efficacement les affections de la peau, sans aucune aide chimique, il suffit d'approcher la pointe de cet appareil à environ 1 cm de distance de la zone infectée. En quelques secondes, son "souffle" **germicide** détruira les bactéries, les champignons ou les germes qui sont éventuellement pré-

EN1480 .... Kit étage alimentation avec boîtier ..... 80,00 €
EN1480B . Kit étage voltmètre ..... 24,00 €
PIL12.1 .... Batterie 12 volts 1,3 A/h ..... 15,10 €

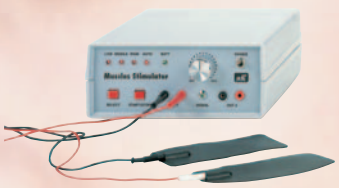
UN GÉNÉRATEUR D'ONDES DE KOTZ POUR SPORTIFS ET KINÉS



Le générateur d'ondes de Kotz est utilisé en médecine pour la récupération musculaire des personnes ayant eu un **accident** ou une maladie et qui sont donc restées longtemps inactives, comme pour le sport ou l'esthétique corporelle afin de tonifier et raffermir les muscles sains.

EN1520-1521 ..... Kit complet avec boîtier, plaques et bat ..... 220,00 €

STIMULATEUR MUSCULAIRE



Tonifier ses muscles sans effort grâce à l'électronique. Tonifie et renforce les muscles (4 électrodes).
Le kit est livré complet avec son coffret sérigraphié mais sans sa batterie et sans électrode.

EN1408 ..... Kit avec boîtier ..... 96,35 €
Bat. 12 V 1.2 A ..... Batterie 12 V / 1,2 A ..... 15,10 €
PC1.5 ..... 4 électrodes + attaches ..... 28,00 €

MAGNETOTHERAPIE BF



Cet appareil électronique permet de se maintenir en bonne santé, parce qu'en plus de soulager les **problèmes infectieux**, il maintient nos cellules en bonne santé.
Il réussit à **revitaliser les défenses immunitaires** et accélère la **calcification** en cas de facture osseuse. Effet sur le système nerveux. Fréquence des impulsions : de 156 à 2500 Hz. Effet sur les tissus osseux. Effet sur l'appareil digestif. Effet sur les inflammations. Effet sur les tissus. Effet sur le sang. Largeur des impulsions : 100 µs. Spectre de fréquence : de 18 MHz à 900 MHz.

EN1293 .... Kit complet avec boîtier et 1 nappe ..... 158,55 €
PC1293 .... Nappe supplémentaire ..... 31,00 €

MAGNETOTHERAPIE VERSION VOITURE



La magnétothéraphie est très souvent utilisée pour soigner les maladies de notre organisme (**rhumatismes, douleurs musculaires, arthroses lombaire et dorsales**) et ne nécessite aucun médicament, c'est pour cela que tout le monde peut la pratiquer sans contre indication. (Interdit uniquement pour les porteurs de Pace-Maker.

EN1324 .... Kit complet avec boîtier..... 66,50 €
..... et une nappe version voiture
PC1324 .... Nappe supplémentaire..... 27,50 €

DIFFUSEUR POUR LA IONOPHORÈSE



Ce kit paramédical, à microcontrôleur, permet de soigner l'**arthrite**, l'**arthrose**, la **sciatique** et les **crampes** musculaires. De nombreux thérapeutes préfèrent utiliser la ionophorese pour inoculer dans l'organisme les produits pharmaceutiques à travers l'épiderme plutôt qu'à travers l'estomac, le foie ou les reins. La ionophorèse est aussi utilisée en esthétique pour combattre certaines affections cutannées comme la cellulite par exemple.

EN1365 .... Kit avec boîtier, hors batterie et électrodes ..... 95,60 €
PIL12.1 .... Batterie 12 V 1,3 A/h ..... 15,10 €
PC2.33x ... 2 plaques conduct. avec diffuseurs ..... 13,70 €

Photos non contractuelles. Publicité valable pour le mois de parution. Prix exprimés en euro toutes taxes comprises. Sauf erreurs typographiques ou omissions.

PASSEZ VOS COMMANDES DIRECTEMENT SUR NOTRE SITE : www.comelec.fr

COMELEC

Tél. :04.42.70.63.90 Fax :04.42.70.63.95

www.comelec.fr CD 908 - 13720 BELCODENE

DEMANDEZ NOTRE CATALOGUE 96 PAGES ILLUSTRÉES AVEC LES CARACTÉRISTIQUES DE TOUS LES KITS

Expéditions dans toute la France.Moins de 5 Kg : port 8,40 €. Règlement à la commande par chèque, mandat ou CB. Bons administratifs acceptés.
De nombreux kits sont disponibles, envoyez nous votre adresse et cinq timbres, nous vous ferons parvenir notre catalogue général de 96 pages.

COMELEC 06/2005

Un contrôle de volume IR

Ce système de réglage de volume à infrarouges concerne tout appareil audio stéréophonique. Grâce à sa distorsion harmonique très faible et à sa diaphonie de 100 dB, ce montage peut être utilisé même avec une chaîne Hi-Fi de haut de gamme.



CARACTÉRISTIQUES TECHNIQUES

- Alimentation : 9 à 12 VDC
- Consommation : 15 mA
- Niveau de signal stéréo E / S : 2 Vrms max
- Champ d'atténuation : 0 dB à –78 dB
- Indication de variation du volume par LED
- Diaphonie 100 dB
- Distorsion harmonique maximale 0,005%
- Portée : environ 20 mètres
- TX de télécommande IR compatibles : EK162 préconisé, mais aussi EV8049, EV8051
- Dimensions : 80 x 55 x 33 mm.

La possibilité de régler à distance le volume de sa chaîne Hi-Fi (sans bouger de son fauteuil) est vraiment un luxe très appréciable. Or, si vous faites partie des audiophiles n'acceptant dans leur chaîne stéréophonique que des éléments séparés (préamplificateur et amplificateur), il est bien possible que votre installation ne comporte pas de télécommande. L'appareil que nous vous proposons ici de construire est un contrôleur de volume que vous devrez, justement, insérer entre la sortie du préampli et l'entrée de l'ampli de puissance final, comme le montre la figure 6. Précisons qu'il fonctionne au moyen d'un faisceau de rayons infrarouges (donc pendant un enregistrement à partir du tuner FM, pas de risque d'interférences radio). De plus, si la vue d'une énorme «palette» de télécommande vous donne des (rêves de bons vieux) boutons, soyez rassuré : ce contrôle de volume ne comporte que deux touches, UP (volume +) et DOWN (volume –), comme le montre la figure 5. Terminons cette introduction en précisant que si la programmation des codes vous est fastidieuse (aussi !), eh bien vous pourrez utiliser le code par défaut à la fois pour ce contrôle IR de volume (RX) EV164 et pour le TX préconisé EV162 : oui, c'est le même !

Le schéma électrique

La partie analogique du montage (voir figure 1) utilise un circuit intégré **Toshiba TC9413AP**, normalement mis en œuvre pour le réglage de volume et de balance des canaux dans les appareils de Hi-Fi domestiques.

C'est un composant présentant de hautes qualités audio : 100 dB de diaphonie entre les canaux, distorsion harmonique maximale 0,005 % ; ces valeurs sont atteintes grâce à l'utilisation de résistances «audiophiles» au silicium polycristallin dans les deux potentiomètres numériques affectés aux canaux droit et gauche (chacun se compose d'un pont de 61 résistances en série dont les extrémités sont reliées à la broche IN et à la broche A-GND, alors que le curseur OUT peut se déplacer sur 62 positions différentes correspondant aux nœuds entre les résistances. L'échelle des valeurs qu'il parcourt n'est pas linéaire : la puce utilise 44 pas de 1 dB entre 0 et –44 dB et 17 pas de 2 dB entre –44 dB et –78 dB (le dernier pas correspond à la fonction «mute», muet). Le **TC9413AP** opère son contrôle au moyen d'une interface de communication série constituée d'un bus à trois lignes

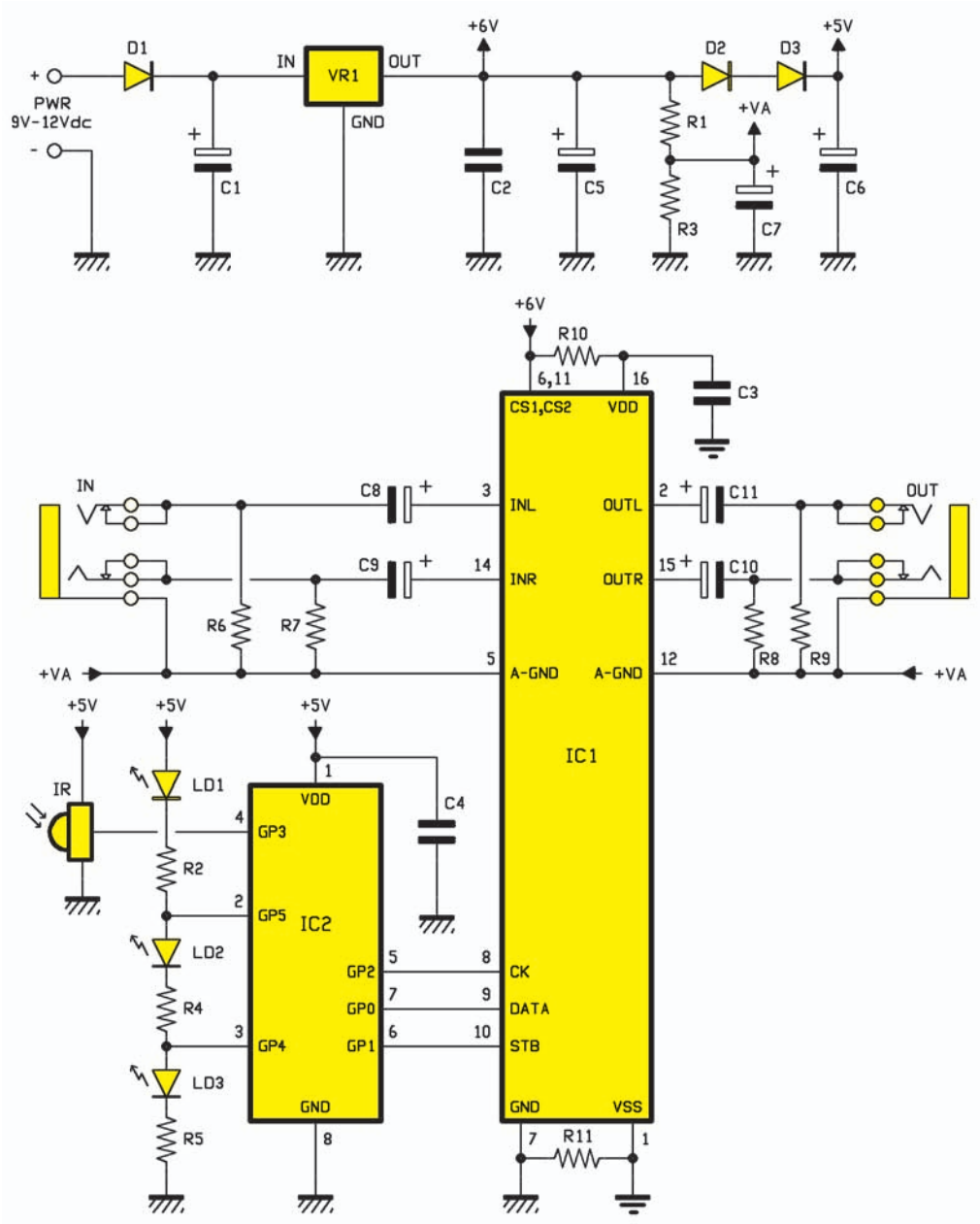


Figure 1: Schéma électrique du contrôle de volume IR.

correspondant aux entrées CK (broche 8), DATA (broche 9) et STB (broche 10); ces broches sont reliées respectivement aux lignes GP2 (broche 5), GP0 (broche 7) et GP1 (broche 6) du micro PIC12F629. Ce dernier s'occupe de la gestion du circuit tout entier, y compris le décodage des signaux à infrarouges reçus par le module intégré **IR38DM**. Celui-ci est un circuit intégré complet avec tous ses étages, prêt pour le réception des signaux infrarouges: il comprend en particulier un filtre optique, une photodiode IR et un étage amplificateur / quadrateur.

Chaque fois que la photodiode est frappée par un signal IR modulé, à sa sortie se trouvent des impulsions logiques envoyées au micro (à l'entrée GP3, broche 4, précisément): cette broche est normalement au niveau logique 1 mais, lorsque le module récepteur détecte les rayons infrarouges provenant du TX, la broche 4 prend le niveau logique 0.

Le programme résident lance alors le sous-programme de décodage laquel analyse la donnée série et, si la compatibilité avec la partie fixe du code est confirmée, il poursuit en élaborant la partie variable puis envoie au TC9413AP les données nécessaires au réglage du volume. Dans la donnée série est présente la valeur binaire à 6 bits correspondant à une valeur d'atténuation particulière.

Si, par exemple, nous maintenons pressé le poussoir gauche du TX, le PIC produit plusieurs paquets de données, régulièrement cadencés, différant par les valeurs d'atténuation à imposer au signal audio élaboré par la puce Toshiba. La valeur d'atténuation de départ est celle imposée par la dernière commande envoyée et tous les paquets qui suivent l'augmentent d'un pas; si on presse le poussoir droit, c'est le contraire qui se passe.

En plus de la gestion du contrôle du circuit intégré audio, le micro s'occupe de la signalisation visuelle de l'état de réglage du volume: en effet, les LED LD1, LD2 et LD3 indiquent visuellement la direction de la

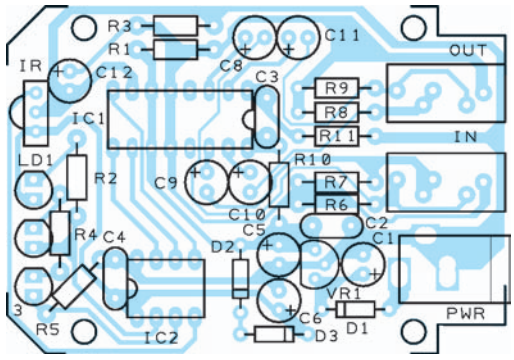


Figure 2a : Schéma d'implantation des composants du contrôle de volume IR.

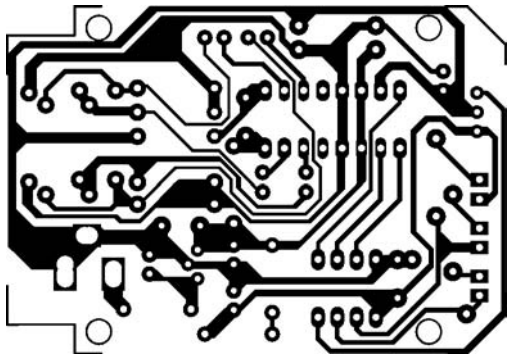


Figure 2b : Dessin, à l'échelle 1, du circuit imprimé du contrôle de volume IR.

variation ; quand le volume augmente les LED commencent à s'allumer une après l'autre de gauche à droite et, si le volume décroît, elles s'allument dans le sens contraire. L'atteinte de l'atténuation maximale (100 dB, pratiquement un niveau de sortie nul) est mis en évidence par l'allumage fixe de LD1 ; avec l'atténuation minimale (0 dB, pratiquement aucune modification du niveau) LD3 reste allumée fixe. Pour piloter les trois LED, on n'utilise que deux sorties du micro, GP4 et GP5. Comme le montre le schéma électrique

Liste des composants

R1 ..... 1 k
R2 ..... 1 k
R3 ..... 1 k
R4 ..... 1 k
R5 ..... 1 k
R6 ..... 100 k
R7 ..... 100 k
R8 ..... 100 k
R9 ..... 100 k
R10 ... 0
R11 ... 0
C1..... 100 µF 16 V électrolytique
C2..... 100 nF multicouche
C3..... 100 nF multicouche
C4..... 100 nF multicouche
C5..... 47 µF 16 V électrolytique
C6..... 47 µF 16 V électrolytique
C7..... 47 µF 16 V électrolytique
C8..... 4,7 µF 50 V électrolytique
C9..... 4,7 µF 50 V électrolytique
C10 ... 4,7 µF 50 V électrolytique
C11 ... 4,7 µF 50 V électrolytique
D1 ..... 1N4148
D2 ..... 1N4148
D3 ..... 1N4148
VR1 ... 78L06
IR..... IR38DM
IC1..... TC9413AP
IC2..... PIC12F629-EK164
LD1 ... LED 3 mm rouge
LD2 ... LED 3 mm rouge
LD3 ... LED 3 mm rouge

Divers :

1 support 2 x 8
1 support 2 x 4
1 prise de pile
1 prise d'alimentation
2 jacks stéréo femelles
1 boîtier plastique spécifique

Sauf spécification contraire, toutes les résistances sont des 1/4 W à 5 %.

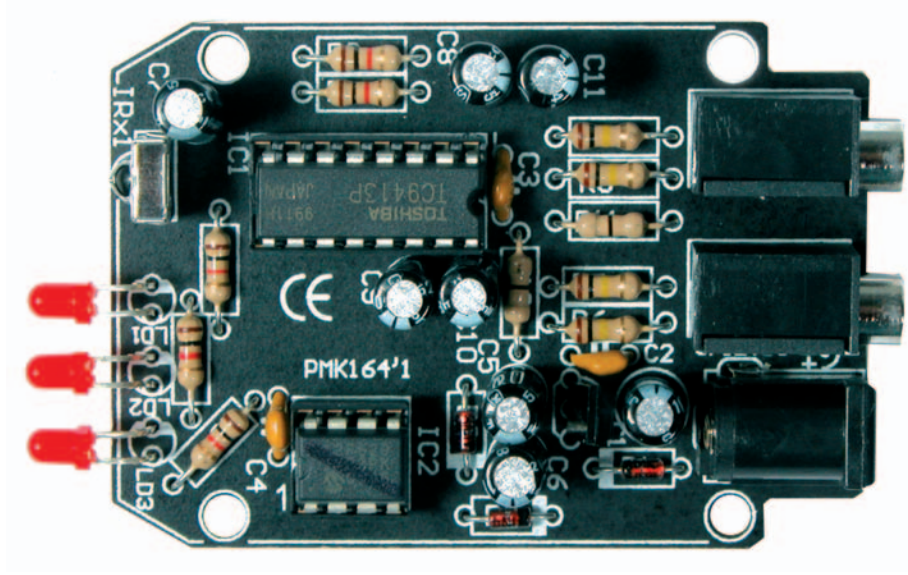


Figure 3 : Photo d'un des prototypes de la platine du contrôle de volume IR.

de la figure 1, les LED sont reliées entre elles et insérées entre le positif d'alimentation et la masse, alors que les deux nœuds sont reliés aux lignes de sortie du micro GP5 (broche 2) et GP4 (broche 3).

En combinant adéquatement les niveaux logiques de ces sorties, il est possible d'activer convenablement les LED : par exemple, pour activer seulement LD2, la ligne GP5 doit être mise au niveau logique haut et GP4 au niveau logique bas ; dans ce cas LD2, avec sa résistance de limitation, est alimentée alors que la tension aux extrémités de LD1 et LD3 est nulle et les LED restent éteintes.

La gestion de ce contrôle de volume étant devenue plus claire, voyons comment le circuit est alimenté : on applique entre les points + et –PWR une tension continue de 9 à 12 V, filtrée par l'électrolytique C1 et appliquée à l'entrée du régulateur VR1. D1 bloque la tension en cas d'inversion accidentelle de polarité.

Le régulateur 7806 fournit une tension particulièrement stable de 6 V alimentant le TC9413A (il accepte entre 6 et 18 V) ; malgré l'utilisation de la tension minimale permise, cette tension ne peut alimenter aussi le micro, lequel requiert le classique 5 V.

Cette dernière tension est obtenue simplement avec deux diodes au silicium D2 et D3 montées en série avec la ligne d'alimentation. Elle alimente aussi le module IR38DM.

Le pont R1 / R3 permet d'obtenir une tension égale à la moitié de celle sortant du régulateur, soit 3 V, ensuite filtrée par C7 et appliquée aux broches 5 et 12 du TC9413A afin de créer la référence analogique (+ VA) des deux potentiomètres électroniques.

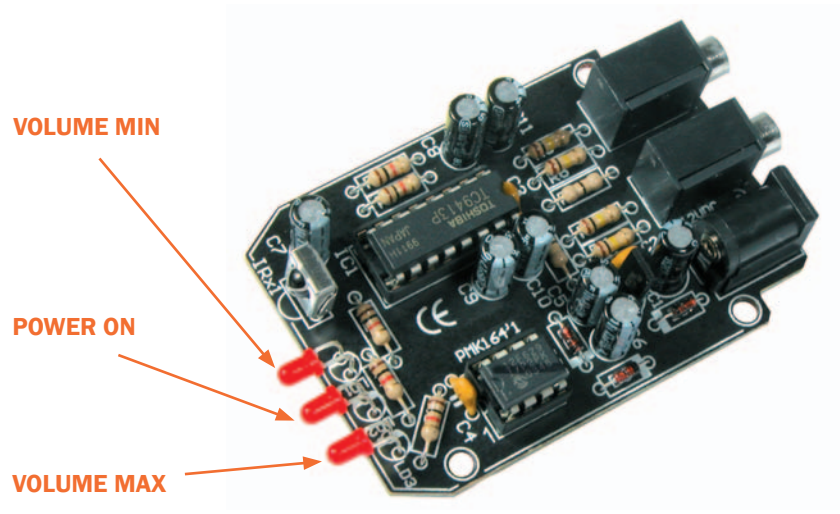


Figure 4 : Signalisations du contrôle de volume IR.

Cette référence est utilisée aussi comme masse pour les signaux audio d'entrée et de sortie.

La réalisation pratique

Les dimensions et le découpage de la platine sont établis pour qu'elle puisse ensuite prendre place dans un boîtier spécifique (voir photo de début d'article). Vous trouverez le dessin à l'échelle 1 du circuit imprimé simple face figure 2b.

Quand vous l'avez devant vous, montez tous les composants (en vous aidant des figures 2a et 3) en commençant par les supports des deux circuits intégrés et en terminant par les périphériques : les deux jacks stéréo, la prise d'alimentation, les trois LED et le capteur IR. Vous n'insèrerez les circuits intégrés dans leurs supports qu'après la dernière soudure terminée.

Procédez ensuite au montage de la petite platine dans le boîtier spécifique (avec capot en plexi rouge favorisant les rayons infrarouges au détriment du reste du spectre lumineux), comme le montre la figure 7. Vous pouvez maintenant connecter l'appareil (qui est bien sûr un récepteur infrarouge) avec des câbles audio de bonne qualité (dotés de connecteurs adéquats, par exemple jacks d'un côté et RCA «cinch» de l'autre) entre le préampli et l'ampli de votre chaîne, comme le montre la figure 6.

Alimentez-le avec une petite alimentation bloc secteur 230 V fournissant de 9 à 12 V continus et débitant quelque 100 mA (attention, prise d'alimentation à positif au centre). Quand le circuit est alimenté, LD2 s'allume et l'appareil est prêt à recevoir les commandes.

Couplez l'appareil avec son émetteur de télécommande IR, le TX «porte-clés» EV162 est le choix le meilleur et le plus économique et procédez aux essais :

Figure 5 : La télécommande la mieux adaptée au pilotage de notre contrôle de volume IR est la EV162 (voir l'article EV161 proposant le RX et le TX qui seul nous intéresse ici).

Il s'agit en effet d'un petit émetteur IR au format porte-clé, contenant cependant un microcontrôleur et pouvant être utilisé sans avoir à effectuer aucune programmation car les codes par défaut sont compatibles avec ceux de notre contrôle de volume IR.

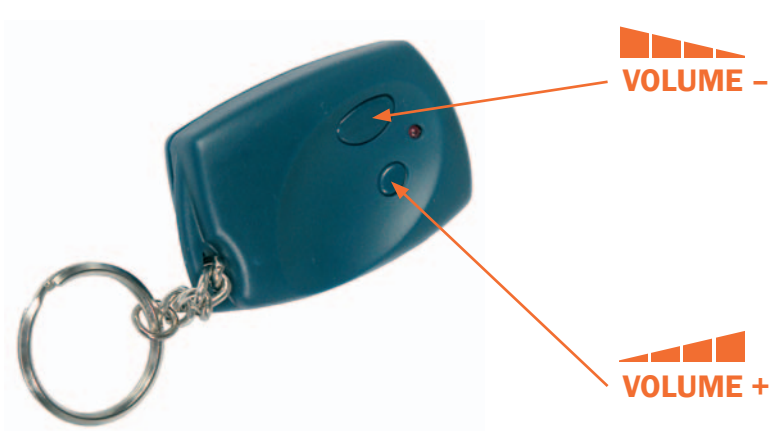
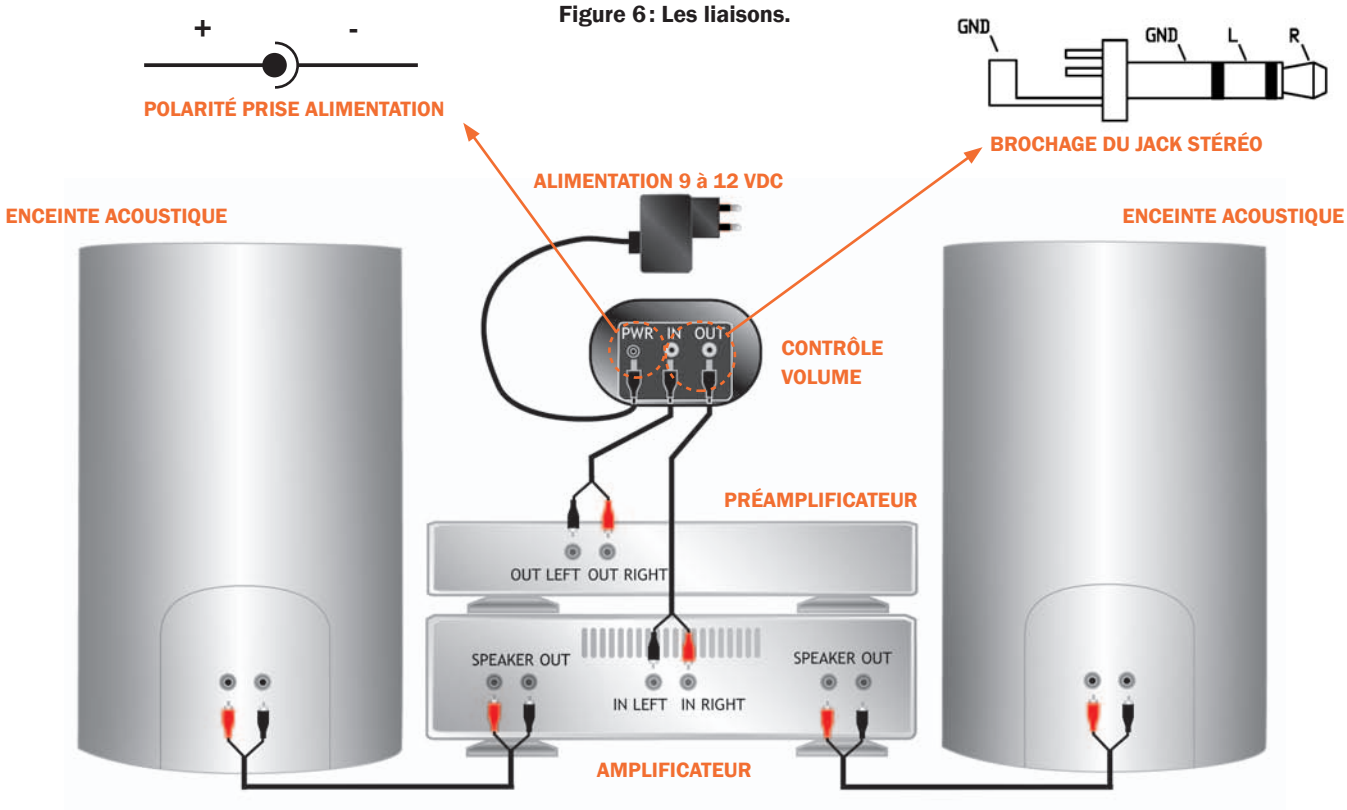


Figure 6 : Les liaisons.



Ce schéma montre comment relier ce contrôle de volume IR à une chaîne Hi-Fi à éléments séparés : le contrôle de volume doit être inséré entre le préamplificateur et l'amplificateur au moyen d'un câble et de connecteurs de bonne qualité afin de tirer le meilleur profit des excellentes caractéristiques du circuit.

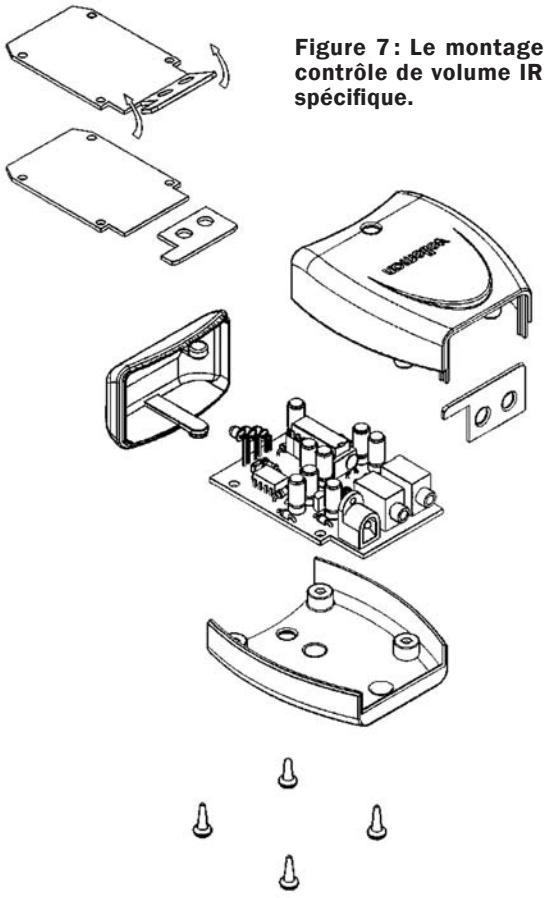


Figure 7 : Le montage de la platine du contrôle de volume IR dans son boîtier spécifique.

mettez initialement les réglages de volume des modules BF à mi course puis effectuez un réglage plus fin.

Si vous pressez le poussoir droit du TX, vous verrez se déplacer l'allumage de la LED de gauche à droite et entendrez le volume du son augmenter dans les enceintes ; l'inverse se produira si vous pressez le poussoir gauche.

Si en revanche vous utilisez notre TX IR à quinze canaux, vous devrez utilisez seulement les premier et deuxième canaux.

Comment construire ce montage ?

Tout le matériel nécessaire pour construire ce contrôle de volume IR EV164, ainsi que le TX de télécommande IR EV162, est disponible chez certains de nos annonceurs. Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp.

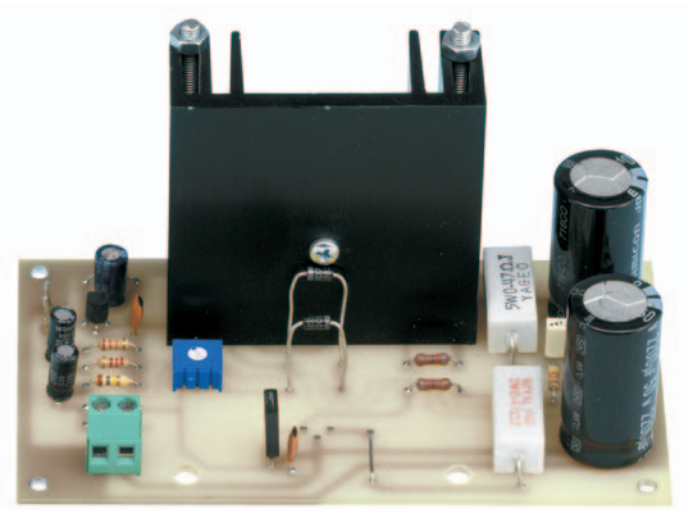
Les programmes disponibles sont sur www.electronique-magazine.com/mc.asp. ◆

EN1616-1617

Un amplificateur Hi-Fi

de 10 WRMS sur 8 ohms

Après que nous vous ayons récemment présenté les schémas d'une dizaine de préamplificateurs BF simples n'utilisant que deux transistors ou FET, vous avez étés nombreux à nous demander de proposer aussi un étage final de puissance simple et entièrement à transistors. Le voici.



CARACTÉRISTIQUES TECHNIQUES

- Alimentation simple: 30 à 33 V environ
- Courant de repos: 70 mA
- Courant à la puissance maximale: 450 mA
- Puissance maximale sur 8 ohms: 12 WRMS environ
- Bande passante à –3 dB: 10 Hz à 40 kHz
- Amplitude maximale du signal d'entrée: 1,6 Vpp
- Distorsion harmonique: 0,4% environ

Votre demande s'accorde particulièrement avec notre ligne éditoriale car, vous le savez puisque vous nous lisez, nous nous sommes imposé une vocation didactique: faire, oui, mais d'abord comprendre comment ça fonctionne! Or la plupart des amplificateurs d'aujourd'hui sont à circuit intégré monolithique: la représentation symbolique est constituée de quatre broches, tout cela pour une multitude de fonctions (allez donc y voir et y comprendre quelque chose!). Avec un amplificateur à transistor, par contre, on peut tout expliquer et tout comprendre. C'est ce que nous allons faire.

Les schémas électriques

Le schéma électrique de l'amplificateur de puissance est visible figure 2 et celui de son alimentation figure 3. La figure 2 donne le schéma électrique d'un amplificateur monophonique (pour construire un amplificateur stéréophonique, il faudrait en monter deux identiques, un pour la voie droite et un pour la voie gauche et, bien sûr, les "attaquer" avec un signal d'entrée lui-même stéréo). Notez que l'alimentation secteur 230 V a été dimensionnée pour pouvoir alimenter deux amplificateurs mono, soit un amplificateur stéréo: pour cela, elle peut débiter jusqu'à 0,9 A sous 32 Vcc.

L'amplificateur

Nous appliquons sur l'entrée (dans le schéma électrique en bas à gauche), par l'intermédiaire d'un petit câble blindé, le signal prélevé à la sortie de n'importe quel préamplificateur ou lecteur CD. À travers l'électrolytique C1, le signal atteint la base du transistor préamplificateur NPN TR1 puis ce signal, préamplifié, sort du collecteur pour aller piloter directement la base du pilote TR3 (comme c'est un PNP, il n'amplifie que les demi ondes positives). Les demi ondes négatives, passent par DS1 et DS2 pour atteindre la base du NPN TR2. Les deux transistors pilotes sont directement reliés aux deux transistors finaux de puissance, les NPN TR4 et TR5 BD241.

Le premier couple pilote/final TR3/TR5 n'amplifie que les demi ondes positives et le second TR2/TR4 seulement les

demi ondes négatives. Le signal amplifié est prélevé à travers l'électrolytique C8 de 2 200 µF sur la jonction R11-R12 (chacune de ces résistances fait 0,47 ohm 5 W, ce sont des résistances de puissance en céramique, l'une est reliée à l'émetteur de TR4 et l'autre au collecteur de TR5 : voir figure 5). De C8 le signal BF est dirigé vers le haut-parleur (ou l'enceinte) dont l'impédance de charge devra être de 8 ohms.

La R8 de 47 k et le C7 en série avec elle, reliés à la base de TR1, constituent un réseau de contre-réaction contrôlant le gain de ce premier étage préamplificateur. DS2 et DS1, montées entre les bases de TR2 et TR3, servent de protection thermique (c'est pourquoi leurs boîtiers devront être en contact avec les dissipateurs, ou du moins très proches, afin de recevoir la chaleur dégagée par TR4 et TR5) : voir figures 1 et 5.

En cas d'augmentation anormale de la température des dissipateurs à ailettes (c'est-à-dire de la dissipation des transistors finaux de puissance), ces diodes réduiraient automatiquement le courant de travail, évitant ainsi la destruction des transistors. Les condensateurs C3, C4 et C6 de 100 pF, reliés entre collecteur et base de chaque TR1, TR2 et TR3, servent à éviter que ces derniers n'auto-oscillent intempestivement sur des fréquences ultrasoniques.

L'alimentation secteur

Pour alimenter cet amplificateur, du moins, nous l'avons dit, dans sa version stéréo, il faut un transformateur secteur 230 V de 20 VA environ doté d'un secondaire donnant une tension de 24 VAC pour un courant de 0,9 A (ou plus bien entendu) : nous avons utilisé un transformateur pourvu d'un secondaire double (12 + 12 V) dont la prise centrale n'a pas été connectée (mod. T020-06).

Attention, si vous vous servez d'un transformateur dont les deux secondaires (12 V chacun) sont séparés, vous devrez les relier en série pour obtenir entre les deux extrémités opposées des secondaires les 24 V requis. En version mono un transformateur de 10 VA et dont le secondaire de 24 V (ou les secondaires de 12 + 12 V) ne pourrait fournir que 0,45 A ferait l'affaire.

Cette tension de 24 VAC redressée par le pont RS1 et lissée par l'électrolytique C1 donne une tension de :

(24 x 1,41) –1,4 = 32 Vcc.

Note : le nombre 1,4 soustrait correspond à la chute de tension dans les diodes redresseuses du pont.

Il n'est pas nécessaire que cette tension soit stabilisée : si elle atteint 33, voire 35 Vcc, la puissance de sortie sera supérieure à celle annoncée en début d'article dans les caractéristiques techniques.

La réalisation pratique

Pour réaliser cet amplificateur de puissance en version stéréo, il vous faut deux circuits imprimés EN1616 et un circuit imprimé EN1617 : ce sont tous deux des simple face dont les figures 4b et 6b donnent respectivement les dessins à l'échelle 1.

Si vous observez bien les figures 4a et 5 (pour les platines amplificateurs) et 6a et 7 (pour la platine alimentation), ainsi que les listes des composants, vous ne rencontrerez aucune difficulté.

N'oubliez surtout pas de monter les "straps" filaires sur les deux platines des amplificateurs. Accordez beaucoup d'attention aux soudures (ni court-circuit entre pistes ou pastilles ni soudure froide collée) et à l'orientation (et donc à la polarité) des condensateurs électrolytiques, des diodes, du pont redresseur et des transistors (dont le repère-détrompeur est constitué par le méplat pour TR1 et par la semelle métallique pour les autres).

Pour les deux transistors finaux (quatre en version stéréo) : fixez-les chacun sur son dissipateur à ailettes (quatre dissipateurs en tout en version stéréo) à l'aide d'une vis autotaraudeuse ou d'un boulon (à passer à travers le petit trou de la semelle et dans le petit trou pratiqué dans le dissipateur) ; ensuite enfilez les pattes du transistor dans les trous du circuit imprimé (attention, respectez bien l'orientation EBC, voir figure 7) ; fixez le dissipateur au circuit imprimé par deux boulons longs et enfin seulement soudez les trois pattes du transistor dans les trois trous EBC du circuit imprimé.

En version stéréo, répétez patiemment quatre fois cette procédure.

Les deux diodes de protection (quatre en version stéréo) conserverons toute la longueur de leurs pattes et seront chacune disposées de telle façon que leur boîtier plastique touche ou effleure le dissipateur à ailettes voisin,

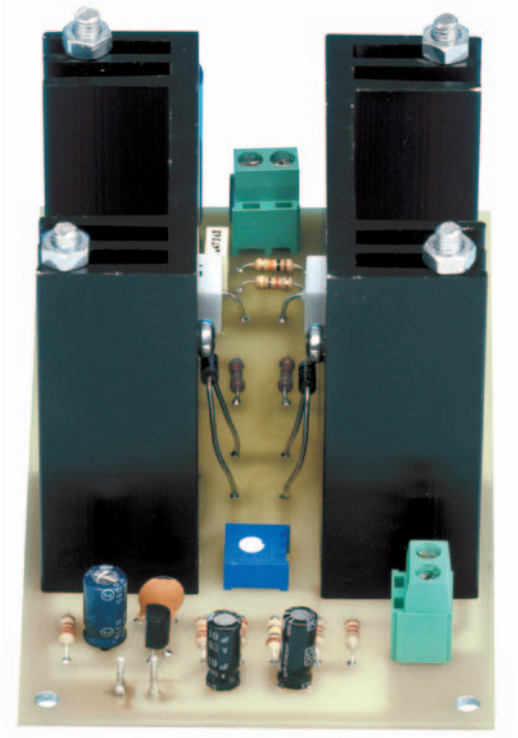


Figure 1 : Photo d'un des prototypes de l'amplificateur vu de face. Les deux dissipateurs à ailettes sont fixés chacun à une extrémité du circuit imprimé à l'aide de deux longs boulons.

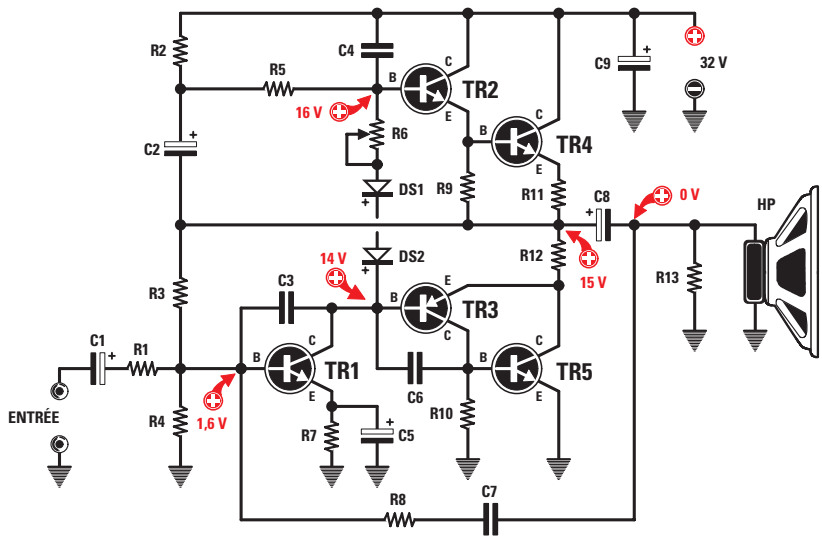


Figure 2 : Schéma électrique de l'amplificateur Hi-Fi entièrement transistorisé fournissant une puissance de 10-12 W sur une charge de 8 ohms d'impédance. Le trimmer R6 est à régler pour lui faire consommer un courant de quelque 70 mA quand aucun signal n'est appliqué à son entrée (courant de repos).

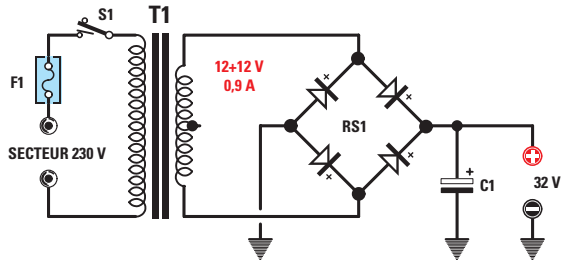


Figure 3 : Schéma électrique de l'alimentation stabilisée EN1617 à utiliser pour alimenter l'amplificateur Hi-Fi EN1616. Le secondaire du transformateur T1 fournit une tension de 12 + 12 V soit 24 V (la prise centrale de l'enroulement secondaire n'est pas utilisée). Une fois redressée par le pont RS1 et lissée par l'électrolytique C1, cette tension sera de 32 V.

Liste des composants

R1..... 2,2 k
R2..... 1,2 k
R3..... 180 k
R4..... 22 k
R5..... 4,7 k
R6..... 500 trimmer
R7..... 220
R8..... 47 k
R9..... 820
R10 ... 820
R11 ... 0,47 5 W
R12 ... 0,47 5 W
R13 ... 10 k
C1..... 10 µF électrolytique
C2..... 47 µF électrolytique
C3..... 100 pF céramique
C4..... 100 pF céramique
C5..... 100 µF électrolytique
C6..... 100 pF céramique
C7..... 470 nF polyester
C8..... 2 200 µF électrolytique
C9..... 2 200 µF électrolytique
DS1 ... 1N4007
DS2 ... 1N4007
TR1.... NPN BC547
TR2.... NPN BD139
TR3.... PNP BD140
TR4.... NPN BD241
TR5.... NPN BD241
HP..... haut-parleur
ou enceinte 8 ohms

Sauf spécification contraire, toutes les résistances sont des 1/4 W à 5 %.

Liste des composants

C1..... 3 300 µF électrolytique
RS1 ... pont 80 V 2 A
F1..... fusible 1 A
T1..... transformateur secteur
230 V 20 VA secondaire
12 + 12 V 0,9 A
(mod. T020.06)
S1..... interrupteur

comme le montrent les figures 1 et 5. Les deux résistances de puissance en céramique R11 et R12 seront de préférence maintenues (par leurs pattes) à deux ou trois mm de la surface du circuit imprimé, afin de permettre une bonne dissipation de la chaleur dégagée.

Pour l'alimentation, attention aux polarités, en particulier utilisez du fil rouge pour le positif + et du noir pour le négatif – (masse) : le + sort en bas à gauche de la platine. Le transformateur est fixé directement sur la platine par quatre petits boulons.

Le montage dans le boîtier

Nous le laissons cette fois à votre discrétion, mais remarquez qu'un boîtier métallique de type "rack 19 pouces" permet une réalisation facile et esthétique : les deux côtés du boîtier peuvent être constitués par des rails en profilé d'aluminium anodisé noir. Disposez verticalement les deux platines amplificateurs (une de chaque côté du rack) et faites en sorte que vous puissiez fixer la section haute des deux dissipateurs de chaque platine au rail constituant le côté du rack (avec les quatre boulons longs ce sera facile).

Entre les deux platines, au milieu du rack, il ne vous reste plus qu'à placer la platine alimentation, transformateur côté panneau arrière. Sur ce panneau arrière faites sortir le cordon secteur 230 V, placez le porte-fusible, montez deux (une par canal droit/gauche) ou quatre (deux par canal, pour deux paires d'enceintes) prises de sortie BF Hi-Fi et deux prises RCA "cinch" pour l'entrée stéréo.

En face avant, rien que l'interrupteur secteur S1 plus peut-être une LED jouant le rôle de voyant M/A alimenté par le 32 Vcc (avec une résistance en série). Et pourquoi pas un Vu-mètre stéréo à aiguille ou à LED, comme nous en avons déjà proposé.

Quant au(x) préamplificateur(s), vous aurez la place de le(s) mettre dans ce boîtier métallique...si vous n'êtes pas un puriste ! Si vous l'êtes, vous prendrez un rack 19 pouces de même aspect, mais plus fin –une unité rack suffira– et vous y mettrez le ou les préamplificateur(s) avec son (leur) alimentation.

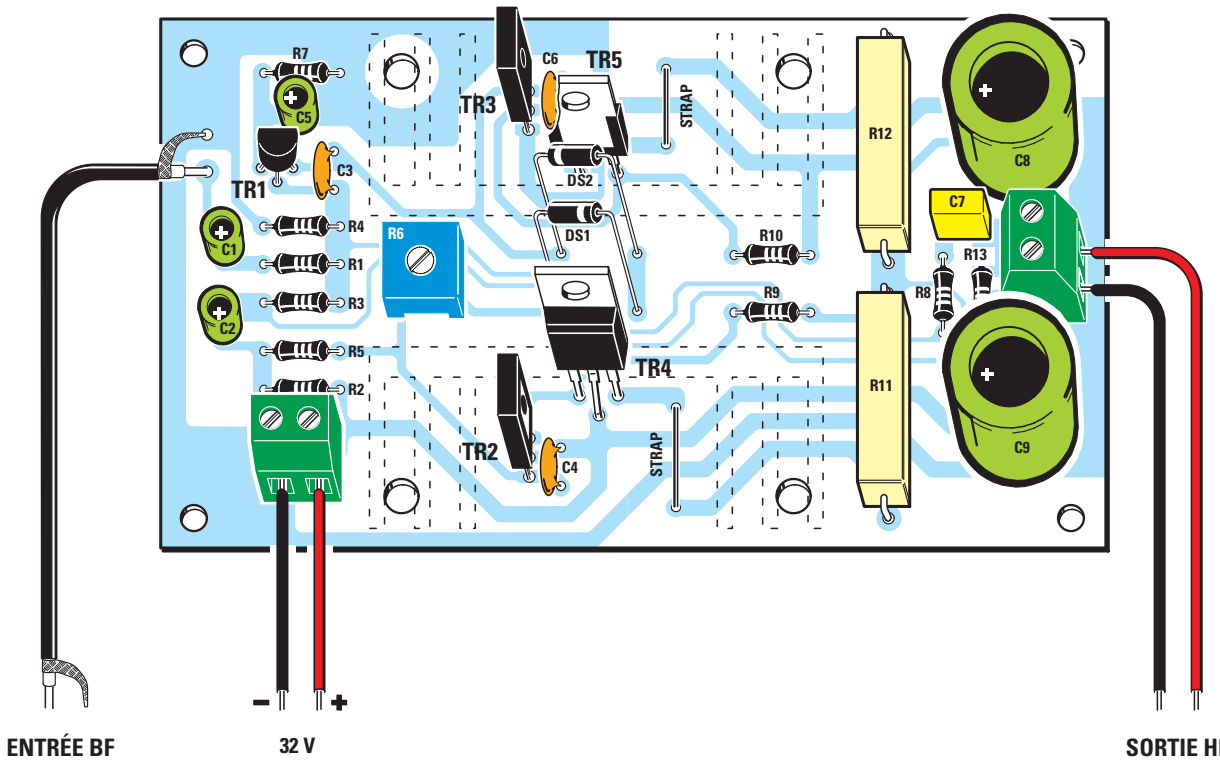


Figure 4a : Schéma d'implantation des composants de l'amplificateur EN1616. N'oubliez pas d'insérer et de souder, près de TR4-TR5, le "strap" filaire noté strap (un simple morceau de fil de cuivre dénudé). Les résistances R12-R11 peuvent avoir des dimensions différentes de celles représentées ici.

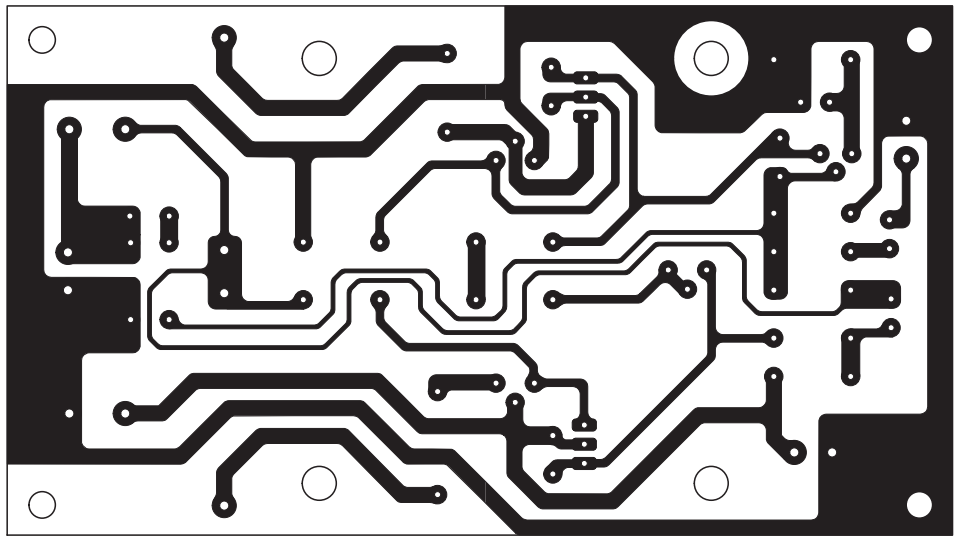


Figure 4b : Dessin, à l'échelle 1, du circuit imprimé de l'amplificateur.

Le réglage du trimmer R6 et le tour est joué

Eh bien, avant de faire fonctionner votre amplificateur, il ne vous reste qu'à régler le courant de repos de chacune des deux voies (en version stéréo) à l'aide des trimmers R6 et R6'. Pour cela, court-circuitez au préalable les entrées (avec des pinces croco de petite taille c'est facile pour des prises RCA "cinch") afin de ne capter aucun signal. Reliez en série avec le fil du positif rouge + alimentant chacune des deux voies (oui, il vous faudra faire deux mesures et deux réglages des deux trimmers) un multimètre réglé sur CC 300 mA fond d'échelle.

Alimentez l'amplificateur (une voie, puis l'autre) et tournez le curseur du premier R6 jusqu'à ce que le multimètre numérique ou analogique indique un courant de 70 mA environ (cette valeur correspond au courant de repos, soit le courant consommé par l'amplificateur, chaque voie du moins, en l'absence de tout signal en entrée). Cette valeur n'est pas critique : si vous trouvez 68 à 72 mA, c'est bon. Après avoir réglé ainsi le premier R6, coupez l'alimentation de la première voie, alimentez la seconde, procédez de même avec R6' et tâchez d'obtenir la même valeur. Enlevez le multimètre, rétablissez les connexions du positif pour les deux voies. Branchez aux sorties des deux voies une paire de

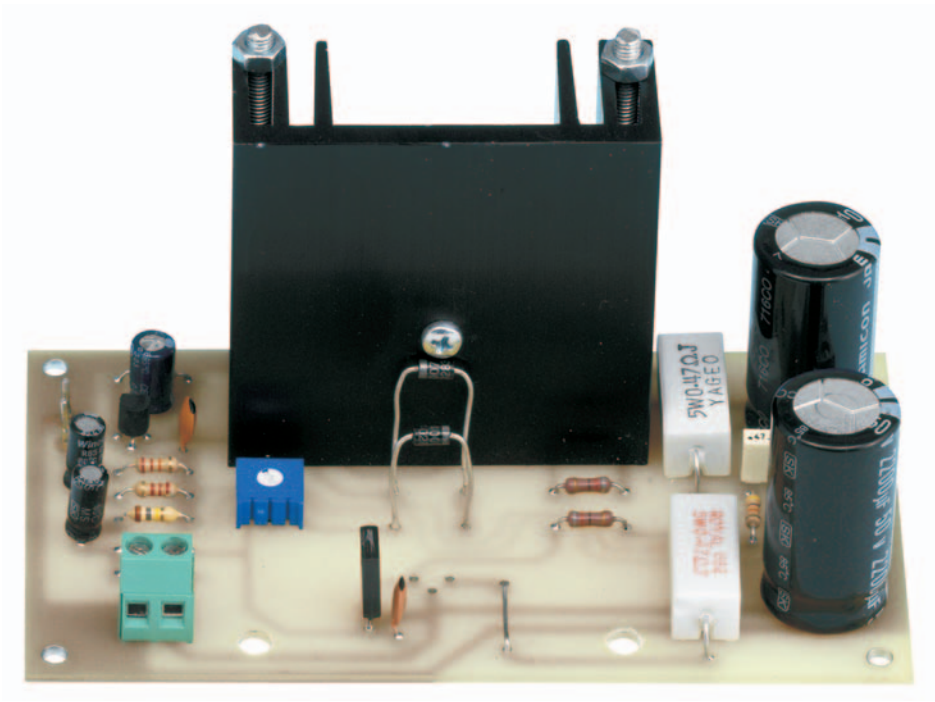


Figure 5 : Photo d'un des prototypes de la platine de l'amplificateur. Ici, pour plus de clarté, on n'a monté qu'un des deux dissipateurs à ailettes, ce qui permet de voir comment sont montées les deux diodes DS2-DS1 (leurs pattes sont laissées longues et chaque diode s'appuie sur un des dissipateurs afin de recevoir la chaleur émise). Fixez (par vis ou boulons) les deux transistors de puissance sur les deux dissipateurs à ailettes avant de les souder sur le circuit imprimé.



Figure 6 : Photo d'un des prototypes de la platine de l'alimentation EN1617. On le voit, c'est une alimentation non stabilisée simple asymétrique fournissant environ 32 Vcc pour un courant de 0,9 A (bien sûr, elle peut également être utilisée pour alimenter d'autres montages réclamant la même tension pour un courant égal ou inférieur).

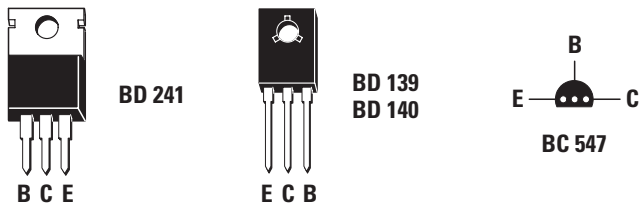


Figure 7 : Brochages des transistors finaux et des pilotes vus de face et du transistor préamplificateur vu de dessous.

haut-parleurs ou d'enceintes ayant une impédance de 8 ohms et capables de supporter la puissance de 10 WRMS et plus par voie.

En entrée, connectez une source (préamplificateur ou simplement lecteur CD) dont l'amplitude ne dépasse pas 1,6 Vpp au maximum.

Conclusion

Vous venez de réaliser vous-même (ce qui n'est pas rien) un amplificateur Hi-Fi qui vous donnera toute satisfaction si vous n'avez pas besoin d'une puissance phénoménale et, souvenez-vous, vous aviez besoin d'un amplificateur simple pour donner suite à vos expérimentations des dix préamplificateurs à deux transistors/FET : eh bien vous l'avez !

Comment construire ce montage ?

Tout le matériel nécessaire pour construire cet amplificateur Hi-Fi de 10 WRMS EN1616-1617 est disponible chez certains de nos annonceurs (pour réaliser un amplificateur stéréo, il vous faut deux platines EN1616 et une platine EN1617 et tous les composants allant avec). Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp. ◆

TABLEAU D'AFFICHAGE MINIATURE LCD AVEC RETRO-ECLAIRAGE



- tableau d'affichage miniature LCD rétro-éclairé à 1 ligne de 16 caractères
- mémoire EEPROM pour 124 caractères
- message glissant à vitesse réglable
- éditeur de message incorporé
- interface RS232 incorporé pour facilement charger et afficher le message
- logiciel d'interface gratuit disponible sur www.velleman.be
- applications: lors d'expositions, pour faire de la publicité, pour accueillir des visiteurs à la porte, comme badge, comme écran sériel dans une installation...
- alimentation: 9VCC / 100mA
- dimensions: 115 x 43 x 75mm

€29,95
MK158

HORLOGE A SIMPLE DIGIT JUMBO



- horloge gadget
- l'heure est affichée par une séquence de chiffres
- digit jumbo 57mm rouge clair à 7 segments
- boîtier inclus
- opération par une seule touche
- fonction écran blanc
- système 12h
- précision: 30ppm
- alimentation: 9-12Vcc/150mA max.
- dimensions: 74 x 59 x 32mm

€16,95
MK153



JOURNAL DEFILANT MINIATURE A LCD

€24,95
MK157

- journal défilant miniature avec afficheur LCD 16 caractères 1 ligne
- déroulement de gauche à droite avec vitesse réglable
- applications: expositions, publicité, accueil de visiteurs, badge, ...
- mémoire pour message de max. 46 caractères
- possibilité d'insérer des pauses pendant le déroulement
- alimentation: 2 x piles LR3 (AAA)
- dimensions: 100 x 38 x 32mm



HORLOGE LED NUMERIQUE



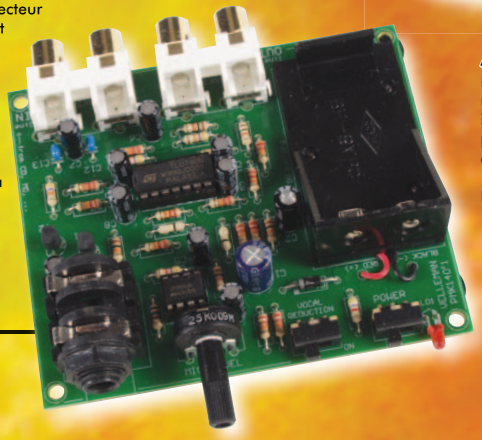
- horloge avec afficheurs 7 segments dans un boîtier attractif
- affichage de l'heure en mode 12h ou 24h
- avec piles de secours (3 x LR6) en cas d'une panne d'alimentation
- affichage de l'heure même lors d'une panne d'alimentation
- alimentation: 9V CA/CC
- dimensions: 92 x 45 x 101mm
- adaptateur recommandé: PS905AC (230VCA)

€21,95
MK151

KARAOKE

- chantez sur vos disques préférés
- connectez l'appareil à votre lecteur CD, lecteur MD ou une autre source de niveau ligne et votre amplificateur
- entrée RCA (cinch) et sortie RCA
- pourvu d'une connexion microphone de 6.35mm, convient pour la plupart des microphones
- niveau microphone réglable
- commutateur "atténuation de la voix" réduit le volume de la partie chantée de la plupart des enregistrements stéréo
- alimentation: accu de 9V
- basse consommation de courant: 16mA typ.
- dimensions: 100 x 84 x 25mm

€14,95
MK140



JEU ELECTRONIQUE

Répétez la série qui devient de plus en plus longue et rapide. 4 niveaux de jeux. Son et/ou indication LED instaurable. Consommation faible. Contrôlée par microprocesseur. •désactivation automatique

- alimentation: 3 piles de 1.5V LR3 (non incl.)
- dimensions: 84 x 67 x 41mm



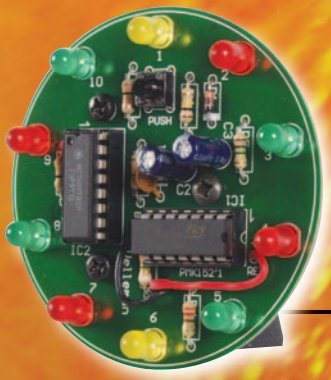
€14,95
MK159



VUMETRE EN FORMAT DE POCHE

- LEDs clignotent au rythme de la musique
- surprenez vos amis dans la discothèque
- microphone incorporé
- sensibilité réglable
- alimentation: 3V, type de pile CR2032 (non incl.)
- interrupteur on/off
- 5 LEDs
- dimensions: 87 x 50 x 25mm

€9,95
MK146



ROUE DE LA FORTUNE

- Roue de la Fortune avec 10 LEDs
- pressez le bouton pour faire tourner la roue
- vitesse de rotation diminue graduellement
- une LED reste allumée quand la roue s'arrête
- désactivation automatique
- basse consommation
- alimentation: 1 pile 9V (non incl.)
- dimensions: Ø64 x 33mm

€6,95
MK152



velleman® HIGH-Q kit

Consultez notre site Internet
http://www.velleman.fr

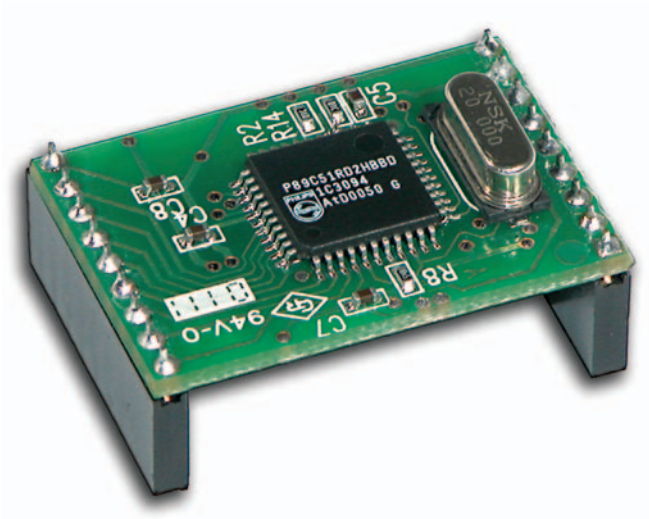
Demandez notre *catalogue KIT* chez votre distributeur VELLEMAN

velleman® électronique

8, Rue du Maréchal de Lattre de Tassigny, 59000 LILLE ☎03 20 15 86 15 03 20 15 86 23

Comment programmer le module SitePlayer SP1

Quatrième partie



Dans cette série d'articles, nous vous apprenons à programmer et à utiliser le module SitePlayer SP1. Ce circuit intégré réalise un véritable serveur pour la Toile ("Web Server"), c'est-à-dire qu'il permet d'interagir avec n'importe quel dispositif électronique à travers une page Internet normale. Nous apprenons à nous servir de ce module pour réaliser des applications nous permettant de faire communiquer sur le réseau des appareils distants en tout genre.

Après avoir vu, dans les parties précédentes, les principes fondamentaux de réalisation d'une application SitePlayer, la structure d'un programme et l'environnement de développement (SiteLinker) permettant de transférer des images et des pages Web dans la mémoire du module SitePlayer, nous commençons cette fois à analyser les programmes que nous avons réalisés pour la platine d'expérimentation ("demoboard"), de façon à acquérir l'indispensable familiarité avec les techniques de programmation.

Tous les programmes des "demoboards" devront avoir un répertoire propre. Pour nous ici, tous sont des sous répertoires que l'on atteint avec le parcours **C:\Programmes\SitePlayer\demoboard**. A l'intérieur de ce répertoire, pour chaque programme, nous avons prévu un autre sous répertoire, par exemple pour la demo1 le sous répertoire est **C:\Programmes\SitePlayer\demoboard\demo1**. Seront placés dans ce répertoire aussi bien le programme en Basic pour la gestion du PIC (extension .BAS pour le fichier source et extension .HEX pour le fichier compilé) que le fichier de définition avec extension .SPD. Dans le même répertoire se trouve ensuite un sous répertoire, que nous appelons "root" (par exemple pour demo1 **C:\Programmes\SitePlayer\demoboard\demo1\root)** dans lequel se trouvent tous les fichiers des pages html. Il y aura donc au moins un fichier index.htm, les éventuelles autres pages html et toutes les images, typiquement au format jpeg, dont nous aurons besoin. Bien sûr, les programmes pourront être situés dans n'importe quel répertoire, en prenant toutefois la précaution de définir exactement les parcours dans le fichier de définition, comme nous venons de l'expliquer.

Pour chaque programme nous devons donc créer au moins trois fichiers. Les fichiers html peuvent être utilisés exactement comme nous les avons réalisés ou alors modifiés à volonté (si vous connaissez le langage html). Par la suite, au lieu d'expliquer en détail les pages que nous avons réalisées, nous mettrons en évidence les points clés permettant de réaliser des pages Web avec insertion d'objets issus du module SitePlayer. Les fichiers de définition peuvent être modifiés eux aussi, par exemple pour paramétrer une adresse IP différente de celle que nous avons fixée et qui pourrait ne pas être compatible avec un réseau particulier. Les programmes de gestion des PIC sont écrits en PicBasic. Ces programmes pourront être écrits aussi en Assembleur (à la place du C) et, dans ce cas aussi, nous avons cherché à mettre en évidence les fonctions que devrait remplir un programme plutôt que de faire une liste des instructions utilisées.

Programme Demo1

Avec cette démo nous voulons expliquer comment il est possible de faire passer des paramètres du micro au module SitePlayer et de ce dernier à une page Web, de telle façon qu'à chaque demande d'envoi de la part du navigateur, la page html envoyée comme réponse par le module contienne les don-

nées mises à jour. En particulier, l'état des huit micro-interrupteurs du dip-switch de la platine d'expérimentation est visualisé dans la page Web. Les fichiers dont nous avons besoin pour créer cette application sont avant tout le fichier index.htm contenant la page html de notre application, les images nécessaires pour la visualisation correcte de la page, le fichier demo1.bas, écrit en PICbasic, à utiliser pour la programmation du micro et le fichier demo1.spd de définition à utiliser pour programmer le module SitePlayer. C'est ce fichier que nous allons maintenant analyser.

Fichier index.htm

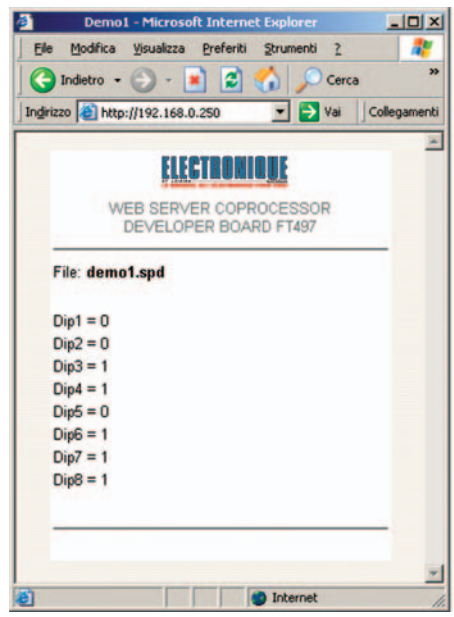
C'est le fichier de la page html (voir figure 3, Demo1: "listing" html). Pour voir comment est la page Web, il suffit d'ouvrir ce fichier avec un navigateur Internet, comme IE. Pour visualiser le code html de cette page, il suffit d'ouvrir ce fichier avec un éditeur de texte, comme NotePad ou BlocNote. A première vue ce fichier peut sembler assez complexe. Étant donné que nous n'avons pas pour but de vous faire un cours sur le langage HTML, voyons seulement les grandes lignes afin de comprendre la signification de la structure du fichier (nous verrons par contre en détail les paramètres insérés à partir de SitePlayer).

Rappelons qu'en HTML on utilise des tags pour définir un objet particulier. Les tags sont toujours écrits entre les symboles **< et >** et précédés par /. Par exemple, tout le code html doit être contenu entre le tag **<html>** (placé au début du fichier) et le tag **</html>**, à la fin du fichier. Le tag **<TITLE>** définit le titre de la page Web, le tag **<P>** renferme un paragraphe, le tag **<CENTRE>** visualise ce que contient le centre de la page et ainsi de suite. Les tags **<TR>** et **<TD>** définissent respectivement une ligne et une colonne dans un tableau. Si nous regardons la page Web telle qu'elle est visualisée par un navigateur et si nous parcourons le "listing", nous devrions pouvoir comprendre comment les différents tags donnent lieu aux diverses parties de la page. La partie qui nous intéresse le plus est la suivante:

```
<p>
<font size="2"
face="Arial, Helvetica, sans-
serif">
Dip1 = ^Dipswitch'7
</font>
</p>
```

Figure 1: Démo 1.

Exemple de gestion des entrées numériques: l'état du dip-switch DS1 est lu puis visualisé 1 ou 0 en fonction de la position du micro-interrupteur intéressé.



Cette ligne, encadrée par les tags **<P>** et **</P>**, est visualisée avec la mesure des caractères identifiée par font size et avec le type de caractères identifié par face. La partie qui nous intéresse vraiment est donc la suivante:

```
Dip1 = ^Dipswitch'7
```

Dip1 est seulement du texte (et il sera visualisé comme tel), alors que ^Dipswitch est un objet du SitePlayer. On a vu précédemment que les objets sont le moyen qu'utilise le SitePlayer pour communiquer avec le monde extérieur: ce sont essentiellement des cellules de mémoire dont le contenu peut d'un côté être modifié par le micro relié au port série et de l'autre constituer des valeurs à insérer dans les pages Web quand elles sont envoyées à un navigateur. Si l'on visualise cette page Web en ouvrant le fichier sur l'ordinateur avec un navigateur, on obtient simplement la mention:

```
Dip1 = ^Dipswitch'7
```

Quand le navigateur interroge le SitePlayer, ^Dipswitch est alors remplacé par la valeur de l'objet correspondant, soit la valeur de la cellule de mémoire RAM que nous avons définie comme objet Dipswitch. Étant donné que Dipswitch, nous allons le voir bientôt, est un octet qui reflète exactement l'état du dip-switch sur la platine d'expérimentation, pour visualiser l'état de chaque micro-interrupteur, il est nécessaire d'isoler chaque bit de cet octet. Pour ce faire, nous avons utilisé un modifieur; en effet, le modifieur ou mot clé:

```
^oggetto'n
```

Figure 2: Démo 2 (définitions).

```
;DEMO1.SPD

;DEFINITIONS
$Devicename "Électronique Loisir Magazine Demo1"
$DHCP off
$DownloadPassword ""
$SitePassword ""
$InitialIP "192.168.0.250"
$PostIRQ on
$Sitefile
"C:\Programmes\SitePlayer\demoboard\demo1\demo1.spb"
$Sitepath
"C:\Programmes\SitePlayer\demoboard\demo1\root"

;OBJECTS
org             05h
Dipswitch               db 0
```

Figure 3 : Démo 1 (“listing” HTML).

```
<html>
<head>
<title>Electronique Loisir Magazine Demo1</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" background="sfondo1.gif">
<table width="300" border="0" align="center" bordercolor="#FFFFFF" bgcolor="#FFFFFF">
  <tr>
    <td><div align="center"><img src="Electronique Loisir Magazine1.jpg" width="100" height="37"></div></td>
  </tr>
  <tr>
    <td><div align="center"><font color="#999999" size="2" face="Arial, Helvetica, sans-serif">WEB SERVER COPROCESSOR <br>
        DEVELOPER BOARD FT497</font></div></td>
  </tr>
  <tr>
    <td><hr noshade></td>
  </tr>
  <tr>
    <td><div align="left"><font size="2" face="Arial, Helvetica, sans-serif">File:
        <strong>demo1.spd</strong></font></div></td>
  </tr>
  <tr>
    <td> </td>
  </tr>
  <tr>
    <td>
        <p><font size="2" face="Arial, Helvetica, sans-serif">Dip1 = ^Dipswitch'7</font></p>
      </td>
  </tr>
  <tr>
    <td>
<p><font size="2" face="Arial, Helvetica, sans-serif">Dip2 = ^Dipswitch'6</font></p>
      </td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip3 = ^Dipswitch'5</font></td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip4 = ^Dipswitch'4</font></td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip5 = ^Dipswitch'3</font></td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip6 = ^Dipswitch'2</font></td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip7 = ^Dipswitch'1</font></td>
  </tr>
  <tr>
    <td><font size="2" face="Arial, Helvetica, sans-serif">Dip8 = ^Dipswitch'0</font></td>
  </tr>
  <tr>
    <td> </td>
  </tr>
  <tr>
    <td><hr noshade></td>
  </tr>
  <tr>
    <td> <div align="center"> <font color="#999999" size="1" face="Arial, Helvetica, sans-serif">&copy;
        2003 Électronique Loisir Magazine. All rights reserved.</font><br>
      </div></td>
  </tr>
</table>
</body>
</html>
```

Figure 4 : Démo 1(“listing” Basic).

```
;File DEMO1.BAS

TEMP            var     byte
DATOIN          var     byte
DATOOUT         var     byte
INDIRIZZO       var     byte
FLAGOK          var     bit
TEMPERATURA             var     byte
TRIMMER         var     byte

Include “modedefs.bas”

‘LED
SYMBOL  LED1 = PORTC.0
SYMBOL  LED2 = PORTC.1
SYMBOL  LED3 = PORTC.2
SYMBOL  LED4 = PORTC.3
SYMBOL  LED5 = PORTC.4
SYMBOL  LED6 = PORTC.5
SYMBOL  LED7 = PORTC.6
SYMBOL  LED8 = PORTC.7

‘DIP
SYMBOL  DIP1 = PORTB.0
SYMBOL  DIP2 = PORTB.1
SYMBOL  DIP3 = PORTB.2
SYMBOL  DIP4 = PORTB.3
SYMBOL  DIP5 = PORTB.4
SYMBOL  DIP6 = PORTB.5
SYMBOL  DIP7 = PORTB.6
SYMBOL  DIP8 = PORTB.7

‘ Communication avec le Site Player
SYMBOL  TX232 = PORTA.2
SYMBOL  RX232 = PORTA.5

‘Interrupt du Site Player
SYMBOL  INTSP = PORTA.4

‘Définition I/O

        ADCON1=%0000010 ‘RA0,RA1 e RA3 analogiques
        ADCON0=%10000001
        OUTPUT  LED1
        OUTPUT  LED2
        OUTPUT  LED3
        OUTPUT  LED4
        OUTPUT  LED5
        OUTPUT  LED6
        OUTPUT  LED7
        OUTPUT  LED8
        OUTPUT  TX232
        INPUT   RX232

‘Dip

        INPUT  DIP1
        INPUT  DIP2
        INPUT  DIP3
        INPUT  DIP4
        INPUT  DIP5
        INPUT  DIP6
        INPUT  DIP7
        INPUT  DIP8

‘Portb avec les pull up internes

        OPTION_REG.7 = 0
        INPUT   Porta.0
        INPUT   Porta.1
        INPUT   INTSP
        Pause 100
        Gosub TEST
        Pause 300

START:
‘Initialise
        Gosub SEND20
        Pause 200
START1:
‘Lit portb
        DATOOUT = Portb
        INDIRIZZO = 5
        Gosub WRITEDATO
        Pause 1000
        Goto    START1

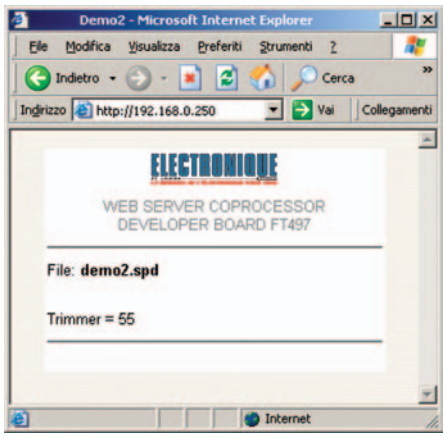
‘*****************************************
‘Routine d’envoi de 20 octets 0 pour initialisation
SEND20:
        For TEMP = 1 to 20
            Serout TX232,T9600,[0]
            Pause 5
        Next TEMP
        Return

‘Routine d’écriture d’un octet
‘Envoi : Commande de write (128),
‘INDIRIZZO, DATOOUT
WRITEDATO:
        Pause 10
        Serout TX232,T9600,[128]
        Pause 5
        Serout TX232,T9600,[INDIRIZZO]
        Pause 5
        Serout TX232,T9600,[DATOOUT]
        Pause 20
        Return

‘Routine de vérification des LED
TEST:
    Gosub SPEGNILED
    Led1 = 1
    Pause 200
    Led1 = 0
    Led2 = 1
    Pause 200
    Led2 = 0
    Led3 = 1
    Pause 200
    Led3 = 0
    Led4 = 1
    Pause 200
    Led4 = 0
    Led5 = 1
    Pause 200
    Led5 = 0
    Led6 = 1
    Pause 200
    Led6 = 0
    Led7 = 1
    Pause 200
    Led7 = 0
    Led8 = 1
    Pause 200
    Gosub SPEGNILED
    Return

‘Routine d’extinction des LED
SPEGNILED:
        LED1 = 0
        LED2 = 0
        LED3 = 0
        LED4 = 0
        LED5 = 0
        LED6 = 0
        LED7 = 0
        LED8 = 0
        Return
```

Figure 5: Démo 2.



Exemple de lecture d'une entrée analogique: la valeur du trimmer R4 est lue puis visualisée au format numérique.

restitue la valeur du n-ième bit d'objet. Dans notre cas, donc, le SitePlayer remplacera ^Dipswitch'7 par la valeur du septième bit de l'objet Dipswitch. Ce bit correspond exactement à l'état logique du micro-interrupteur correspond sur la platine d'expérimentation.

Fichier demo1.spd

Comme nous l'avons vu dans les parties précédentes, il est nécessaire de créer un fichier avec extension .SPD lequel, une fois compilé par le SiteLinker, sera chargé dans le module SitePlayer. Ce fichier (voir figure 2 Demo1: définitions) contient une série de définitions et les objets. Comme définitions nous avons:

$Devicename "Electronique Loisir Magazine" qui donne un nom au dispositif,
$DHCP off qui déshabilite le mode d'attribution de l'IP de la part du serveur (le SitePlayer a un adresse IP statique),
$DownloadPassword ""** et **$SitePassword "" qui en fait ne donnent aucun mot de passe,
;$InitialIP "192.168.0.250" qui attribue l'adresse IP au module SitePlayer,
$PostIRQ on qui permet d'activer sur la broche 11 du SitePlayer la signalisation d'arrivée des données,
$Sitefile "C:\Programmes\SitePlayer\demoboard\demo1\demo1.spb" qui définit exactement la localisation et le nom du fichier spb qui sera créé (le fichier spb est celui que l'on chargera dans le module SitePlayer),
$Sitepath "C:\Programmes\SitePlayer\demoboard\demo1\root" qui définit le parcours permettant de trouver les pages Web et les images correspondantes.

Pour personnaliser cette partie des définitions, il faut prêter attention aux deux définitions $Sitefile et $Sitepath, dans lesquelles il faut clairement indiquer les parcours des fichiers. Autre paramètre particulièrement important: $InizialIP car il attribue l'adresse IP au dispositif. En ce qui concerne les objets, cette première demo n'en présente qu'un que nous avons identifié par l'adresse 5 (c'est ce qu'indique la directive org 05h) et qui est un octet avec valeur par défaut égale à 0 (db 0).

Figure 6: Démo 2 (Définitions).

```
;DEMO2.SPD

;DEFINITIONS
$Devicename "Électronique Loisir Magazine Demo2"
$DHCP off
$DownloadPassword ""
$SitePassword ""
$InitialIP "192.168.0.250"
$PostIRQ on
$Sitefile C:\Programmes\SitePlayer\demoboard\demo2\demo2.spb"
$Sitepath
"C:\Programmes\SitePlayer\demoboard\demo2\root"

;OBJECTS
org             05h
Trimmer         db 0
```

Fichier demo1.bas

Après avoir analysé en détail le fichier de la page Web et du SitePlayer, il nous nous reste qu'à analyser et comprendre le programme écrit pour le microcontrôleur (en Basic) et dont le "listing" est visible figure 4, Demo1: "Listing" Basic. La première partie, jusqu'à l'étiquette START, est pratiquement la même pour tous les programmes que nous avons réalisés. Il s'agit de la définition des variables utilisées dans le programme, de la définition des diverses lignes d'E / S et des lignes de communication avec le SitePlayer. Le convertisseur A / N, qui dans cette application n'est pas utilisé, est initialisé. Une routine de test (TEST) est appelée avant le lancement du programme proprement dit, afin de vérifier le bon fonctionnement des LED. Le programme à proprement parler commence avec l'étiquette START. La routine SEND20, qui envoie au module SitePlayer 20 octets, tous à 0, est appelée. Cette procédure est nécessaire afin que nous soyons bien sûrs que le module SitePlayer interprète correctement les commandes reçues par le port série (et nous insèrerons cette routine dans tous nos programmes). A propos de la communication sérielle, rappelons que le module SitePlayer utilise une vitesse de communication normale de 9 600 bauds.

L'instruction SEROUT du PicBasic nous simplifie beaucoup l'écriture du programme. Il suffit en effet de spécifier dans cette instruction sur quelle broche envoyer les données, la vitesse de transmission et la donnée à transmettre (et automatiquement nous avons déjà réalisé la communication sérielle). Dans notre cas l'instruction:

```
Serout TX232,T9600,[0]
```

dit d'envoyer l'octet 0 sur la broche TX232 à 9 600 bauds. Une fois le module initialisé, avec l'étiquette START1, le programme principal commence, lequel n'est autre qu'une boucle qui chaque seconde lit l'état des micro-interrupteurs et réclame le sous programme WRITEDATO. Ce sous programme envoie trois octets au SitePlayer, ce sont: la commande d'écriture (128), l'adresse de la cellule de mémoire que l'on veut écrire (qui dans le sous programme est représentée par la variable INDIRIZZO), ainsi que

Figure 7 : Démo 2(“listing” HTML).

```
<html>
<head>
<title>Electronique Loisir Magazine Demo2</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<meta http-equiv="refresh" content="10">
</head>
<body bgcolor="#FFFFFF" background="sfondo1.gif">
<table width="300" border="0" align="center" bordercolor="#FFFFFF" bgcolor="#FFFFFF">
  <tr>
   <td><div align="center"><img src="Electronique Loisir Magazine1.jpg" width="100" height="37"></div></td>
  </tr>
  <tr>
   <td><div align="center"><font color="#999999" size="2" face="Arial, Helvetica, sans-serif">WEB
   SERVER COPROCESSOR <br>
   DEVELOPER BOARD FT497</font></div></td>
  </tr>
  <tr>
   <td><hr noshade></td>
  </tr>
  <tr>
  <td><div align="left"><font size="2" face="Arial, Helvetica, sans-serif">File:
     <strong>demo2.spd</strong></font></div></td>
  </tr>
  <tr>
   <td> </td>
  </tr>
  <tr>
   <td> <p><font size="2" face="Arial, Helvetica, sans-serif">Trimmer = ^Trimmer</font></p></td>
  </tr>
  <tr>
   <td><hr noshade></td>
  </tr>
  <tr>
  <td> <div align="center"> <font color="#999999" size="1" face="Arial, Helvetica, sans-serif">&copy;
  2003 Électronique Loisir Magazine. All rights reserved.</font><br>
  </div></td>
  </tr>
</table>
</body>
</html>
```

la donnée à écrire. On avait attribué à la variable INDIRIZZO la valeur 5 car 5 est l’adresse de mémoire de l’objet Dipswitch définit dans le fichier de définitions du SitePlayer.

L’octet de la donnée à écrire est composé de la variable DATOOUT ; à cette variable a été attribuée la valeur du Port B, soit l’état du micro-interrupteur. Ici nous devons apporter une précision à propos de la commande d’écriture. Cette dernière est en effet établie grâce à la formule :

commande = 128 + nombre d’octets à envoyer –1.

En fait, comme dans notre cas un seul octet est envoyé, on a la commande = 128 + 1 –1, soit 128. Avec cette commande on peut envoyer jusqu’à 16 octets, mémorisés à partir de l’adresse spécifiée.

Programme Demo2

Avec ce programme le réglage du trimmer de la platine d’expérimentation est visualisé sur la page Web. Si nous regardons le code html de la figure 7, nous remarquons tout de suite la présence d’un objet dans la ligne :

```
Trimmer = ^Trimmer.
```

Cet objet se retrouve bien sûr dans le fichier de définition demo2.spd, en tout point semblable au fichier de définition de demo1, sauf que le nom de l’objet change. Dans la description nous approfondirons davantage le programme résident du PIC. La partie initialisation est semblable à celle de demo1, tout comme la boucle principale du programme, à la différence qu’au lieu de lire le port B, auquel était relié le dipswitch, c’est maintenant le canal AN0 du convertisseur A/N, auquel est relié le trimmer, qui est lu. C’est la routine CONVERTI qui procède à la conversion analogique-numérique.

Ce programme, déjà prévu pour lire la température de la sonde LM35, présente sur la platine d’expérimentation, effectue les opérations suivantes. D’abord le canal AN0, auquel le trimmer est relié, est sélectionné. Puis la conversion démarre et après cinq ms le registre ADRESH, dans lequel est contenu le résultat de la conversion, est lu. Cette valeur est mise ensuite dans la variable TRIMMER. Puis le canal AN1, auquel est reliée la sonde de température, est sélectionné, à nouveau la conversion est lancée et après cinq ms le résultat est lu pour être attribué à la TEMPERATURA. Dans le cas de la demo1, seule la valeur de la variable TRIMMER est utilisée pour être envoyée, à travers la routine WRITEDATO, au module SitePlayer.

Figure 8 : Démo 2 (“listing” Basic).

```
;File DEMO2.BAS

TEMP                    var     byte
DATOIN                  var     byte
DATOOUT                 var     byte
INDIRIZZO               var     byte
FLAGOK                  var     bit
TEMPERATURA             var     byte
TRIMMER                 var     byte

Include "modedefs.bas"

'LED
SYMBOL  LED1 = PORTC.0
SYMBOL  LED2 = PORTC.1
SYMBOL  LED3 = PORTC.2
SYMBOL  LED4 = PORTC.3
SYMBOL  LED5 = PORTC.4
SYMBOL  LED6 = PORTC.5
SYMBOL  LED7 = PORTC.6
SYMBOL  LED8 = PORTC.7

'DIP
SYMBOL  DIP1 = PORTB.0
SYMBOL  DIP2 = PORTB.1
SYMBOL  DIP3 = PORTB.2
SYMBOL  DIP4 = PORTB.3
SYMBOL  DIP5 = PORTB.4
SYMBOL  DIP6 = PORTB.5
SYMBOL  DIP7 = PORTB.6
SYMBOL  DIP8 = PORTB.7

' Communication avec le Site Player
SYMBOL  TX232 = PORTA.2
SYMBOL  RX232 = PORTA.5

'Interrupt du Site Player
SYMBOL  INTSP = PORTA.4

'Définition I/O
        ADCON1=%00000100
        ADCON0=%10000001
        OUTPUT  LED1
        OUTPUT  LED2
        OUTPUT  LED3
        OUTPUT  LED4
        OUTPUT  LED5
        OUTPUT  LED6
        OUTPUT  LED7
        OUTPUT  LED8
        OUTPUT  TX232
        INPUT   RX232

'Dip
        INPUT  DIP1
        INPUT  DIP2
        INPUT  DIP3
        INPUT  DIP4
        INPUT  DIP5
        INPUT  DIP6
        INPUT  DIP7
        INPUT  DIP8

'Portb avec les pull up internes

        OPTION_REG.7 = 0
        INPUT   Porta.0
        INPUT   Porta.1
        INPUT   INTSP

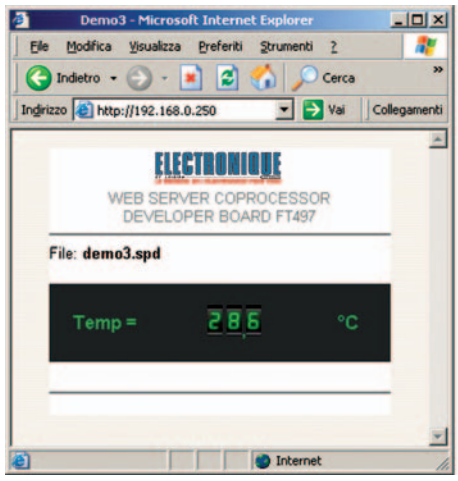
START:
'Initialise
        Gosub SEND20
        Pause 500
START1:
'Lit entrées analogiques
        Gosub CONVERTI
        DATOOUT = TRIMMER
        INDIRIZZO = 5
        Gosub WRITEDATO
        Pause 1000
        Goto    START1

'**********************************
'Routine d'envoi de 20 octets 0 pour initialisation
SEND20:
        For TEMP = 1 to 20
            Serout TX232,T9600,[0]
            Pause 5
        Next TEMP
        Return

'Routine d'écriture d'un octet
'Envoi : Commande de write (128), INDIRIZZO, DATOOUT
WRITEDATO:
        Pause 10
        Serout TX232,T9600,[128]
        Pause 5
        Serout TX232,T9600,[INDIRIZZO]
        Pause 5
        Serout TX232,T9600,[DATOOUT]
        Pause 20
        Return

'Routine de conversion
'Met le résultat en TRIMMER et TEMPERATURA
CONVERTI:
    ADCON0.3 = 0
    ADCON0.4 = 0
    ADCON0.5 = 0
    Pause 1
    ADCON0.2 = 1   'Fait partir la conversion
    Pause 5
    TRIMMER = ADRESH
    Pause 1
    ADCON0.3 = 1
    ADCON0.4 = 0
    ADCON0.5 = 0
    Pause 1
    ADCON0.2 = 1   'Fait partir la conversion
    Pause 5
    TEMPERATURA = ADRESH
    Pause 1
    Return
```

Figure 5 : Démo 2.



Exemple de lecture d'une entrée analogique et visualisation graphique : la température ambiante est lue au moyen du capteur SENS puis visualisée au format graphique.

Figure 10 : Démo 3 (Définitions).

```
;DEMO3.SPD

;DEFINITIONS
$Devicename "Électronique Loisir Magazine Demo3"
$DHCP off
$DownloadPassword ""
$SitePassword ""
$InitialIP "192.168.0.250"
$PostIRQ on
$Sitefile
"C:\Programmes\SitePlayer\demoboard\demo3\demo3.spb"
$Sitepath
"C:\Programmes\SitePlayer\demoboard\demo3\root"

;OBJECTS
org             05h
Cifra3          db 0
Cifra2          db 0
Cifra1          db 0
```

Programme Demo3

Avec cette démo nous entendons visualiser sur la page Web la température de la sonde LM35 présente sur la platine d'expérimentation. Comme nous voulons une visualisation graphiquement intéressante, nous avons réalisé avec les chiffres de 0 à 9, en plus du point décimal, un programme de graphe et nous en avons tiré dix images jpeg, que nous avons nommées 0_blk.jpg, 1_blk.jpg, 2_blk.jpg et ainsi de suite. Avec cette platine d'expérimentation nous voulons vous montrer comment on peut visualiser une image en fonction d'un objet. Dans le cas présent, la valeur de température passe du micro

Perdre son temps ? C'est ne pas être à
HAMEXPO !

27 ème Salon International Radioamateur
Techniques de Radiocommunication et d'Informatique
22-23 octobre 2005 AUXERRE - Entrée 8 €
Participez ! C'est déjà communiquer !

- 4000 m2 d'exposants professionnels
- 1000 m2 pour le marché de l'occasion
- Conférences et démonstrations diverses
- Validation des QSL pour le DXCC
- Contrôle de vos émetteurs-récepteurs
- Stands associatifs
- Station F5REF
- **Visite du chantier médiéval de Guédelon pour les XYL** (samedi après-midi 22 octobre 2005) *(construction d'un château fort dans le respect des techniques du XIII ème siècle)*
- Entrée gratuite pour les femmes et les enfants
- Accès : AUXERREXPO rue des Plaines de l'Yonne AUXERRE 89000

La ville d'Auxerre est située au nord de la Bourgogne, aux portes du bassin parisien. Elle est facilement accessible par tous les moyens traditionnels de communication.

- Par la route : Auxerre est implantée directement sur l'autoroute A6 et la route nationale RN6, deux des principaux axes routiers nord-sud de la France. La ville est distante de Paris et Dijon de seulement 1 heure 30, et de 3 heures de Lyon. La RN 77 relie facilement Auxerre au nord-est de la France, à la Belgique, au Luxembourg et à l'Allemagne.
- Par le train : sur la ligne Paris-Lyon descendre à Laroche-Migennes, puis prendre la correspondance vers Auxerre, en train toujours, ou en taxi. Renseignements à la gare d'Auxerre-St-Gervais : 08 36 35 35 35, ou 03 86 46 28 50.
- L'aérodrome d'Auxerre-Brabches est situé à une dizaine de minutes seulement d'Auxerre. Renseignements : 03 86 48 31 89.
- Si vous arrivez par la RN6, en provenance du nord, contournez la ville par la voie express, puis suivez la signalétique AUXERREXPO. En provenance du sud, AUXERREXPO vous accueille dès votre arrivée aux portes de la ville.
- Si vous arrivez par l'autoroute A6, prenez la sortie (intitulée « Tonnerre-Chablis » en venant de Paris ou « AUXERRE » en venant de Dijon-Lyon) et prenez immédiatement la RN65 en direction d'Auxerre. En rejoignant la RN6, suivez la signalétique AUXERREXPO.

RÉSERVATION
Visite du chantier médiéval de Guédelon
Nom : ..... Prénom : .....
Adresse : .....
Code postal : ..... Ville : .....
Téléphone : .....
Réserve ..... places à 21 €
Soit un total de ..... €

RÉSERVATION D'UN EMPLACEMENT
Marché de l'occasion
Nom : ..... Prénom : .....
Indicatif : ..... Téléphone : .....
N° carte d'identité (joindre photocopie) : .....
Adresse : .....
Code postal : ..... Ville : .....
- Samedi 22 octobre ..... 40 €
- Dimanche 23 octobre ..... 20 €
- Samedi 22 et dimanche 23 octobre ..... 45 €

Branchement électrique 25 €

PCB-POOL®
Notre service repond a tous vos besoins de prototype

- Des prototypes a un prix plus bas
- Inclusive de frais d'outillage
- Tous contours poosibles
- Fr4 1.6mm, 35µm Cu
- Une qualite industrielle
- Nouvelle commande SERIES XXS
- Conseil CAO/FAO

Exemple de prix
1 EUROCARTE (double face/ mpl)
+ Outillage
+ Phototraceurs
+ TVA
= €49

GRATUIT
Un cadeau avec votre premiere commande
Votre cadeau peut etre legerement different de celu ci

Beta LAYOUT
Tel.: +353 (0)61 701170
Fax: +353 (0)61 701165
E-Mail: sales@beta-layout.com

Téléphone gratuit!
0800-903-330

Envoyez tout simplement vos fichiers et commandez en ligne
WWW.PCB-POOL.COM

Figure 11 : Démo 3 (“listing” HTML).

```
<html>
<head>
<title>Electronique Loisir Magazine Demo3</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<meta http-equiv="refresh" content="10">
</head>
<body bgcolor="#FFFFFF" background="sfondo1.gif">
<table width="300" border="0" align="center" cellpadding="0" cellspacing="0" bordercolor="#FFFFFF" bgcolor="#FFFFFF">
  <tr>
    <td colspan="3"><div align="center"><img src="Electronique Loisir Magazine1.jpg" width="100" height="37"></div></td>
  </tr>
  <tr>
    <td colspan="3"><div align="center"><font color="#999999" size="2" face="Arial, Helvetica, sans-serif">WEB
        SERVER COPROCESSOR <br>
        DEVELOPER BOARD FT497</font></div></td>
  </tr>
  <tr>
    <td colspan="3"><hr noshade></td>
  </tr>
  <tr>
    <td colspan="3"><div align="left"><font size="2" face="Arial, Helvetica, sans-serif">File:
        <strong>demo3.spd</strong></font></div></td>
  </tr>
  <tr>
    <td colspan="3"> </td>
  </tr>
  <tr bgcolor="#333333">
    <td colspan="3"> </td>
  </tr>
  <tr>
    <td width="98" bgcolor="#333333"><div align="center"><font color="#00FF00" size="3" face="Arial, Helvetica,
sans-serif"><strong>Temp
        =</strong></font></div></td>
    <td width="129" bgcolor="#333333"> <div align="center"><img src="/^Cifra3:1_BLK.jpg">
        <img src="/^Cifra2:1_BLK.jpg"> <img src="DP_BLK.jpg"> <img src="/^Cifra1:1_BLK.jpg"> </div></td>
    <td width="73" bgcolor="#333333"><div align="center"><font color="#00FF00" size="3" face="Arial, Helvetica,
sans-serif"><strong>&deg;C</strong></font></div></td>
  </tr>
  <tr bgcolor="#333333">
    <td colspan="3"> </td>
  </tr>
  <tr>
    <td colspan="3"> </td>
  </tr>
  <tr>
    <td colspan="3"><hr noshade></td>
  </tr>
  <tr>
    <td colspan="3"> <div align="center"> <font color="#999999" size="1" face="Arial, Helvetica, sans-
serif">&copy;
        2005 Électronique Loisir Magazine. All rights reserved.</font><br>
      </div></td>
  </tr>
</table>
</body>
</html>
```

au SitePlayer en trois objets, correspondant aux trois chiffres constituant la valeur de température. Par exemple, si la température est de 21,4 °C, les trois objets, qui s’appellent Cifra1, Cifra2 et Cifra3, vaudront respectivement 4,1 et 2.

Ceci établi, voyons maintenant comment utiliser ces trois objets pour obtenir la visualisation correcte de la température dans la page html. Pour ce faire, parcourons le “listing”, figure 11, du fichier index.htm. La partie réalisant la visualisation proprement dite est la suivante :

Figure 12 : Modifieurs applicables aux objets.

Opérations	Signification du modifieur
^oggetto:n	Visualise le n-ième chiffre de objet, en partant de la droite. Par exemple, si objet vaut 35, ^oggetto:1 visualise seulement le chiffre 5.
^oggetto+n	Additionne n à objet et visualise le total.
^oggetto-n	Soustrait n a objet et visualise le total.
^oggetto*n	Multiplie objet par n et visualise le total.
^oggetto/n	Divise objet par n et visualise le total.
^oggetto&n	Exécute l'AND logique entre n et objet et visualise le total.
^oggetto|n	Exécute l'OR logique entre n et objet et visualise le total.
^oggetto~n	Exécute le XOR logique entre n et objet et visualise le total.
^oggetto#n	Exécute l'AND logique entre n et objet et visualise "CHECKED" si le résultat est différent de zéro ou bien ne visualise rien si le résultat est 0.
^oggetto$n	Si objet = n visualise "CHECKED" sinon ne visualise rien.
^oggetto'n	Restitue le n-ième bit de objet en comptant à partir de la droite (0 est le premier bit).

```
<td width="129" bgcolor="#333333">
<div align="center">
<img src="/^Cifra3:1_BLK.jpg">
<img src="/^Cifra2:1_BLK.jpg">
<img src="DP_BLK.jpg">
<img src="/^Cifra1:1_BLK.jpg">
</div></td>
```

Pour la rendre plus lisible, omettons les diverses commandes de formatage et d'alignement des caractères, ce qui donne :

```
<img src="/^Cifra3:1_BLK.jpg">
<img src="/^Cifra2:1_BLK.jpg">
<img src="DP_BLK.jpg">
<img src="/^Cifra1:1_BLK.jpg">
```

Le tag img src indique l'insertion dans la page html d'une image. Typiquement on devrait trouver une syntaxe du type :

```
img src="image.jpg".
```

Pour nous cependant l'image à insérer dépend des objets Cifra3, Cifra2 et Cifra1. En effet avec Cifra3:1 on indique qu'il faut considérer le premier chiffre de l'objet Cifra3.

Dans l'exemple précédent, où la température valait 21,4 °C, Cifra3:1 vaut 2. Le nombre 2 remplace donc Cifra3 :1, ce qui donne 2_BLK.jpg.

Ainsi, on a la commande : **img src ="/2_BLK.jpg"**. En d'autres termes, on insère l'image 2_BLK.jpg, qui est l'image du chiffre 2. Avec une telle procédure les nombres 1 et 4 sont visualisés à partir de Cifra2 et Cifra1.

Avant Cifra1 est inséré le point décimal, qui cette fois est une image définie et correspond au fichier DP_BLK.

Rappelons que tous les fichiers image doivent nécessairement se trouver dans le même répertoire que le fichier index.htm. C'est seulement ainsi en effet qu'ils seront correctement traités par le SiteLinker au moment de la création du fichier à charger dans la mémoire "flash" du module.

Il faut prendre la précaution de ne pas réaliser d'images trop grandes, car la mémoire du SitePlayer est tout de même limitée !

Après avoir vu le fichier de la page Web, analysons le fichier de définition. Ce fichier aussi est très semblable aux précédents.

La seule différence est qu'on trouve dans le présent fichier trois objets nommés Cifra1, Cifra2 et Cifra3.

Fichier demo3.bas

C'est le fichier à introduire dans le PIC. Comme on peut le voir, il est assez semblable au précédent et donc nous n'expliquerons que la manière de trouver les trois chiffres à communiquer au SitePlayer.

Mais nous devons commencer par le fonctionnement de la sonde LM35 : elle fournit 10 mV par °C ; s'il fait 21,4 °C, la sonde fournit exactement :

21.4 * 10 mV = 214 mV.

Étant donné que U4a a un gain de 10 exactement, en sortie on aura :

214mV * 10 = 2140 V = 2.14 V.

Comme le fond d'échelle du convertisseur A / N est de 5 V (entre parenthèses, si on applique 0 V on obtient le nombre 0 et si on applique 5 V on obtient 255), à la tension de 2,14 V correspond le nombre :

(2.14 / 5) * 255, soit 109.

Si on multiplie à nouveau par deux ce nombre à travers le logiciel du micro, on obtient 218, qui n'est pas exactement 214 mais qui s'en rapproche assez, surtout si on considère la tolérance des résistances déterminant le gain de l'amplificateur opérationnel ! Une fois ce nombre obtenu, nous n'avons plus qu'à le décomposer sur les trois chiffres BCD et à l'envoyer au SitePlayer. L'opération de décomposition dans les chiffres BCD est effectuée par l'instruction Basic DIG. En fait les instructions :

CIFRA3 = DATALONG DIG 2,
CIFRA2 = DATALONG DIG 1
CIFRA1 = DATALONG DIG 0

mettent dans les trois variables CIFRA1, CIFRA2 et CIFRA3 les trois chiffres BCD de la valeur de température. Pour envoyer ces trois données nous utilisons le sous programme WRITETEMP qui envoie la commande de WRITE au module SitePlayer (cette commande vaut maintenant 130 car nous avons trois données à envoyer) et ensuite les trois chiffres.

Comment construire ce montage ?

Tout le matériel nécessaire pour construire cette platine d'expérimentation ET497, ainsi que le programmateur de PIC "in circuit" ET386 précédemment décrit, est disponible chez certains de nos annonceurs. Le module SitePlayer est disponible monté et essayé chez certains de nos annonceurs. Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp.

Les composants programmés sont disponibles sur www.electronique-magazine.com/mc.asp. ◆

MESURES & LABORATOIRES

FRÉQUENCEMÈTRE NUMÉRIQUE 10 HZ À 2 GHZ



Sensibilité (Veff.):
2,5 mV de 10 Hz à 1,5 MHz.
3,5 mV de 1,6 MHz à 7 MHz.
10 mV de 8 MHz à 60 MHz.
5 mV de 70 MHz à 800 MHz.
8 mV de 800 MHz à 2 GHz.
Base de temps sélectionnable: 0,1 - 1 - 10 sec.
Lecture sur 8 digits. Alimentation 220 VAC.

EN1374..... Kit complet avec boîtier ..... 195,15 €

FRÉQUENCEMÈTRE ANALOGIQUE



Ce fréquencemètre permet de mesurer des fréquences allant jusqu'à 100 kHz.
La sortie est à connecter sur un multimètre afin de visualiser la valeur.

EN1414..... Kit complet avec boîtier ..... 29,25 €

TRANSISTOR PIN-OUT CHECKER



Ce kit va vous permettre de repérer les broches E, B, C d'un transistor et de savoir si c'est un NPN ou un PNP. Si celui-ci est défectueux vous lirez sur l'afficheur "bAd".
Alimentation: pile de 9 V (non fournie).

EN1421..... Kit complet avec boîtier ..... 38,10 €

PRÉAMPLI D'INSTRUMENTATION 400 KHZ À 2 GHZ



Impédance d'entrée et de sortie: 52 Ω.
Gain: 20 dB env. à 100 MHz, 18 dB env. à 150 MHz, 16 dB env. à 500 MHz, 15 dB env. à 1000 MHz, 10 dB env. à 2000 MHz.
Figure de bruit: < à 3 dB. Alimentation: 9 Vcc (pile non fournie).

EN1169..... Kit complet avec boîtier..... 18,30 €

GÉNÉRATEUR DE MIRE POUR TV ET PC



Ce générateur de mire permet de tester tous les postes TV mais aussi les moniteurs pour PC.
Il possède 3 modes de fonctionnement: CCIR625, VGA 640*480, VGA 1024*768. La sortie peut-être de la vidéo composite ou du RGB. Une prise PERITEL permet de connecter la TV tandis qu'une prise VGA 15 points permet de connecter un moniteur. **Spécifications techniques:** Alimentation: 230 V / 50 Hz.
Type de signal: CCIR625 - VGA 640*480 - VGA 1024*768.
Type de sortie: RGB - Vidéo composite.
Connecteur de sortie: PERITEL - VGA 15 points.

EN1351..... Kit complet avec boîtier ..... 102,15 €

UN SELFMÈTRE HF...

...ou comment mesurer la valeur d'une bobine haute fréquence. En connectant une self HF quelconque, bobinée sur air ou avec support et noyau, aux bornes d'entrée de ce montage, on pourra prélever, sur sa prise de sortie, un signal HF fonction de la valeur de la self. En appliquant ce signal à l'entrée d'un fréquencemètre numérique, on pourra lire la fréquence produite. Connaissant cette fréquence, il est immédiatement possible de calculer la valeur de la self en µH ou en mH. Ce petit "selfmètre HF" n'utilise qu'un seul circuit intégré µA720 et quelques composants périphériques.



EN1522..... Kit complet avec boîtier ..... 30,00 €

TESTEUR DE FET



Cet appareil permet de vérifier si le FET que vous possédez est efficace, défectueux ou grillé.

EN5018..... Kit complet avec boîtier ..... 51,80 €

UN MESUREUR DE PRISE DE TERRE



Pour vérifier si la prise de terre d'une installation électrique est dans les normes et surtout si elle est efficace, il faut la mesurer et, pour ce faire, on doit disposer d'un instrument de mesure appelé Mesureur de Terre ou "Ground-Meter".

EN1512..... Kit complet avec boîtier ..... et galvanomètre ..... 62,00 €

TESTEUR DE MOSPOWER - MOSFET IGBT



D'une utilisation très simple, ce testeur universel permet de connaître l'état d'un MOSPOWER - MOSFET - IGBT.
Livré avec sondes de tests.

EN1272..... Kit complet avec boîtier ..... 19,70 €

ANALYSEUR DE SPECTRE POUR OSCILLOSCOPE



Ce kit vous permet de transformer votre oscilloscope en un analyseur de spectre performant. Vous pourrez visualiser n'importe quel signal HF, entre 0 et 310 MHz environ. Avec le pont réflectométrique EN1429 et un générateur de bruit, vous pourrez faire de nombreuses autres mesures...

EN1431..... Kit complet avec boîtier ..... sans alimentation..... 100,60 €
EN1432..... Kit alimentation ..... 30,60 €

CAPACIMÈTRE DIGITAL AVEC AUTOZÉRO



Cet appareil permet la mesure de tous les condensateurs compris entre 0,1 pF et 200 µF. Un bouton poussoir permet de compenser automatiquement les capacités parasites.
6 gammes sont sélectionnable par l'intermédiaire d'un commutateur présent en face avant.
Un afficheur de 4 digits permet la lecture de la valeur.
Spécifications techniques:
Alimentation: 230 V / 50 Hz.
Etendue de mesure: 0,1 pF à 200 µF. Gammes de mesure: 0,1 pF / 200 pF - 1 pF / 2 000 pF - 0,01 nF / 20 nF - 0,1 nF / 200 nF - 0,001 µF / 2 µF - 0,1 µF / 200 µF.
Autozéro: oui. Affichage: 5 digits.

EN1340..... Kit completavec boîtier ..... 124,25 €

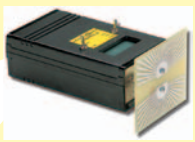
UN GÉNÉRATEUR BF À BALAYAGE



Afin de visualiser sur l'écran d'un oscilloscope la bande passante complète d'un amplificateur Hi-Fi ou d'un préamplificateur ou encore la courbe de réponse d'un filtre BF ou d'un contrôle de tonalité, etc., vous avez besoin d'un bon sweep generator (ou générateur à balayage) comme celui que nous vous proposons ici de construire.

EN1513..... Kit complet avec boîtier ..... 85,00 €
ENCAB3..... Ensemble de trois câbles BNC/BNC ..... 18,00 €

POLLUOMÈTRE HF...



...ou comment mesurer la pollution électromagnétique.
Cet appareil mesure l'intensité des champs électromagnétiques HF, rayonnés par les émetteurs FM, les relais de télévision et autres relais téléphoniques.

EN1435..... Kit complet avec boîtier ..... 93,00 €

VFO PROGRAMMABLE DE 20 MHZ À 1,2 GHZ



Ce VFO est un véritable petit émetteur avec une puissance HF de 10 mW sous 50 Ω. Il possède une entrée modulation et permet de couvrir la gamme de 20 à 1 200 MHz avec 8 modules distincts (EN1235/1 à EN1235/8). Basé sur un PLL, des roues codeuses permettent de choisir la fréquence désirée. Puissance de sortie: 10 mW. Entrée: modulation. Alim.: 220 VAC. Gamme de fréquence: 20 à 1 200 MHz en 8 modules.

EN1234..... Kit complet avec boîtier ..... et 1 module au choix ..... 158,40 €

MODULES CMS



Modules CMS pour le EN1234/K, livrés montés.

EN1235-1.. Module 20 à 40 MHz..... 19,70 €
EN1235-2.. Module 40 à 85 MHz..... 19,70 €
EN1235-3.. Module 70 à 150 MHz..... 19,70 €
EN1235-4.. Module 140 à 250 MHz..... 19,70 €
EN1235-5.. Module 245 à 405 MHz..... 19,70 €
EN1235-6.. Module 390 à 610 MHz..... 19,70 €
EN1235-7.. Module 590 à 830 MHz..... 19,70 €
EN1235-8.. Module 800 MHz à 1,2 GHz.... 19,70 €

ALTIMÈTRE DE 0 À 1 999 MÈTRES



Avec ce kit vous pourrez mesurer la hauteur d'un immeuble, d'un pylône ou d'une montagne jusqu'à une hauteur maximale de 1 999 m.

EN1444..... Kit complet avec boîtier ..... 62,35 €

GÉNÉRATEUR PROFESSIONNEL 2 HZ - 5 MHZ



D'une qualité professionnelle, ce générateur intègre toutes les fonctions nécessaires à un bon appareil de laboratoire. Trois types de signaux disponibles: sinus - carré - triangle. Leur fréquence peut varier de 2 Hz à 5 MHz. Deux sorties (50 Ω et 600 Ω) permettent de piloter plusieurs types d'entrées. Un atténuateur de 0 à –20 dB peut être commuté. Niveau de sortie variable de 0 à 27 Vpp. Le réglage de la fréquence de sortie s'effectue avec deux potentiomètres (réglage "rapide" et calibrage "fin". L'afficheur de 5 digits permet de contrôler la fréquence de sortie. 6 gammes de fréquences sont disponibles. Une tension d'offset peut être insérée de façon à décaler le signal de sortie. Cet appareil permet aussi de régler le rapport cyclique du signal sélectionné. Une fonction "sweep" permet un balayage de la fréquence de sortie. Ce balayage, réglable par potentiomètre, couvre toute la gamme de fréquence sélectionnée.
Cette fonction est très intéressante pour la mesure de bobine et de filtre dans le domaine de la HF. Alimentation: 230 V / 50 Hz. Gammes de fréquences: 2 Hz / 60 Hz - 60 Hz / 570 Hz - 570 Hz / 5,6 kHz - 5,6 kHz / 51 kHz - 51 kHz / 560 kHz - 560 kHz / 5 MHz. Sortie trigger: oui.

EN1345..... Kit complet avec boîtier ..... 282,00 €

SONDE LOGIQUE TTL ET CMOS



Cette sonde vous rendra les plus grands services pour dépanner ou élaborer des cartes électroniques contenant des circuits logiques CMOS ou TTL

EN1426 Kit complet avec boîtier..... 27,30 €

UN DÉTECTEUR DE FUITES SHF POUR FOURS À MICRO-ONDES



Avec ce détecteur de fuite d'ondes SHF pour four à micro-ondes nous complétons la série de nos instruments de détection destinés à contrôler la qualité des conditions environnementales de notre existence, comme les détecteurs de fuite de gaz, de champs magnétiques et HF, les compteurs Geiger, etc...

EN1517..... Kit avec boîtier plastique: ..... 27,00 €

IMPÉDANCEMÈTRE RÉACTANCEMÈTRE NUMÉRIQUE



Cet appareil permet de connaître la valeur Ohmique d'un dipôle à une certaine fréquence. Les applications sont nombreuses: impédance d'un haut-parleur, d'un transformateur audio, de l'entrée d'un amplificateur audio, d'un filtre "Cross-Over", de l'inductance parasite d'une résistance, la fréquence de résonance d'un haut-parleur, etc..
Gamme de mesure: 1 Ω à 99,9 kΩ en 4 échelles.
Fréquences générées: 17 Hz à 100 kHz variable.
Niveau de sortie: 1 Veff. Alimentation: 220 VAC.

EN1192..... Kit complet avec boîtier ..... 154,75 €

TESTEUR POUR LE CONTRÔLE DES BOBINAGES



Permet de déceler des spires en court-circuit sur divers types de bobinages comme transformateurs d'alimentation, bobinages de moteurs, selfs pour filtres Hi-Fi.

EN1397..... Kit complet avec boîtier ..... 19,05 €

INDUCTANCEMÈTRE 10 µH À 10 MH



À l'aide de ce simple inductancemètre, vous pourrez mesurer des selfs comprises entre 10 µH et 10 mH. La lecture de la valeur se fera sur un multimètre analogique ou numérique (non fourni).

EN1422..... Kit complet avec boîtier ..... 42,70 €

DÉCIBELMÈTRE



A l'aide de ce kit vous allez pouvoir mesurer le niveau sonore ambiant.
Gamme couverte: 30 dB à 120 dB.
Indication: par 20 LED.
Alimentation: 9 V (pile non fournie).

EN1056..... Kit complet avec boîtier ..... 51,70 €

GÉNÉRATEUR DE BRUIT BF



Couplé à un analyseur de spectre, ce générateur permet le réglage de filtre BF dans beaucoup de domaine: réglage d'un égaliseur, vérification du rendement d'une enceinte acoustique etc.
Couverture en fréquence: 1 Hz à 100kHz. Filtre commutable: 3 dB / octave env. Niveau de sortie: 0 à 4 Veff. env. Alimentation: 12 Vcc.

EN1167..... Kit complet avec boîtier ..... 33,55 €

GÉNÉRATEUR DE BRUIT 1MHZ À 2 GHZ



Signal de sortie: 70 dBV. Fréquence max.: 2 GHZ. Linéarité: +/- 1 dB. Fréquence de modulation: 190 Hz env. Alimentation: 220 VAC.

EN1142..... Kit complet avec boîtier ..... 79,00 €

DÉTECTEUR DE TÉLÉPHONES PORTABLES



Ce détecteur vous apprend, en faisant sonner un buzzer ou en allumant une LED, qu'un téléphone portable, dans un rayon de 30 mètres, appelle ou est appelé. Ce précieux appareil trouvera son utilité dans les hôpitaux (où les émissions d'un portable peuvent gravement perturber les appareils de surveillance vitale), chez les médecins, dans les stations service, les cinémas et, plus généralement, dans tous les services privés ou publics où se trouvent des dispositifs ou des personnes sensibles aux perturbations radioélectriques. On peut, grâce à ce détecteur, vérifier que le panneau affichant "Portables interdits" ou "Eteignez vos portables" est bien respecté.

EN1523..... Kit complet avec boîtier ..... 30,00 €

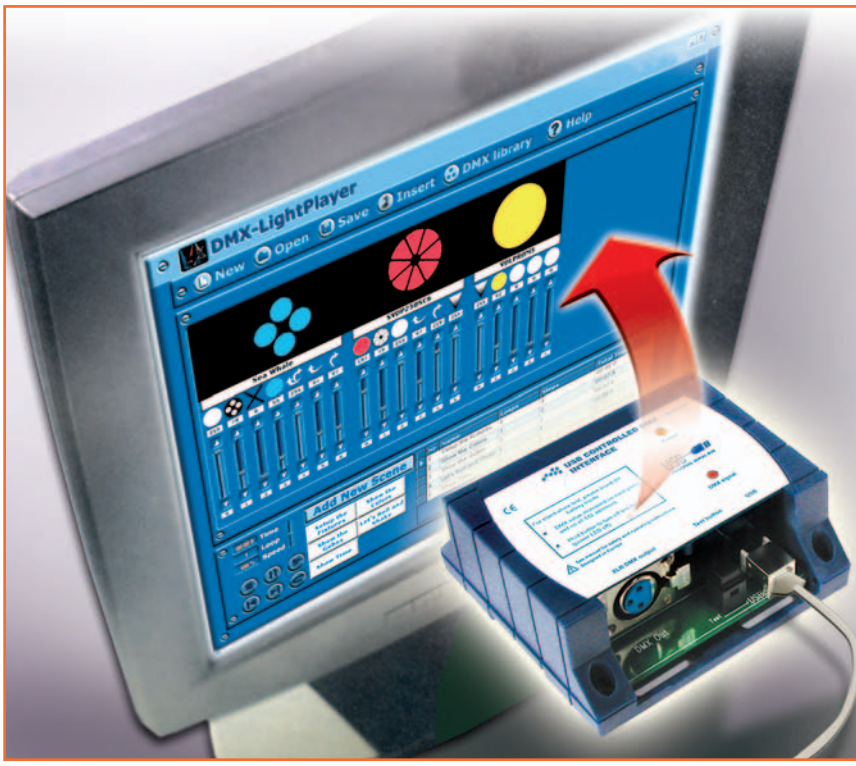
COMELEC
Tél.: 04 42 70 63 90 • Fax: 04 42 70 63 95
CD 908 - 13720 BELCODENE
Visitez notre site www.comelec.fr

Expéditions dans toute la France. Moins de 5 kg : Port 8,40 €. Règlement à la commande par chèque, mandat ou carte bancaire. Bons administratifs acceptés.
Le port est en supplément. De nombreux kits sont disponibles, envoyez votre adresse et cinq timbres, nous vous ferons parvenir notre catalogue général.

EV8062

Un contrôleur DMX sur port USB

Ce système de régie lumière au protocole DMX512 comporte une interface USB et bien sûr un programme de gestion. Toutes les ressources nécessaires (DLL et routines de test) sont fournies afin que vous puissiez réaliser des programmes personnalisés en Delphi, Visual Basic.NET et C++ Builder.



CARACTÉRISTIQUES TECHNIQUES

- Connexion USB
- 512 canaux
- 256 niveaux de luminosité pour chaque canal
- Fusible de protection de sortie
- Batterie optionnelle pour test en “stand alone” (mode autonome)
- Connecteur de sortie à 3 broches de type XLR-DMX
- Logiciel et DLL inclus pour la gestion
- Dimensions: 106 x 101 x 44,5 mm
- Câble USB inclus.

Cet article vous propose de monter une interface USB en mesure de piloter n’importe quel appareil au standard DMX512. Nous vous fournissons en outre un programme complet de gestion pouvant être utilisé pour commander toute régie lumière obéissant à ce protocole: ce programme permet aussi de réaliser et de mémoriser des séquences pour éclairer des spectacles, de musique, de théâtre, etc. (voir figure 5) et tout cela à un coût incomparablement plus bas que pour du matériel professionnel...lors même que le nôtre n’a rien à lui envier! En effet, des interfaces de ce type s’achètent dans le commerce entre 300 et 500 euro, alors que l’acquisition du matériel nécessaire à la réalisation de notre contrôleur DMX512 vous coûtera peut-être le quart de cette somme. Aussi, ceux qui n’ont jamais osé, pour des raisons de prix, s’équiper avec du matériel de régie fonctionnant selon ce protocole, pourront enfin s’y lancer et se féliciter ensuite de ses performances et de sa commodité.

Mais ce n’est pas tout: en plus des schémas et de l’assurance de trouver le matériel auprès de nos annonceurs (comme d’habitude), nous vous fournissons la DLL et toutes les informations nécessaires à la réalisation de programmes personnalisés en Delphi, Visual Basic, C++ Builder ou tout autre outil de développement sous Windows à 32 bits supportant l’appel d’une DLL (voir figure 6). Ce montage a donc, une fois de plus, une vocation didactique supplémentaire de grande importance: ceux qui étudient l’informatique ou l’électronique (ou les deux conjointement) pourront mettre à profit leurs connaissances et réaliser des programmes personnalisés en mesure d’interagir avec le monde réel à travers cette interface.

Mais revenons au matériel: la liaison de cette interface au PC se fait par le port USB (qui, comme son nom l’indique, devient vraiment universel). L’ordinateur utilisé devra donc comporter un tel port et celui-ci devra être libre (voir figure

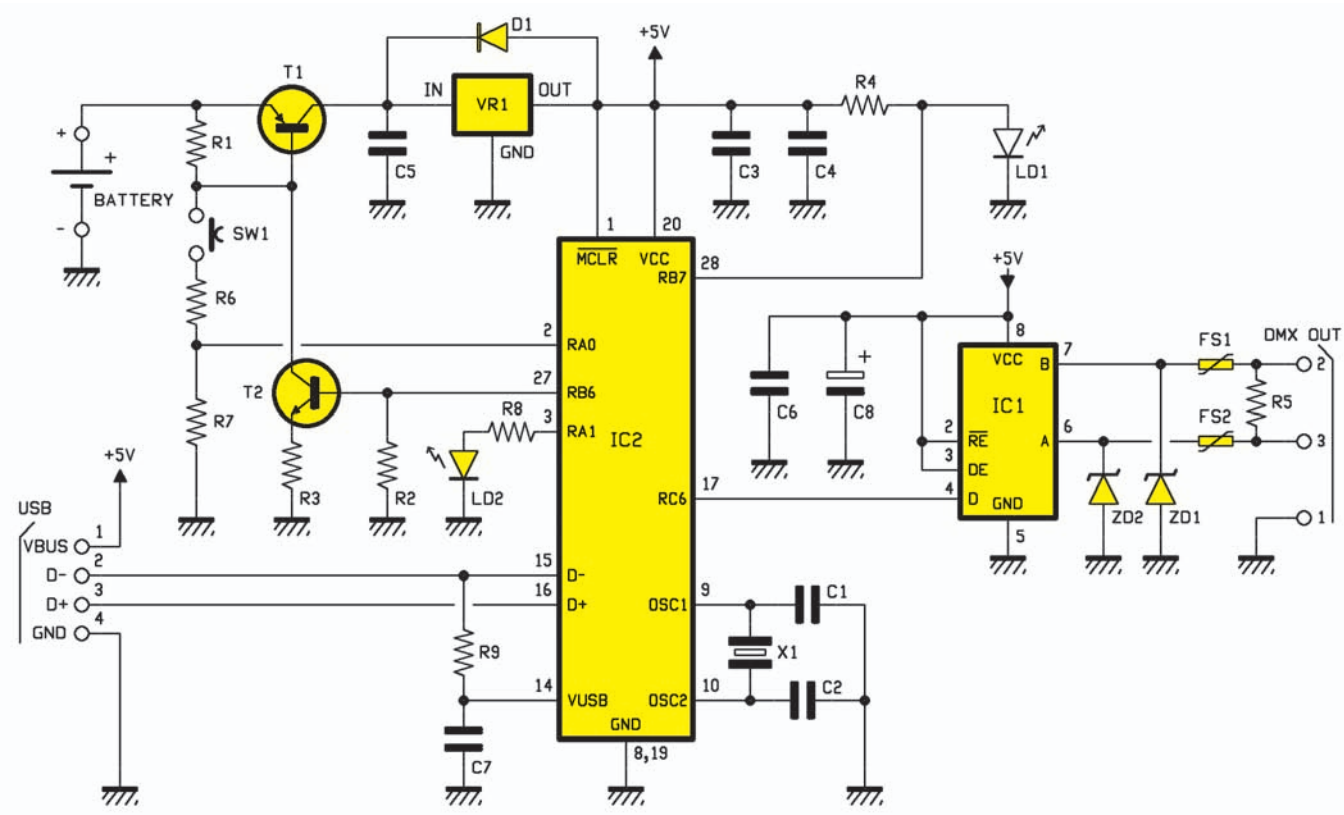


Figure 1: Schéma électrique du contrôleur DMX..

5 les autres resources nécessaires). Quant à l'alimentation de l'interface, eh bien, USB oblige, elle se fait à travers le port dont l'un des avantages est de fournir la tension continue de 5 V provenant du PC.

Nous avons prévu également la possibilité de faire fonctionner l'interface en "stand alone" (autonome), c'est-à-dire sans ordinateur, pour tester les systèmes DMX: dans ce cas il sera nécessaire d'alimenter la platine avec une pile de 9 V. Le système produit l'une après l'autre les 512 adresses et donne à chacune un niveau prédéfini.

Le schéma électrique

Comme le montre le schéma électrique de la figure 1, le cœur de l'interface DMX/USB est un microcontrôleur Microchip PIC16C745 déjà programmé en usine (IC2). Cette puce est un des rares micro à 8 bits qui intègre une interface USB.

Les lignes D+ et D- du connecteur USB sont reliées directement aux broches correspondantes du PIC, plus précisément aux 16 et 15, alors que la ligne GND est reliée à la masse du circuit et que le + 5 V alimente le PIC et le convertisseur TTL / RS485. Le circuit comporte aussi une autre section d'alimentation, celle qui correspond à T1 et T2 et au régulateur VR1: son rôle est de fournir aux deux circuits intégrés une tension stabilisée de 5 V en partant de la tension fournie par la pile (si on souhaite faire fonctionner l'interface en mode "stand alone"); un banal 7805 suffit à cet effet; l'étage, qui comporte en outre deux transistors et un poussoir, est relié aux ports RA0 et RB6 du PIC.

La ligne RA0 est une entrée numérique capable de reconnaître combien de fois et pendant combien de temps SW1 a été pressé. Il est ainsi possible de programmer le niveau à donner aux 512 canaux DMX quand l'interface est utilisée sans ordinateur. Le poussoir permet aussi d'inhiber cet étage: en effet, si nous maintenons pressé SW1 plus de trois secondes, la ligne RB6, utilisée comme sortie, bloque T1 (monté en série avec le positif d'alimentation).

L'horloge du micro bat au rythme d'un quartz de 6 MHz et LD1 (présence alimentation) et LD2 (signal DMX) se chargent de la signalisation. LD2 est pilotée par le port RA1 (utilisé comme ligne numérique de sortie).

Le circuit est complété par le convertisseur TTL / RS485 (IC1), un SN75176: mais on peut le remplacer par un MAX485.

Son rôle est de convertir la donnée présente sur la ligne RC6 (broche 17 du micro) de TTL (niveau 0 / 5 V) en RS485, soit en une ligne différentielle capable d'offrir une grande insensibilité aux parasites ainsi qu'une portée nettement supérieure à celle des autres protocoles série (comme le RS232).

Deux zener et deux fusibles à réarmement (polyswitchs) protègent la ligne (c'est-à-dire le ci) contre les parasites éventuellement captés par les câbles, surtout s'ils sont très longs (et on sait que cela est un des avantages du RS485 que de permettre de grandes longueurs de câble).

Pour fonctionner correctement et aussi pour éviter d'écoper tout de même des parasites gênants, la ligne doit être terminée...par un "terminator", justement (bouchon), constitué en l'occurrence par une résistance de 100 ou 120 ohms. Quelques condensateurs opportunément semés le long de la ligne d'alimentation compléteront l'élimination des parasites indésirables en les envoyant à la masse.

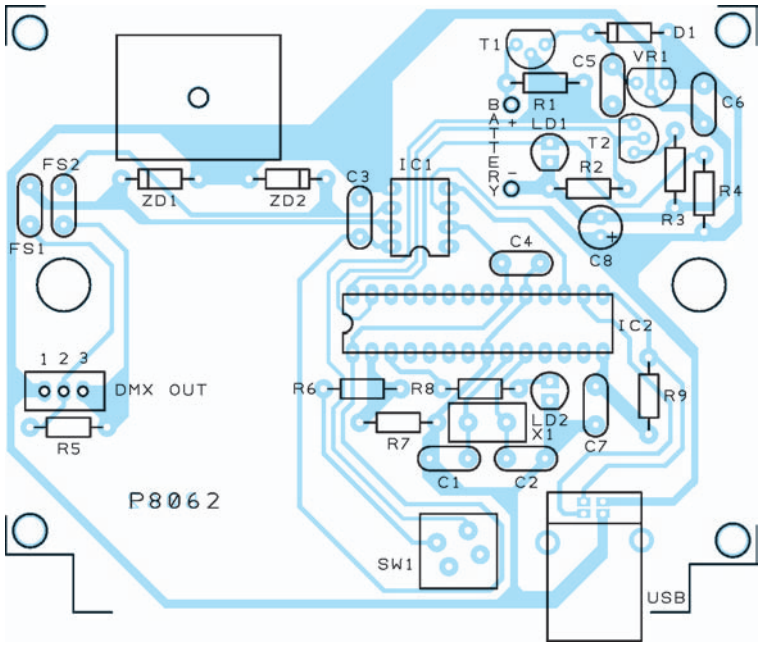


Figure 2a : Schéma d'implantation des composants du contrôleur DMX.

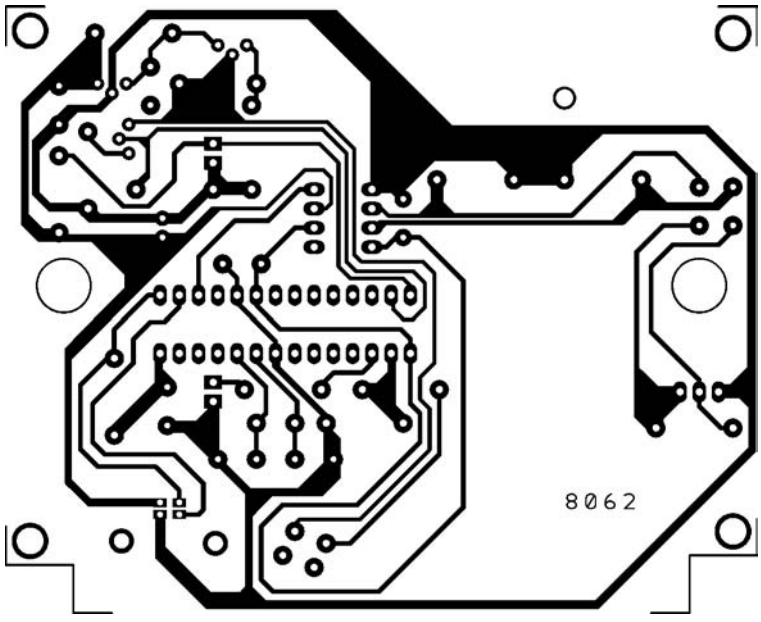


Figure 2b : Dessin, à l'échelle 1, du circuit imprimé du contrôleur DMX.

Liste des composants

R1..... 10 k
R2..... 10 k
R3..... 1,5 k
R4..... 470
R5..... 1,5 k
R6..... 1,5 k
R7..... 2,2 k
R8..... 470
R9..... 1,5 k
C1..... 33 pF céramique
C2..... 33 pF céramique
C3..... 100 nF multicouche
C4..... 100 nF multicouche
C5..... 100 nF multicouche
C6..... 100 nF multicouche
C7..... 220 nF multicouche
C8..... 4,7 µF 50 V électrolytique
D1..... 1N4148
ZD1 ... zener 6,2 V 1/2 W
ZD2 ... zener 6,2 V 1/2 W
T1 ..... BC327
T2 ..... BC337
VR1.... 78L05
SW1... poussoir pour ci
IC1..... SN75176 ou MAX485
IC2..... PIC16C745-EK8062
LD1.... LED 3 mm rouge
LD2.... LED 3 mm rouge
X1 ..... quartz 6 MHz
FS1.... PTC 3 A 60 Vdc
FS2.... PTC 3 A 60 Vdc

Divers :

1 support 2 x 4
1 support 2 x 14
1 prise de pile
1 clip pour pile
1 barrette verticale mâle 3 broche
1 connecteur USB à 90° (horizontal) pour ci
1 connecteur XLR 3 pôles femelle
1 boulon 3 MA 6 mm
1 boîtier plastique format "barre DIN"

Sauf spécification contraire, toutes les résistances sont des 1/4 W à 5 %.

La réalisation pratique

Les dimensions de la platine sont établies pour qu'elle puisse ensuite prendre place dans un boîtier au format "barre DIN" dont vous avez désormais l'habitude.

Vous trouverez le dessin à l'échelle 1 du circuit imprimé simple face figure 2b. Quand vous l'avez devant vous, montez tous les composants en commençant par les supports des circuits intégrés et en terminant par les périphériques, poussoir SW1, connecteur barrette DMX OUT, connecteur USB et clip de pile 9 V (si vous ne pensez pas utiliser l'appareil sans ordinateur, inutile de prévoir ce qui concerne cette dernière). Vous insèrerez les circuits intégrés qu'après la dernière soudure terminée.

Le connecteur XLR à 3 pôles est fixé sur un côté du boîtier (voir photo de première page), la liaison au circuit imprimé se faisant par le connecteur barrette à 3 pôles (connecteurs M / F et câble à trois fils, soudés côté XLR). La platine est à fixer dans le boîtier et l'éventuelle pile est logée à l'intérieur dans son clip. Fermez le boîtier avec des vis.

Il vous reste à relier l'interface au PC

avec un câble USB. Si vous voulez utiliser votre montage pour tester une chaîne de régie lumière au standard DMX512, vous pouvez, au moins pour cet usage, faire sans ordinateur, en suivant la procédure ci-après.

L'utilisation en mode "stand-alone"

Dans ce mode, il faut faire usage de la pile de 9 V car on ne bénéficiera pas du 5 V en provenance de l'ordinateur via le câble USB (et pour cause, on n'aura ni l'un ni l'autre).

À travers le connecteur XLR à trois pôles, reliez les projecteurs et les spots en aval du circuit et vérifiez que la chaîne se termine bien par la résistance de 100 à 120 ohms.

Une légère pression sur le poussoir SW1 détermine l'activation de l'appareil (la LED de M / A s'allume). Le circuit commence alors à envoyer une trame complète , c'est-à-dire une série de commandes pour les 512 (en théorie) unités de puissance (reliées aux projecteurs qu'elles pilotent).

Combien de ces unités existent effectivement et donc combien d'adresses ont été paramétrées, cela importe peu : les impulsions de contrôle agiront sur tous les éléments de la chaîne.

Initialement, le niveau appliqué aux 512 canaux est 0 et par conséquent les sorties n'allument aucun projecteur. Mais à chaque pression sur le poussoir de contrôle, le niveau augmente d'un "cran" (d'un pas : il y en a de 0 à 255) et la luminosité du projecteur augmente aussi légèrement. LD2 (sortie DMX) commence à clignoter en même temps que l'envoi de la séquence de commande et la durée du clignotement augmente au fur et à mesure que le niveau appliqué augmente aussi.

Si, par exemple, nous pressons le poussoir 127 fois, la luminosité des projecteurs arrivera à la moitié de la valeur maximale ; si nous pressons 255 fois le poussoir, nous obtenons la luminosité maximale. Si après 255 fois nous continuons à presser le poussoir, la luminosité diminue progressivement pour s'annuler si nous pressons encore 255 fois.

Utilisé en mode test, notre appareil permet de vérifier facilement, sans avoir besoin d'un ordinateur, n'importe quel variateur et, plus généralement, n'importe quel appareil suivant le protocole DMX512. Pour éteindre le dispositif, il suffit de presser SW1 pendant plus de trois secondes.

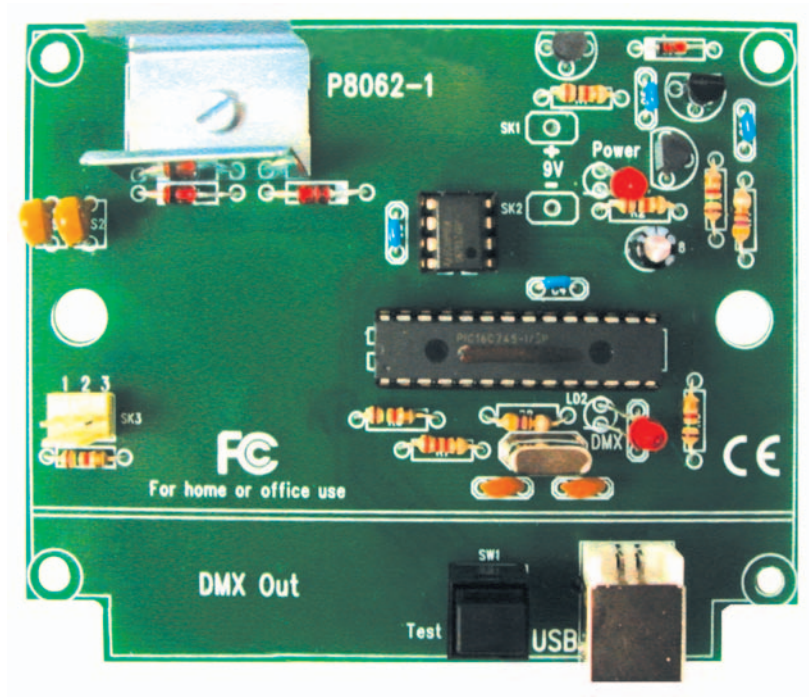


Figure 3 : Photo d'un des prototypes de la platine du contrôleur DMX.

Le logiciel

Voyons maintenant comment utiliser l'interface avec les programmes que nous avons mis au point pour elle.

Ces programmes sont au nombre de deux : l'un est très simple (programme demo), il permet de tester les fonctions de l'interface DMX et de la régie lumière ; l'autre, bien plus complet (programme DMX Light Player), de contrôler le fonctionnement de n'importe quel système de lumières (même les plus complexes multi canaux).

Nous donnons l'exécutable des deux programmes : ceux qui voudront seulement utiliser notre interface n'auront rien d'autre à faire que de lancer les programmes et de suivre les instructions d'utilisation.

En revanche, ceux qui souhaitent réaliser un programme personnalisé pourront se référer aux routines de communication contenues dans la DLL ("Dynamic Link Library") nommée K8062D.DLL et utilisée dans notre programme (elle est disponible avec l'interface).

Si l'on appelle les fonctions et les procédures exportées à partir de la DLL, on peut en effet écrire des applications personnalisées Windows (98SE, 2000, Me, XP) en Delphi, Visual Basic, C++ Builder ou n'importe quel autre outil de développement supportant l'appel à DLL. La figure 6 donne des exemples permettant de comprendre comment construire soi-même ses propres programmes d'application.

Ces exemples, écrits en Delphi, Visual Basic, Visual Basic.NET et C++ Builder) contiennent les déclarations complètes sur les fonctions et procédures DLL et les exemples donnés dans la description de la DLL sont écrits uniquement pour le Delphi.

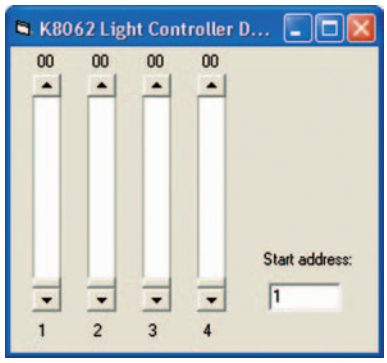
Panoramique des procédures et des fonctions de la K8062D.DLL

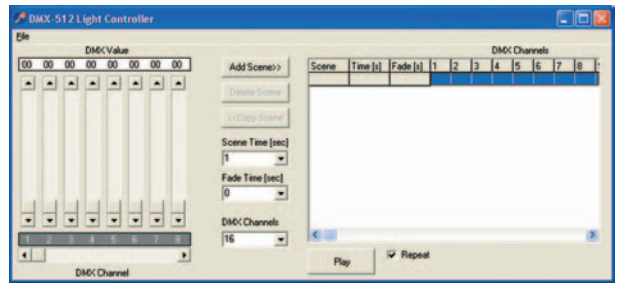
Procédures générales
StartDevice: Ouvre la liaison avec le dispositif;
StopDevice: Ferme la liaison avec le dispositif;
Procédures de sortie
SetChannelCount(Count): Définit le nombre maximal de canaux DMX utilisés;
SetData(Channel, Data): Définit la valeur du canal DMX;

Procédures et fonctions de la K8062D.DLL

StartDevice
Ouvre la communication avec l'interface USB et charge les pilotes nécessaires à

Figure 4 : Ecrans obtenus avec les programmes fournis.

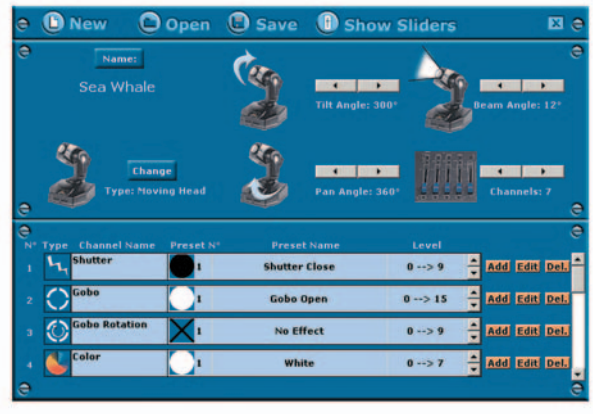




Les programmes demo fournis permettent de vérifier le fonctionnement de l'interface DMX-USB ainsi que celle des régies de lumière comme les variateurs. Ces programmes peuvent être facilement personnalisés en utilisant les ressources logicielles mise à votre disposition, comme la DLL spécifique nommée K8062D. Les exemples de la figure 6 (écrits en Delphi, Visual Basic, Visual Basic .NET et C++ Builder) permettent une approche plus facile du logiciel.

Figure 5 : Le logiciel DMX LIGHT PLAYER.





Le programme de régie des lumières fourni avec l'interface USB (nommée DMX Light Player) est particulièrement simple et intuitif à utiliser. Il permet de contrôler directement et en temps réel n'importe quel système d'éclairage de salle ou de scène de spectacle, même les plus complexes et multi canaux. Il est possible aussi de paramétrer et de mémoriser des séquences à appeler quand c'est nécessaire. Nous pouvons ainsi reproduire automatiquement toute la séquence des éclairages d'un spectacle. La création de séquences personnalisées n'est pas davantage un problème, même pour quelqu'un qui n'est pas spécialiste des éclairages commandés par PC (un régisseur lumières formé à l'ancienne ne s'arrachera pas les cheveux !). Le programme fonctionne seulement avec l'interface DMX présentée dans cet article et exige l'utilisation d'un ordinateur ayant les caractéristiques suivantes :

- Processeur Pentium II ou supérieur
- 256 Mo de RAM
- Port USB libre
- Résolution video 1024 x 768 16 bits ou supérieur
- Lecteur de CDROM disponible
- Souris.



Dans la bibliothèque interne se trouve déjà une série de systèmes d'éclairage auxquels on peut en ajouter d'autres. Il est possible d'importer les pilotes correspondants et de paramétrer manuellement les caractéristiques du nouveau système. Les valeurs par défaut de chaque dispositif peuvent être paramétrées, tout comme il est possible d'effectuer le couplage à un canal DMX spécifique. En ce qui concerne la création et la mémorisation des séquences de lumières, comme toujours en pareil cas, il est nécessaire de programmer pas à pas la luminosité des divers projecteurs ou spots ; avec un certain nombre de pas on constitue des scènes et en reliant entre elles plusieurs scènes on obtient les séquences d'un spectacle. Chaque séquence peut durer 5 à 10 minutes alors que les scènes durent 30 à 60 secondes. Bien sûr cela dépend du type de spectacle. Bien que les indications soient en Anglais, l'utilisation est, encore une fois, intuitive : au bout de quelques minutes de pratique, vous deviendrez un régisseur de lumières très expert, tout en restant "zen" au cours d'un spectacle dont vous auriez la responsabilité des éclairages.

Figure 6 : Exemples de “listings”.

a) Un exemple en Delphi

```
unit K8062_1;

interface

uses
	Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
	StdCtrls;
type
	TForm1 = class(TForm)
		ScrollBar1: TScrollBar;
		ScrollBar2: TScrollBar;
		ScrollBar3: TScrollBar;
		ScrollBar4: TScrollBar;
		Edit1: TEdit;
		Label1: TLabel;
		Label2: TLabel;
		Label3: TLabel;
		Label4: TLabel;
		Label5: TLabel;
		Label6: TLabel;
		Label7: TLabel;
		Label8: TLabel;
		Label9: TLabel;
		procedure Edit1Change(Sender: TObject);
		procedure FormCreate(Sender: TObject);
		procedure FormClose(Sender: TObject; var Action: TCloseAction);
		procedure ScrollBar1Change(Sender: TObject);
		procedure ScrollBar2Change(Sender: TObject);
			procedure ScrollBar3Change(Sender: TObject);
			procedure ScrollBar4Change(Sender: TObject);
	private
	{ Private declarations }
	public
	{ Public declarations }
	end;
var
	Form1: TForm1;
	StartAddress: Longint;

implementation

{$R *.DFM}
PROCEDURE StartDevice; stdcall; external 'K8062d.dll';
PROCEDURE SetData(Channel: Longint ; Data: Longint); stdcall; external K8062d.dll';
PROCEDURE SetChannelCount(Count: Longint); stdcall; external 'K8062d.dll';
PROCEDURE StopDevice; stdcall; external 'K8062d.dll';

procedure TForm1.FormCreate(Sender: TObject);
begin
	StartDevice;
	StartAddress:=1;
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
	StopDevice;
end;

procedure TForm1.Edit1Change(Sender: TObject);
begin
	if (StrToInt(Edit1.Text)>0) and (StrToInt(Edit1.Text)< 509) then
```

```
	begin
		SetChannelCount(StartAddress+ 3);
		StartAddress:=StrToInt(Edit1.Text);
		Label5.Caption:=IntToStr(StartAddress);
		Label6.Caption:=IntToStr(StartAddress+ 1);
		Label7.Caption:=IntToStr(StartAddress+ 2);
		Label8.Caption:=IntToStr(StartAddress+ 3);
	end;
end;

procedure TForm1.ScrollBar1Change(Sender: TObject);
begin
	Label1.Caption:=IntToStr(255-ScrollBar1.Position);
	etData(StartAddress, 255-ScrollBar1.Position);
end;

procedure TForm1.ScrollBar2Change(Sender: TObject);
begin
	Label2.Caption:=IntToStr(255-ScrollBar2.Position);
	SetData(StartAddress+1, 255-ScrollBar2.Position);
end;

procedure TForm1.ScrollBar3Change(Sender: TObject);
begin
	Label3.Caption:=IntToStr(255-ScrollBar3.Position);
	SetData(StartAddress+2, 255-ScrollBar3.Position);
end;

procedure TForm1.ScrollBar4Change(Sender: TObject);
begin
	Label4.Caption:=IntToStr(255-ScrollBar4.Position);
	SetData(StartAddress+3, 255-ScrollBar4.Position);
end;

end.
```

b) Un exemple en Visual Basic.NET

```
Public Class Form1
	Inherits System.Windows.Forms.Form

	Private Declare Sub StartDevice Lib «k8062d.dll» ()
	Private Declare Sub SetData Lib «k8062d.dll» (ByVal Channel As Integer, ByVal Data As
Integer)
	Private Declare Sub SetChannelCount Lib «k8062d.dll» (ByVal Count As Integer)
	Private Declare Sub StopDevice Lib «k8062d.dll» ()
	Dim StartAddress As Integer

	Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles
MyBase.Load
		StartDevice()
		StartAddress = 1
	End Sub

	Private Sub Form1_Closed(ByVal sender As Object, ByVal e As System.EventArgs) Han-
dles
MyBase.Closed
		StopDevice()
	End Sub

	Private Sub VScrollBar1_Scroll(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.ScrollEventArgs) Handles VScrollBar1.Scroll
		Label1.Text = Str(255 - VScrollBar1.Value)
		SetData(StartAddress, 255 - VScrollBar1.Value)
	End Sub
```

```
	Private Sub VScrollBar2_Scroll(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.ScrollEventArgs) Handles VScrollBar2.Scroll
		Label2.Text = Str(255 - VScrollBar2.Value)
		SetData(StartAddress + 1, 255 - VScrollBar2.Value)
	End Sub

	Private Sub VScrollBar3_Scroll(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.ScrollEventArgs) Handles VScrollBar3.Scroll
		Label3.Text = Str(255 - VScrollBar3.Value)
		SetData(StartAddress + 2, 255 - VScrollBar3.Value)
	End Sub

	Private Sub VScrollBar4_Scroll(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.ScrollEventArgs) Handles VScrollBar4.Scroll
		Label4.Text = Str(255 - VScrollBar4.Value)
		SetData(StartAddress + 3, 255 - VScrollBar4.Value)
	End Sub

	Private Sub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
		If Val(TextBox1.Text) > 0 And Val(TextBox1.Text) < 510 Then
			StartAddress = Val(TextBox1.Text)
			SetChannelCount(StartAddress + 3)
			Label5.Text = Str(StartAddress)
			Label6.Text = Str(StartAddress + 1)
			Label7.Text = Str(StartAddress + 2)
			Label8.Text = Str(StartAddress + 3)
		End If
	End Sub
End Class
```

c) Un exemple en Visual Basic

```
Option Explicit
Private Declare Sub StartDevice Lib «k8062d.dll» ()
Private Declare Sub SetData Lib «k8062d.dll» (ByVal Channel As Long, ByVal Data As Long)
Private Declare Sub SetChannelCount Lib «k8062d.dll» (ByVal Count As Long)
Private Declare Sub StopDevice Lib «k8062d.dll» ()

Private Sub Form_Load()
	StartDevice
End Sub

Private Sub Form_Terminate()
	StopDevice
End Sub

Private Sub UpdateLabels()
	Dim i As Integer
	Dim n As Integer
	n = 0
	If (Val(Text1.Text) > 0) And (Val(Text1.Text) < 510) Then
		For i = Val(Text1.Text) To Val(Text1.Text) + 3
			Label2(n) = Str(i)
			n = n + 1
		Next i
		SetChannelCount Val(Text1.Text) + 3
	End If
End Sub

Private Sub Text1_Change()
	UpdateLabels
End Sub

Private Sub VScroll1_Change(Index As Integer)
```

```
        Label3(Index) = Str(255 - VScroll1(Index).Value)
        SetData Val(Label2(Index)), Val(Label3(Index))
End Sub

Private Sub VScroll1_Scroll(Index As Integer)
        Label3(Index) = Str(255 - VScroll1(Index).Value)
        SetData Val(Label2(Index)), Val(Label3(Index))
End Sub
```

d) Un exemple en Borland C++ Builder

```
//Listing K8062D.h
#ifdef __cplusplus
extern «C» { /* Assume C declarations for C++ */
#endif

#define FUNCTION __declspec(dllimport)

FUNCTION __stdcall StartDevice();
FUNCTION __stdcall SetData(long Channel, long Data);
FUNCTION __stdcall SetChannelCount(long Count);
FUNCTION __stdcall StopDevice();

#ifdef __cplusplus
}
#endif

//Listing K8062.cpp
//---------------------------------------------------------------------------

#include <vcl.h>
#pragma hdrstop
#include «K8062D.h»
#include «K8062.h»
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
int StartAddress = 1;
//---------------------------------------------------------------------------
__fastcall TForm1::TForm1(TComponent* Owner)
        : TForm(Owner)
{
}
//---------------------------------------------------------------------------
void __fastcall TForm1::FormCreate(TObject *Sender)
{
        StartDevice();
}
//---------------------------------------------------------------------------
void __fastcall TForm1::FormClose(TObject *Sender, TCloseAction &Action)
{
StopDevice();
}
//---------------------------------------------------------------------------
void __fastcall TForm1::Edit1Change(TObject *Sender)
{
        if (StrToInt(Edit1->Text)>0)
        {
                if (StrToInt(Edit1->Text)< 509)
                {
                SetChannelCount(StartAddress+ 3);
                StartAddress = StrToInt(Edit1->Text);
                Label5->Caption = IntToStr(StartAddress);
                Label6->Caption = IntToStr(StartAddress+ 1);
                Label7->Caption = IntToStr(StartAddress+ 2);
```

```
                Label8->Caption = IntToStr(StartAddress+ 3);
                }
        }
}
//---------------------------------------------------------------------------
void __fastcall TForm1::ScrollBar1Change(TObject *Sender)
{
        Label1->Caption = IntToStr(255-ScrollBar1->Position);
        SetData(StartAddress, 255-ScrollBar1->Position);
}
//---------------------------------------------------------------------------
void __fastcall TForm1::ScrollBar2Change(TObject *Sender)
{
        Label2->Caption = IntToStr(255-ScrollBar2->Position);
        SetData(StartAddress + 1, 255-ScrollBar2->Position);
}
//---------------------------------------------------------------------------
void __fastcall TForm1::ScrollBar3Change(TObject *Sender)
{
        Label3->Caption = IntToStr(255-ScrollBar3->Position);
        SetData(StartAddress + 2, 255-ScrollBar3->Position);
}
//---------------------------------------------------------------------------
void __fastcall TForm1::ScrollBar4Change(TObject *Sender)
{
        Label4->Caption = IntToStr(255-ScrollBar4->Position);
        SetData(StartAddress + 3, 255-ScrollBar4->Position);
}
//---------------------------------------------------------------------------
```

la communication.
Cette procédure doit être effectuée avant tout autre opération.
Syntaxe:
StartDevice
Exemple:
BEGIN
StartDevice; // Ouvre la connexion avec la platine
END;

StopDevice

Ferme la communication avec l'interface USB DMX. C'est la dernière instruction avant la fin de la liaison.
Syntaxe:
StopDevice
Exemple:
BEGIN
StopDevice; // Ferme la communication avec la platine
END

SetChannelCount

Cette fonction permet de définir entre 8 et 512 le nombre de canaux DMX contrôlés. Cela permet de réduire nettement les délais d'exécution quand le nombre de canaux est supérieur à 512.
Syntaxe:
SetChannelCount(Count: Longint)
Exemple:
BEGIN
SetChannelCount(32); // Paramètre le nombre maximum de canaux DMX à 32
END;

SetData

Cette fonction permet de définir le niveau de chaque canal DMX (de min 0 à max 255).
Syntaxe:
SetData(Channel: Longint; Data: Longint)
Exemple:
BEGIN
SetData(1,127); // Règle le niveau du canal 1 à la valeur 127 (environ 50% de la valeur max).
END;

Les figures présentent une série d'exemples en Delphi, Visual Basic, Visual Basic.NET et Borland C++Builder dans lesquels nous illustrons l'utilisation des instructions StartDevice et StopDevice ainsi que des procédures pour établir le nombre de canaux DMX et la valeur à donner à chacun.

Grâce à ces exemples, ceux qui veulent réaliser un logiciel personnalisé pourront commencer la rédaction du programme avec plus de facilité.

Un exemple, même simple, constitue toujours une bonne base de départ!

L'interface DMX-USB représente certainement un des montages les plus remarquables de ceux réalisés avec le protocole DMX512 ; toutefois nous vous en avons déjà proposé d'autres qui ont rencontré un vif succès, comme le mélangeur à 8 canaux pour régie de lumières et d'autres sont à paraître ou en cours de parution comme l'unité de puissance à 8 canaux elle aussi.

Sans parler des articles un peu plus anciens déjà consacrés à ce protocole comme l'unité de puissance DMX qui était proposée avec le programme résident du PIC complet (donné afin que vous puissiez sur ce modèle écrire tous les programmes dont vous aurez besoin pour monter vos propres régies DMX personnalisées).

Comment construire ce montage ?

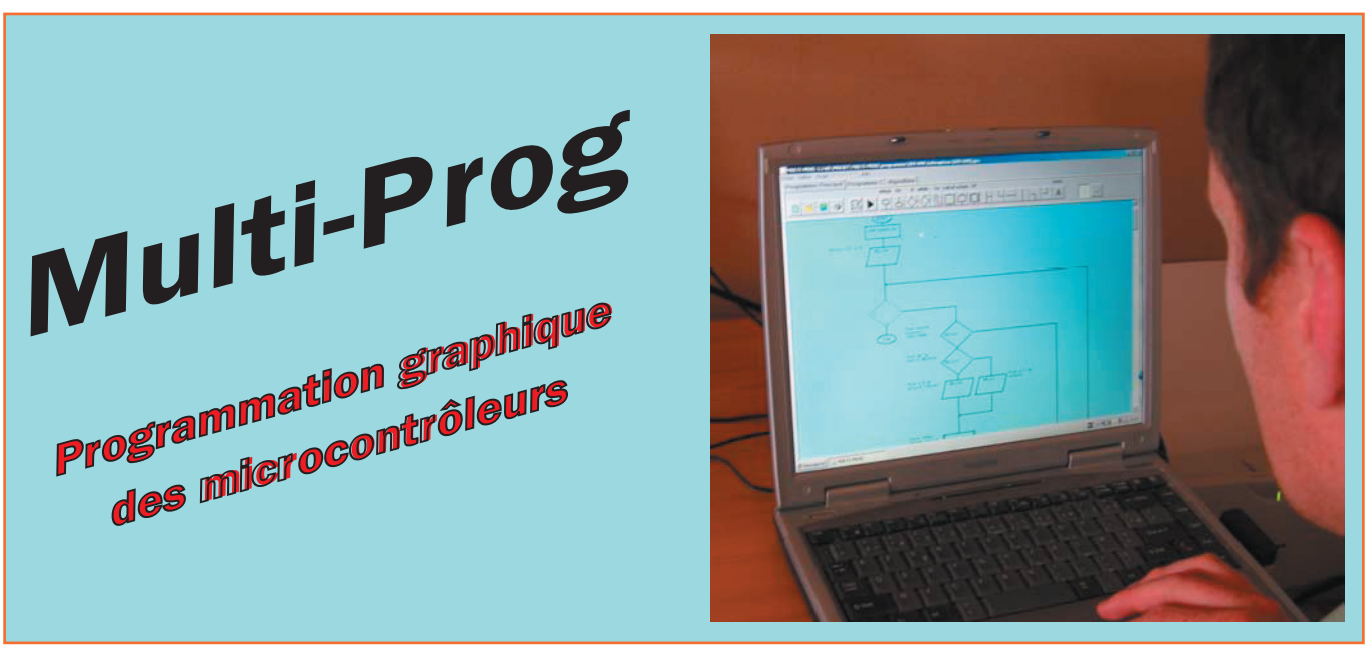
Tout le matériel nécessaire pour construire ce contrôleur DMX sur port USB EV8062 est disponible chez certains de nos annonceurs.

Voir les publicités dans la revue.

Les typons des circuits imprimés sont sur www.electronique-magazine.com/ci.asp.

Les programmes disponibles sont sur www.electronique-magazine.com/mc.asp. ◆

Programmation graphique des microcontrôleurs



Dans le monde des techniciens, le microcontrôleur a permis de simplifier la réalisation des cartes électronique car il n'y a qu'un composant maître sur la carte électronique et ce composant réalise toutes les fonctions (Commande de ports numériques ou analogiques, mesure de temps, variation de vitesse d'un moteur, dialogue avec un ordinateur....). Il est donc devenu très aisé de réaliser le routage d'une carte électronique.
La phase la plus difficile à aborder reste la programmation, car en effet, il faut connaître les registres du microcontrôleur, les paramétrer et écrire ces satanées lignes de code. Et bien NON! C'en est fini de ces centaines de lignes de codes, de ces heures passées à comprendre la signification des registres qui portent tous des noms de guerre!!!!! Sans oublier que lorsque vous connaissez parfaitement un microcontrôleur, le monde s'écroule autour de vous lorsque vous changez de modèle, ou pire encore de marque.

Un nouveau logiciel: MULTI-PROG

Le logiciel MULTI-PROG a été créé pour faciliter la prise en main des microcontrôleurs par des élèves et des débutants en programmation. En effet, ce logiciel permet d'utiliser toutes les fonctions liées autour du microcontrôleur sans écrire une seule ligne de code. Les élèves peuvent comprendre le rôle d'un CAN sans paramétrer un seul registre: ils dessinent l'algorigramme et obtiennent le programme directement.
Il n'y a plus à travailler les structures itératives ou alternatives,

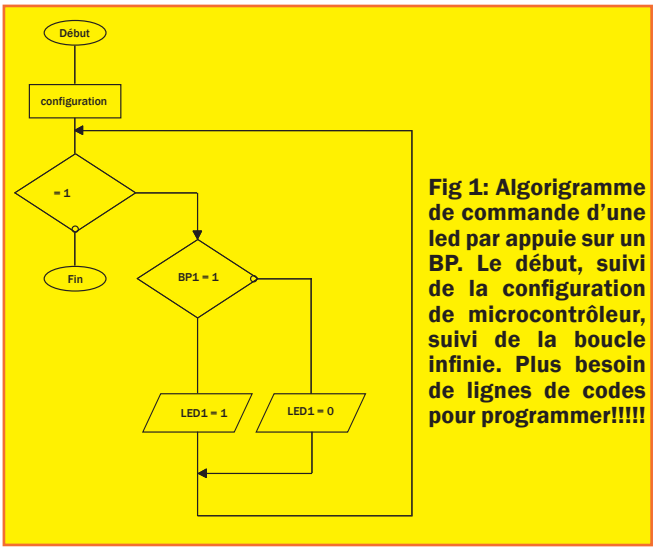


Fig 1: Algorigramme de commande d'une led par appuie sur un BP. Le début, suivi de la configuration de microcontrôleur, suivi de la boucle infinie. Plus besoin de lignes de codes pour programmer!!!!!

il n'y a qu'à dessiner l'algorigramme!!!!

Exemple de programme:

Regardez ci-avant la facilité pour allumer une led en appuyant sur un bouton poussoir. On reconnaît la zone de début (activation du microcontrôleur), la zone de configuration (configu-

ration du microcontrôleur: port d'entrée, de sortie, le port utilisé en CAN,....), le test qui toujours vrai (while (1)) pour signifier au microcontrôleur de faire toujours le rebouclage. Et dans la partie droite: notre programme. Ici, si on appuie sur le BP1, la led est allumée sinon, elle est éteinte. Vous trouvez ça difficile non? Mais ne jetez pas vos cours de programmation, ils peuvent toujours servir ..... pour combler votre bibliothèque, impressionner vos amis....mais, avec ce logiciel, c'est sûr, vous ne les ouvrirez plus.

Programmer des systèmes avec MULTI-PROG, c'est simple et passionnant!!!!!

Des fonctions variées:

Ce logiciel permet donc de programmer toutes les fonctions classiques des microcontrôleurs ATMEL et PIC, c'est-à-dire: CAN, Timer, RS232, afficheurs 4 bits, claviers, interruptions....De plus, l'auteur de ce logiciel est sympa et vous propose de lui envoyer des mails pour faire évoluer son logiciel afin de répondre à vos propres exigences. C'est sympa, non?

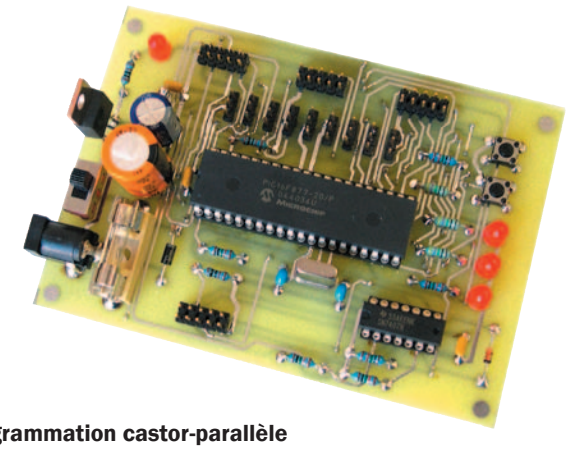
L'animation dynamique:

Une des dernières nouveautés qui va intéresser beaucoup de monde, c'est le mode mise au point: je m'explique mieux: Vous dessinez votre algorigramme vite fait et bien fait et après l'avoir transféré dans votre microcontrôleur, votre système ne marche toujours pas, et bien vous pouvez visualiser le fonctionnement de l'algorigramme en même temps qu'il s'exécute. Ce mode est très intéressant pour comprendre la structure d'un programme mais aussi, pour dépanner un système ou un programme.

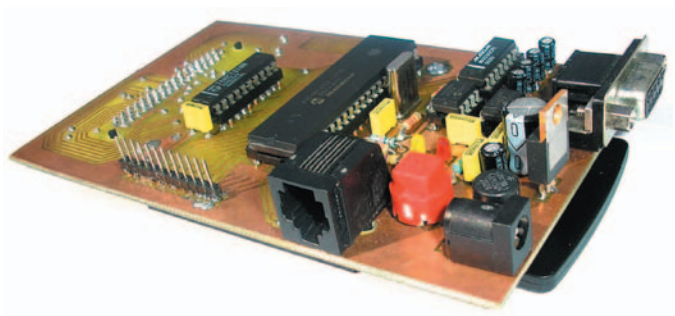
Bilan de MULTI-PROG:

En résumé, ce logiciel qui a été créé pour des étudiants afin de les aider à comprendre toutes les fonctionnalités d'un microcontrôleur et à les programmer avec le maximum d'autonomie, vous permettra de programmer ou de créer votre propre application sans passer des heures à écrire des lignes de codes et à vous former à la programmation.
Ce logiciel vous permet de programmer votre système intuitivement et rapidement et après, de voir votre algorigramme évoluer en même temps que votre application.

Si vous voulez tester ce logiciel, une version démo est téléchargeable sur le site www.micrelec.com (http://www.micrelec.fr/lyctech/electronique.htm) et pour joindre l'auteur de MULTI-PROG: http://perso.wanadoo.fr/MULTI-PROG



Carte de programmation castor-parallèle



Programmateur de microcontrôleurs Castor-Wifi

Les cartes CASTOR:

L'auteur de ce logiciel a également prévu des cartes de programmation complètement compatibles avec le logiciel MULTI-PROG (mais aussi avec les autres logiciels de programmation). En effet, ces cartes dédiées aux microcontrôleurs ATMEL ou PIC, permettent non seulement de programmer vos microcontrôleurs ATMEL ou PIC mais aussi de tester vos programmes grâce aux boutons poussoirs, aux leds et les connecteurs disponibles sur ces cartes. Avec ces cartes, outre une documentation complète vous avez des cours et des exemples fournis afin que vous soyez autonome le plus rapidement possible. L'objectif des cartes de programmation « Castor » aidées du logiciel MULTI-PROG est de vous permettre de réaliser vos projets le plus rapidement possible et le plus aisément possible, pour un budget le plus minime possible.

Fini la lecture de 2 ou 3 livres de 500 pages chacun et des heures passées pour réussir votre projet? Tout est prêt pour vous.

Programmez vos microcontrôleurs avec les cartes Castor-Parallèle, Castor-USB ou Castor-Wifi:

Il existe 3 types de cartes de programmation: programmation par câble parallèle (Castor-parallèle), programmation par câble USB (Castor-USB) et programmation par liaison WIFI (Castor-Wifi). Et oui, vous avez bien compris, vous pouvez programmer sans fils, il suffit d'avoir une carte Wifi sur votre ordinateur et d'avoir la carte Castor-Wifi et ainsi, vous pouvez programmer votre microcontrôleur à distance.

Pour plus amples informations: www.micrelec.com ou encore sur le site des auteurs ;
http://perso.wanadoo.fr/electrocastor

Régis Boullard,
Jean-Yves Laronde,
Olivier Snoeck

Note de la rédaction

L'article ci-avant a entièrement été rédigé par des auteurs externes à la revue. La rédaction d'Electronique et Loisirs Magazine décline toute responsabilité quant à la teneur et l'exactitude de l'article qui n'engagent que leurs auteurs.

SERILEC

E.MAIL : SERILEC2@WANADOO.FR

SPÉCIALISATION:
PROTOTYPE
PETITES ET MOYENNES SÉRIES

RÉALISATION:
CIRCUITS SIMPLE FACE
CIRCUITS DOUBLE FACE
CIRCUITS MULTICOUCHES DE 3 à 6

FINITION SELECTIVE
TEST À SONDE

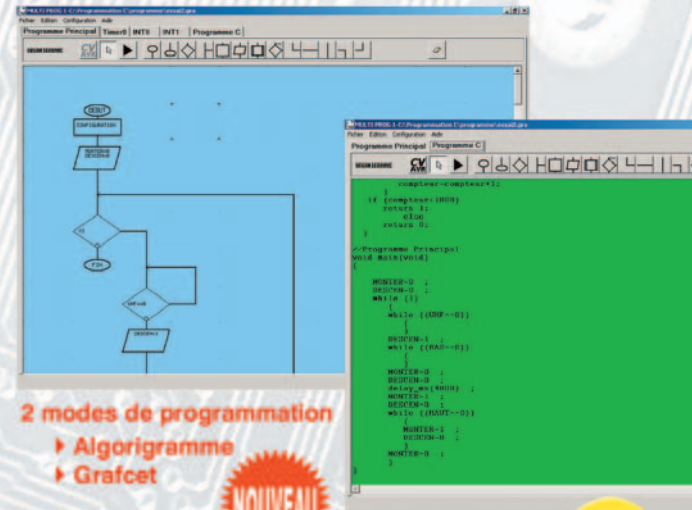
TÉL. : 04.42.24.39.52
FAX : 04.42.24.47.55

70,RUE LOUIS ARMAND BP 174000
13795 AIX EN PROVENCE CEDEX

Multi-PROG

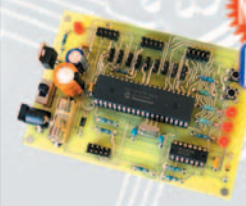
Programmation graphique !

des microcontrôleurs ATMEL & PIC



2 modes de programmation
- Algorigramme
- Grafcet

NOUVEAU
Pack Logiciel + Carte associée 150 €



Ensemble carte PIC (en kit) + log. MultiPROG PIC Réf. M14P610 150 €
PIC16F877
Ensemble carte ATMEL (en kit) + log. MultiPROG ATMEL Réf. M14P611 150 €
ATMEGA8535

démo téléchargeable sur : www.micrelec.fr rubrique S.T.I./Génie Électronique

MICRELEC 4, place Abel Leblanc - 77120 Coulommiers
tel : 01 64 65 04 50 - Fax : 01 64 03 41 47

BATTERIES AU PLOMB



AP6V1,2AH	6V1,2Ah	97x25x51 mm	4,60 €	AP12V1,3AH	12V1,3Ah	97x48x52 mm	8,60 €	AP12V12AH	12V12Ah	151x98x94 mm	27,00 €
AP6V3AH	6V3Ah	134x34x60 mm	10,80 €	AP12V2,2AH	12V2,2Ah	178x34x60 mm	10,60 €	AP12V18AH	12V18Ah	180x75x167 mm	29,00 €
AP6V4,5AH	6V4,5Ah	70x47x1011 mm	4,80 €	AP12V3AH	12V3Ah	34x67x60 mm	11,00 €	AP12V26AH	12V26Ah	175x166x125 mm	129 €
AP6V7AH	6V7Ah	34x151x98 mm	10,50 €	AP12V4,5AH	12V4,5Ah	90x70x101 mm	11,20 €	AP12V44AH	12V44Ah	197x165x174 mm	185 €
AP6V12AH	6V12Ah	151x50x94 mm	12,80 €	AP12V8AH	12V8Ah	151x65x94 mm	13,30 €	AP12V65AH	12V65Ah	350x166x175 mm	281 €

ALIMENTATIONS POUR PC PORTABLE





AP70C Alimentation universelle de voiture 70 W: entrée 12 à 15 V DC et sorties 15-16-18-19-22-24V DC.....30 €

AP120C Alimentation universelle de voiture 120 W: entrée 12 à 15 V DC et sorties 15-16-18-19-22-24V DC.....43 €

AP70 Alimentation universelle secteur 70 W: entrée 100 à 240 V AC et sorties 12-15-16-18-19-22-24V DC.....55 €

AP120 Alimentation universelle secteur 120 W: entrée 100 à 240 V AC et sorties 12-15-16-18-19-22-24V DC.....73 €

CONVERTISSEURS DE TENSION



Alarme batterie faible-Tension d'entrée : 10 - 15 volt DCTension de sortie : 220 volt ACFréquence : 50 HzRendement : 90 %Protection thermique : 60 °Ventilation forcée sur tous les modèles sauf G12-015

G12015 Convertisseur de 12 V - 220 V - 150 W - 162x104x58 mm - 0,700 kg.....58,60 €

G12030 Convertisseur de 12 V - 220 V - 300 W - 235x100x60 mm - 0,830 kg ..... 89,20 €

G12060 Convertisseur de 12 V - 220 V - 600 W - 290x205x73 mm - 2,100 kg .....125,00 €

COMELEC

CD 908 - 13720 BELCODENE
WWW.comelec.fr
Tél. : 04 42 70 63 90 - Fax : 04 42 70 63 95

Expéditions dans toute la France. Moins de 5 kg : Port 8,40 €. Règlement à la commande par chèque, mandat ou carte bancaire. Bons administratifs acceptés. Le port est en supplément. De nombreux kits sont disponibles, envoyez votre adresse et cinq timbres, nous vous ferons parvenir notre catalogue général.

HAMEXPO

27 ème Salon International Radioamateur

Techniques de radiocommunication et d'informatique

22-23 Octobre 2005



Réseau des Emetteurs Français - Union Française des Radioamateurs
REF-UNION 32, rue de Suède BP 77429 - 37074 TOURS cedex 2
Tél: 02 47 41 88 73 - www.ref-union.org

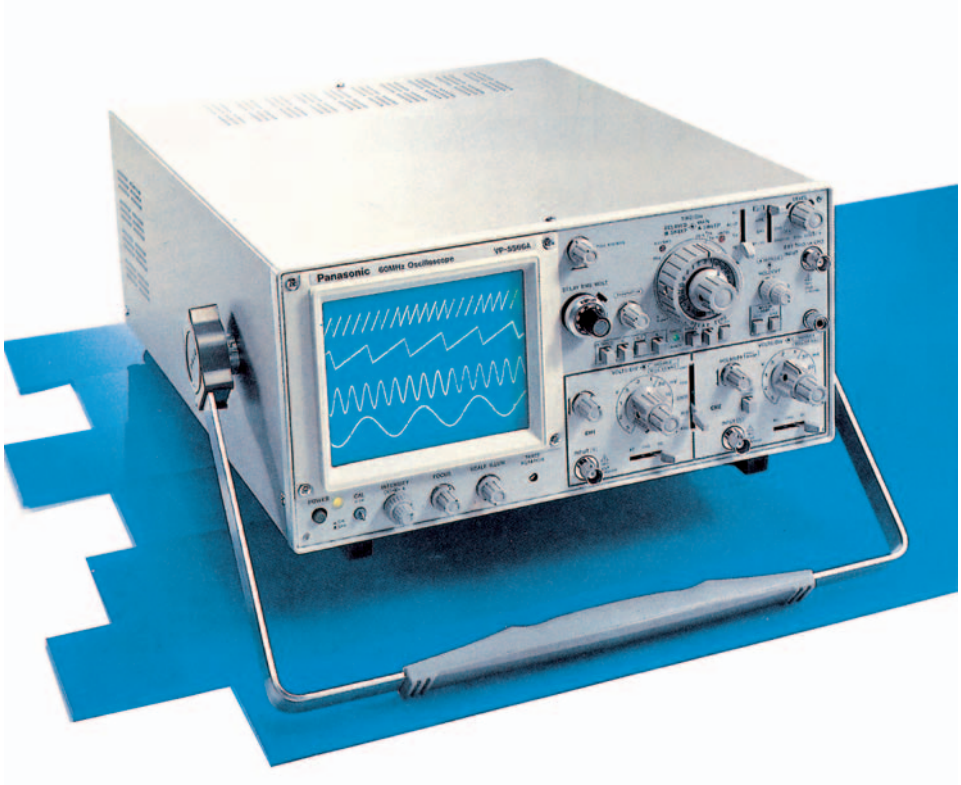
F6CPI

NIVEAU 3
LEÇON N° 47-3

Comment utiliser l'oscilloscope

Comment mesurer des tensions alternatives de 50 Hz avec l'oscilloscope

Étant donné que l'oscilloscope nous permet de visualiser à l'écran n'importe quelle forme d'onde, qu'elle soit sinusoïdale, triangulaire ou carrée, il est un excellent instrument pour la mesure des tensions alternatives. Cette Leçon vous apprendra comment mesurer l'amplitude d'une onde et comment convertir les Vpp en Veff.



Nous poursuivons nos leçons dédiées à l'oscilloscope avec, cette fois, la mesure des tensions alternatives à une fréquence de 50 Hz. Mais pourquoi se servir d'un oscilloscope alors qu'un simple multimètre réglé en position AC («Alternate Current») fait parfaitement l'affaire ? Eh bien parce que l'oscilloscope permet de préciser si la tension alternative mesurée est celle d'une onde sinusoïdale ou bien triangulaire ou encore carrée et si le signal est parfait ou bien déformé par des auto-oscillations ou des distorsions. De plus, l'oscilloscope nous permet de connaître la valeur de la tension continue que nous obtiendrions si nous redressions ce signal et sa fréquence exacte.

La mesure des tensions alternatives à 50 Hz

Avant d'effectuer une mesure de tension alternative à 50 Hz, vous devez régler les commandes de l'oscilloscope comme suit :

- Trigger MODE (voir figure 1) = sous cette indication pressez la touche Auto.

- Trigger SOURCE (voir figure 2) = à cette indication correspond un sélecteur que vous devez positionner sur Norm (Normal). Sur certains oscilloscopes on trouve Int (Internal).

- Time/div (voir figure 3) = quand on mesure une tension alternative de 50 Hz, il faut positionner ce bouton sur la portée 10 ms et ainsi cinq sinusoïdes complètes sont visualisées à l'écran.
  Si la fréquence du signal alternatif à mesurer était de 1 kHz, pour visualiser cinq sinusoïdes complètes, vous devriez placer ce bouton sur la portée 0,5 ms.

- Vertical MODE (voir figure 4) = près de cette indication se trouvent des poussoirs (presque toujours en ligne verticale) servant à sélectionner l'entrée de l'oscilloscope que vous désirez utiliser. Comme on utilise normalement l'entrée CH1, pressez le poussoir CH1.

- Sélecteur AC-GND-DC (voir figure 5) = ce sélecteur du canal CH1 est positionné normalement sur GND de façon à court-circuiter l'entrée.

- Bouton déplacement vertical = ce petit bouton sert à placer la trace horizontale au centre de l'écran, comme le montre la figure 5.

- Sonde oscilloscope = il est conseillé de positionner l'inverseur de la sonde sur x10, comme le montre la figure 6.

- Sélecteur V/div de CH1 = si vous connaissez à peu près la valeur de la tension que vous vous apprêtez à mesurer, vous pouvez placer le bouton des V/div sur la portée adéquate, sans oublier que, si la trace horizontale est placée au centre de l'écran, vous aurez un total de huit carreaux disponibles, quatre pour les demi ondes positives et quatre pour les négatives. Si vous ne connaissez pas cette valeur, mettez le sélecteur V/div sur la portée maximale, soit 5 V/div, comme le montre la figure 7.

Important : avant de positionner ce bouton sur V/div, souvenez-vous que les V d'une tension alternative ne correspondent pas à ceux que vous liriez sur un multimètre quelconque, car le multimètre lit les Veff (tension efficace) et l'oscilloscope les Vpp (tension crête-crête), comme le montrent les figures 16, 17 et 18.

Par conséquent, si vous lisez 12 V sur un multimètre (voir figure 8), sur l'écran de l'oscilloscope vous verrez les sinusoïdes atteindre une amplitude de 33,9 Vpp, en effet :

12 x 2,828 = 33,9 V.

Si le multimètre indique 30 V alternatifs, l'écran visualisera des sinusoïdes de :

30 x 2,828 = 84,8 V.

Nous vous expliquerons plus bas quelle est la différence entre les deux valeurs de tension Veff et Vpp.

Un exemple de mesure AC

Après avoir mis le sélecteur AC-GND-DC sur GND (voir figure 5) et agit sur le petit bouton de déplacement de la courbe verticalement de façon à la placer au centre de l'écran, avant d'exécuter une quelconque mesure, n'oubliez pas de faire passer le sélecteur de GND à AC (voir figure 7).

Si vous avez un transformateur secteur 230 V/12 V sous la main, contrôlez la tension du secondaire avec un multimètre et avec l'oscilloscope.

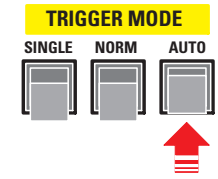


Figure 1 : Pour visualiser une trace parfaitement stable, pressez le bouton AUTO du Trigger Mode.

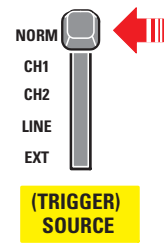


Figure 2 : Pour synchroniser le signal, mettez la touche du Trigger Source en position NORM.

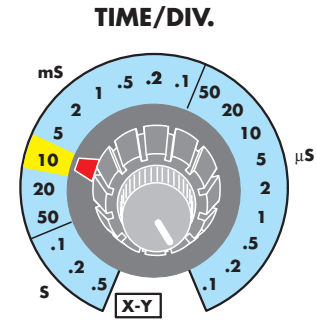


Figure 3 : Pour mesurer une tension AC de 50 Hz, mettez le sélecteur Time/div sur la position 10 ms.

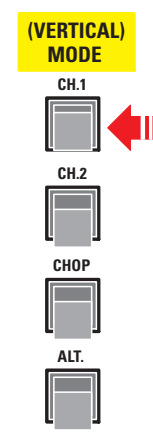


Figure 4 : Dans le Vertical Mode, pressez le poussoir du canal que vous voulez visualiser (c'est-à-dire l'entrée CH1 ou bien CH2).

Important : quand vous reliez les fils du secteur 230 V au primaire de ce transformateur, pensez à les manipuler avec des pinces et un tournevis isolés, à ne pas y mettre les doigts et à les isoler (gaine thermorétractable ou ruban plastique), car le contact du corps humain avec cette tension peut être mortel !

Si vous possédez un transformateur dont les primaire et secondaire ne sont pas identifiés, repérez l'enroulement utilisant le fil le plus fin : c'est le primaire (le fil de l'enroulement secondaire est bien plus gros). Après avoir mis le primaire sous tension, lisez avec un multimètre à aiguille ou

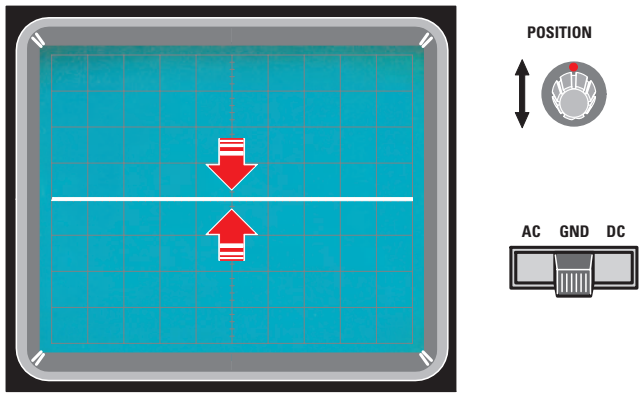


Figure 5 : Pour placer la trace au centre de l'écran, mettez le sélecteur AC-GND-DC en position GND puis tournez le petit bouton Vertical Position pour le régler sur AC (voir figure 7).

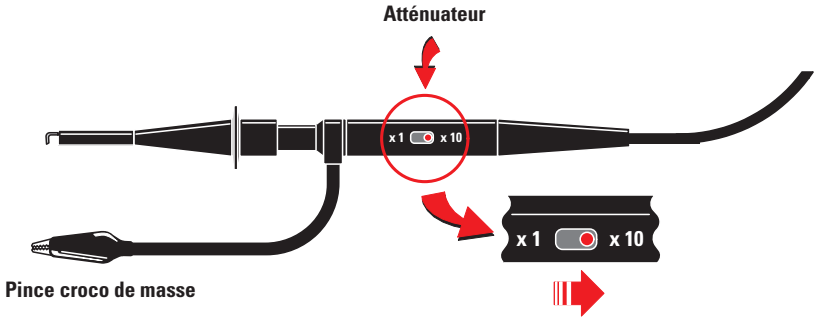


Figure 6 : Pour mesurer des tensions alternatives supérieures à 40 Vpp, il faut placer l'inverseur de la sonde sur la position x10.

numérique la tension du secondaire 12 V, comme le montre la figure 8. Pour exécuter cette mesure à l'oscilloscope, vous devez procéder comme le montre la figure 9.

Sélecteur V/div du canal CH1 sur 5 V/div, inverseur de la sonde sur x10 (voir figure 6) : chaque carreau vertical correspond à une tension de 5 x 10 = 50 V. La tension alternative de 12 Veff correspondant à une amplitude de 33,9 Vpp (voir figure 16), les sinusoïdes visualisées à l'écran n'occuperont qu'un demi carreau. Pour les développer, adoptez l'une des solutions suivantes :

- mettez l'inverseur de la sonde sur x1 et maintenez le sélecteur V/div du canal CH1 sur 5 V/div.

- maintenez l'inverseur de la sonde sur x10 et réglez le sélecteur V/div du canal CH1 sur 0,5 V/div.

La première solution vous simplifiera tous les calculs car, en fonction du nombre de carreaux et de parties, vous pourrez tout de suite connaître la tension Vpp.

Si vous mesurez la tension alternative de 12 Veff, soit 33,9 Vpp, l'écran visualisera des sinusoïdes dépassant légèrement six carreaux, comme le montre la figure 12.

Si vous déplacez légèrement ces sinusoïdes sur la première ligne du bas à l'aide du bouton de Position (voir figure 13), vous verrez leur extrémité supérieure dépasser le 6ème carreau de quatre parties. Le sélecteur de l'entrée CH1 étant sur 5 V/div, cela nous fait une tension de :

6 x 5 = 30 V.

Reste les quatre parties. Vous savez qu'en fonction du réglage du bouton des V/div chaque partie a une valeur déterminée que donne le Tableau 1. Sur la portée 5 V, la 4ème partie correspond à 4 V, ce qui fait en additionnant

30 + 4 = 34 V

au lieu des 33,9 V attendus (la petite différence constitue une tolérance vraiment inférieure à celle d'un multimètre à aiguille).

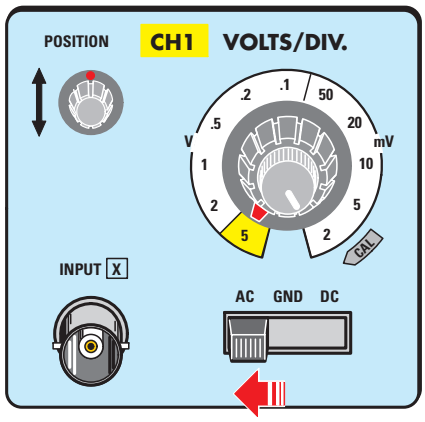


Figure 7 : Si vous connaissez approximativement la valeur de la tension alternative à mesurer, vous pouvez tout de suite choisir la portée adéquate avec le sélecteur V/div. Les nombres marqués autour du sélecteur (5-2-1-0,5 etc.) indiquent les V/carreau quand l'inverseur de la sonde est sur x1. Donc, sur la portée 5 V (la sonde étant sur x1), vous pouvez mesurer une valeur maximale de tension de 40 Vpp ; sur x10 une valeur maximale de 400 Vpp.

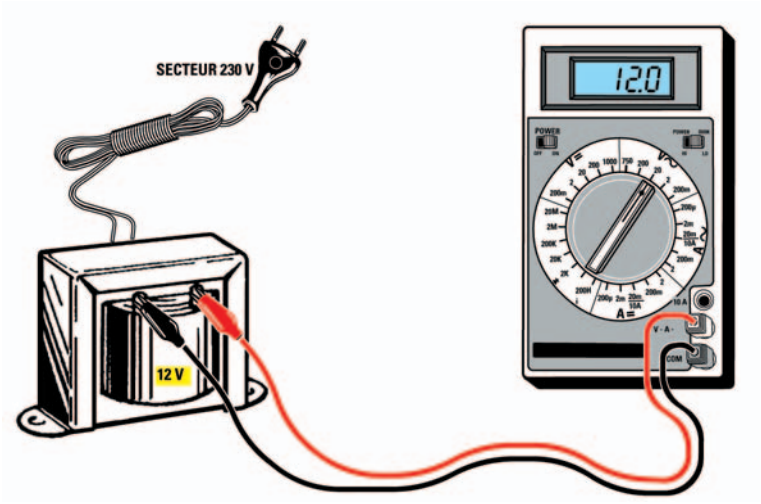


Figure 8 : Si vous mesurez la tension issue du secondaire d'un transformateur secteur 230 V/12 V avec un multimètre, il affichera 12 V puisqu'il lit les Veff (tension efficace).

Tension efficace Veff et tension crête-crête Vpp

Quand on utilise l'oscilloscope comme voltmètre pour mesurer des tensions alternatives, il faut savoir qu'il visualise à l'écran des ondes sinusoïdales, triangulaires ou carrées complètes.

En lisant le nombre de carreaux compris entre le maximum (pic ou crête) positif et le maximum (pic ou crête) négatif (voir figure 14), on obtient une valeur de tension de crête à crête (ou de pic à pic) Vpp nettement supérieure à la valeur de la tension efficace Veff que vous pouvez lire sur un multimètre analogique ou numérique (les Veff sont exprimés également par l'expression anglaise VRMS, pour «Root Mean Square»).

Afin de vous faire bien comprendre la différence entre tension crête-crête Vpp et tension efficace Veff, nous vous proposons une analogie, qu'illustre la figure 15 : la tension Vpp est comme deux cônes de glace (un positif, l'autre négatif) mis l'un sur l'autre en sens inverse (placés dans un récipient) et qui atteignent ainsi une certaine hauteur ; la tension Veff est comme la quantité d'eau que l'on obtient quand la glace des deux cônes a fondu (bien sûr la hauteur d'eau est inférieure à la hauteur de glace).

Eh bien, en terme l'électricité, la hauteur Veff est inférieure à la hauteur Vpp de 2,828 fois si le signal est une onde sinusoïdale. Comme le montre la figure 14, la sinusoïde occupe une amplitude de six carreaux à l'écran et la portée sélectionnée étant de 5 V/div (voir figure 7), pour connaître la tension lue sur un multimètre Veff, il faut diviser Vpp par ce nombre :

(6 x 5) : 2,828 = 10,6 volts efficaces.

A l'inverse, si l'on veut connaître la Vpp d'une tension alternative quand le multimètre utilisé affiche 45 Veff, il faut multiplier par ce nombre :

45 x 2,828 = 127,26 volts pic-pic.

Comme la tension maximale que l'on peut visualiser à l'écran est de 40 V, quand on est sur la portée 5 V/div et

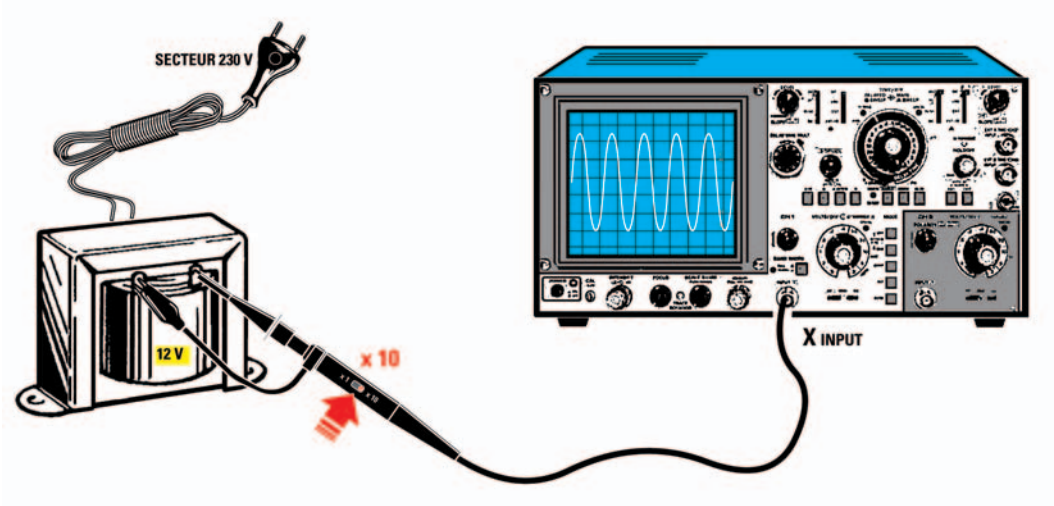


Figure 9 : Si vous mesurez cette tension avec un oscilloscope, vous verrez que l'amplitude du signal atteint une valeur de 33,9 V car l'oscilloscope visualise la valeur des Vpp (tension de crête à crête, ou «peak», pic) que l'on obtient avec la formule Vpp = Veff x 2,828 (voir figure 16). Afin d'éviter des erreurs de lecture, contrôlez la position de l'inverseur de la sonde (x1 ou x10).

TABLEAU 1

Volts/div.	1°	2°	3°	4°	5° partie
2 mV	0,4 mV	0,8 mV	1,2 mV	1,6 mV	2,0 mV
5 mV	1,0 mV	2,0 mV	3,0 mV	4,0 mV	5,0 mV
10 mV	2,0 mV	4,0 mV	6,0 mV	8,0 mV	10 mV
20 mV	4,0 mV	8,0 mV	12 mV	16 mV	20 mV
50 mV	10 mV	20 mV	30 mV	40 mV	50 mV
0,1 V	0,02 V	0,04 V	0,06 V	0,08 V	0,1 V
0,2 V	0,04 V	0,08 V	0,12 V	0,16 V	0,2 V
0,5 V	0,1 V	0,2 V	0,3 V	0,4 V	0,5 V
1 V	0,2 V	0,4 V	0,6 V	0,8 V	1,0 V
2 V	0,4 V	0,8 V	1,2 V	1,6 V	2,0 V
5 V	1,0 V	2,0 V	3,0 V	4,0 V	5,0 V

Figure 10 : Le Tableau 1 donne la valeur des petites parties divisant chaque carreau (voir figure 11). Les valeurs correspondant à chaque partie, du 1er au 5ème, sont données en mV en fonction de la portée en V/div.

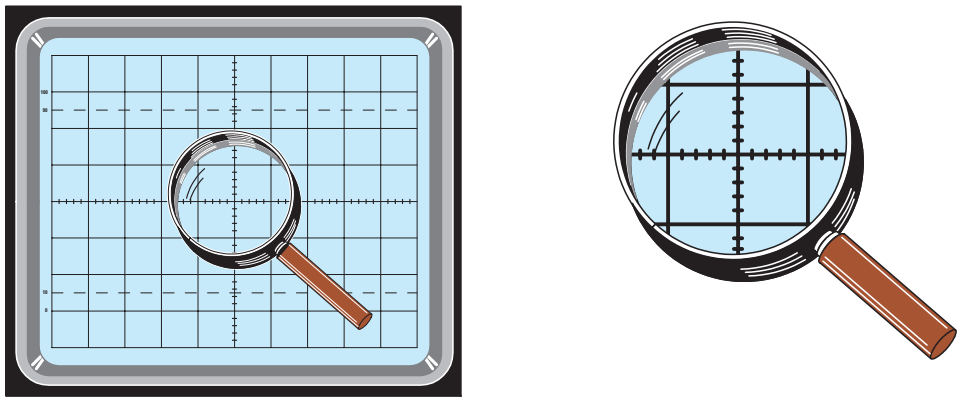


Figure 11 : La croix graduée au centre de l'écran permet d'évaluer les décimales d'une tension. En effet, chaque carreau vertical est divisé en cinq parties (4 petits traits plus les deux limites du carreau). En fonction du réglage du bouton V/div chaque partie a une valeur de tension indiquée par le Tableau 1.

que la sonde est sur x1 (8 carreaux verticaux, 5 x 8 = 40), pour visualiser un signal alternatif de 127 Vpp d'amplitude, vous devez mettre l'inverseur de la sonde sur x10, obtenant ainsi la possibilité de visualiser une tension d'une amplitude maximale de

5 x 8 x 10 = 400 Vpp.

Important : pour des raisons de sécurité, nous déconseillons d'effectuer des mesures de tension secteur 230 V à l'oscilloscope ; en effet, les 230 Veff correspondent à :

230 x 2,828 = 650,44 Vpp.

Si le boîtier métallique de l'oscilloscope se trouvait relié au fil de phase des 650 V en question, vous courriez un grave danger d'électrocution.

Vpp signal sinusoïdal

Pour effectuer la conversion des Veff aux Vpp et inversement quand il s'agit d'un signal sinusoïdal (voir figure 16 Tableau 2), on utilise le nombre 2,828 : mais où a-t-on trouvé ce nombre ? Oh, c'est tout simple : c'est la racine carrée de 8 ! En effet,

$$\sqrt{8} = 2{,}828.$$

En fait, dans les calculs on utilise seulement les deux premières décimales, donc 2,82.

Quand un signal sinusoïdal est redressé, pour trouver la tension continue il faut connaître la tension de crête ; aussi, pour convertir la tension efficace en tension de crête et inversement, on utilise le nombre 1,414 qui n'est autre que la racine carrée de 2 ! En effet,

$$\sqrt{2} = 1{,}414.$$

Là encore on n'utilise que les deux première décimales de ce nombre, donc 1,41. Exemple : si l'écran visualise des ondes sinusoïdales atteignant une amplitude de 20 Vpp, pour savoir quelle tension mesurerait un multimètre, on doit faire le calcul suivant :

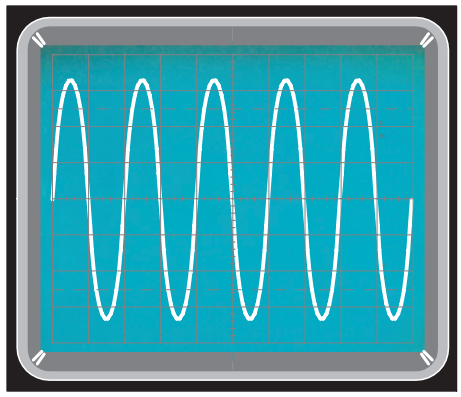
20 : 2,82 = 7 volts efficaces lus sur le multimètre.

Vpp signal triangulaire

Si la forme de l'onde n'est plus sinusoïdale mais triangulaire (voir figure 17 Tableau 2), c'est le nombre 3,464 que l'on utilisera pour les conversions Veff vers Vpp et inversement. Il s'agit de la racine carrée de 12 ! En effet,

$$\sqrt{12} = 3{,}464.$$

Figure 12 : Si vous mesurez une tension alternative de 12 Veff avec le sélecteur des V/div sur 5 V, vous obtenez une sinusoïde dont l'amplitude dépasse légèrement les six carreaux. Notez les extrémités supérieure et inférieure.



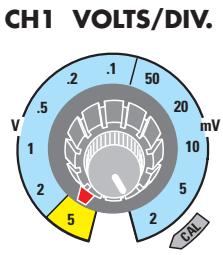
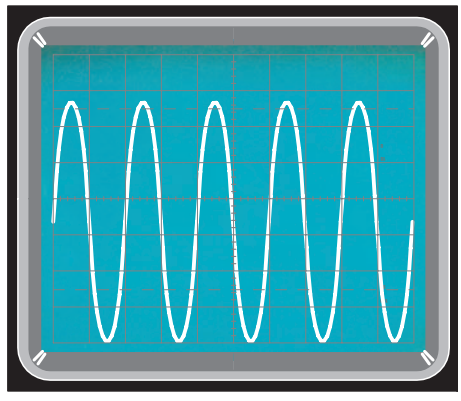


Figure 13 : Pour connaître la valeur exacte en V de cette sinusoïde, tournez le bouton Vertical Position de manière à placer l'extrémité inférieure de la sinusoïde sur la première ligne en bas. Vous verrez ainsi que la partie supérieure dépasse les six carreaux de quatre parties et, si vous consultez le Tableau 1, vous verrez que ces parties correspondent à 4 V : la sinusoïde a donc une valeur de (6 x 5) + 4 = 34 V.



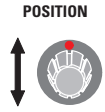
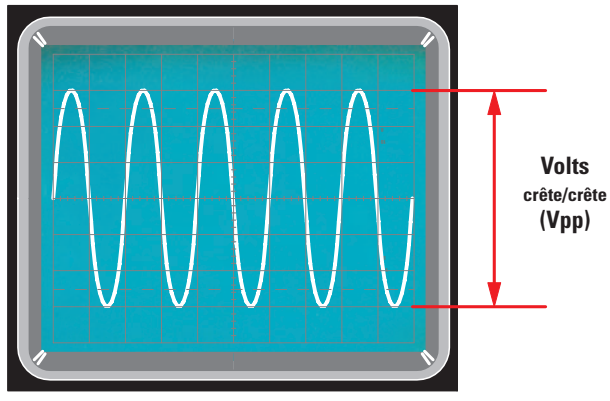


Figure 14 : Un oscilloscope mesure toujours les Vpp (tension crête-crête) d'un signal alternatif et donc, pour trouver la valeur efficace (Veff), vous devez appliquer la formule Veff = Vpp : 2,828 si l'onde est sinusoïde et Veff = Vpp : 3,464 si elle est carrée (voir figures 16, 17 et 18).



En fait, là encore dans les calculs on utilise seulement les deux premières décimales, donc 3,46. Quand un signal triangulaire est redressé, pour trouver la tension continue il faut connaître la tension de crête ; aussi, pour convertir la tension efficace en tension de crête et inversement, on utilise le nombre 1,732 qui n'est autre que la racine carrée de 3 ! En effet,

$$\sqrt{3} = 1{,}732.$$

Exemple : si l'écran visualise des ondes triangulaires atteignant une amplitude de 20 Vpp, pour savoir quelle tension mesurerait un multimètre, on doit faire le calcul suivant :

20 : 3,46 = 5,77 volts efficaces lus sur le multimètre.

ABONNEZ-VOUS A
ELECTRONIQUE
ET LOISIRS magazine
LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS

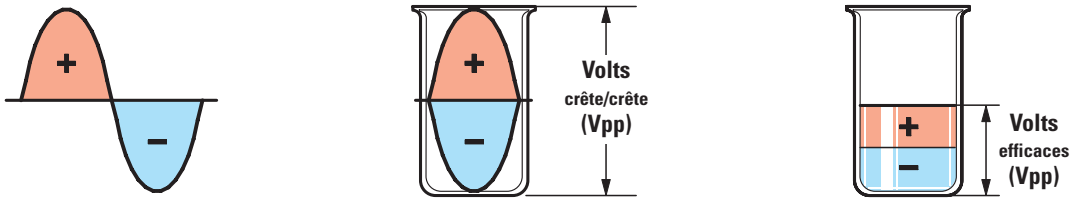
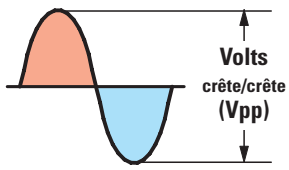


Figure 15 : Les Vpp peuvent être comparés à deux cônes de glice placés l'un sur l'autre en sens inverses. Les Veff sont comme le niveau d'eau obtenu après leur fusion (ce niveau est bien entendu inférieur).

Tableau 2 : conversions

ONDES SINUSOÏDALES

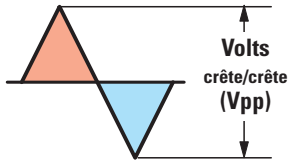


Vpp : 2,828 = Veff
Vp : 1,414 = Veff (Vp = volts crête, une seule alternance)

Veff x 2,828 = Vpp
Veff x 1,414 = Vp

Figure 16 : Les Veff d'une onde sinusoïdale sont obtenus en divisant les Vpp : 2,828.

ONDES TRIANGULAIRES

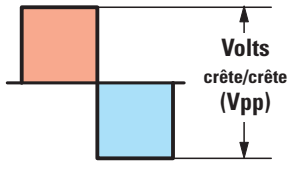


Vpp : 3,464= Veff
Vp : 1,732 = Veff (Vp = volts crête, une seule alternance)

Veff x 3,464 = Vpp
Veff x 1,732 = Vp

Figure 17 : Les Veff d'une onde triangulaire sont obtenus en divisant les Vpp : 3,464.

ONDES CARREES



Vpp : 2 = Veff
Vp : 1 = Veff (Vp = volts crête, une seule alternance)

Veff x 2 = Vpp
Veff x 1 = Vp

Figure 18 : Les Veff d'une onde carrée sont obtenus en divisant les Vpp : 2.

Vpp signal carré

Si la forme de l'onde n'est plus sinusoïdale ou triangulaire mais carrée (voir figure 18 Tableau 2), c'est le nombre 2 que l'on utilisera pour les conversions Veff vers Vpp et inversement.

Quand un signal carré est redressé, pour trouver la tension continue il faut connaître la tension de crête ; aussi, pour convertir la tension efficace en tension de crête et inversement, on utilise le nombre 1. Exemple : si l'écran visualise des ondes carrées atteignant une amplitude de 20 Vpp, pour savoir quelle tension mesurerait un multimètre, on doit faire le calcul suivant :

20 : 2 = 10 volts efficaces lus sur le multimètre.

Comme vous venez de le voir, il est possible avec un oscilloscope de trouver la tension Veff de n'importe quelle forme d'onde et c'est très facile en partant de la tension Vpp visualisée à l'écran.

A suivre

Dessinateur en CAO électronique sous PADS 2004 recherche travail en région parisienne ou région centre. Tél. : 01.64.56.95.13 ou 06.15.88.20.73.

Cause 80 ans cesse activité, vends oscilloscope Hameg HM303-6 2X30 meg. avec schémas 550 € + port, (sous garantie). Selfmètre et capacimètre les deux dans pe même coffret avec galva 100 ka et alimentation 52 € + port. Générateur de fonction 2 hertz à 5 meg. Sinus – carre – triangle – sweep 60 € + port tél. : 03.44.50.48.23.

Info : vous recherchez un appareils de mesure ? Allez voir sur le site www.catamesure.com

LE CATALOGUE DE LA MESURE

Vends oscilloscope Schlumberger type 5220 3 x 100 m voltmètre digital mémoire 2 BDT + retard numérique notice faire offre M. Villette tél. : 04.94.57.96.90

Vends une insoleuse 2 tubes 15 W neuf 100 € Graveuse avec pompe + chauffage, contenance 3 litres de perchlorure 150 €. Alarme sans fil valeur 700 € vendu 300 €. Adresse : Lang Marcel 4 rue des aloès 34540 Balaruc les Bains.

Vends étude et matériel pour réalisation générateur BF universel à 2 voies de sortie, à haut niveau, toutes fonctions réglable en fréquence, phrase forme et niveaux avec 5 afficheurs numériques. Coffrets pro. 3u, tous PCB, câblés ou vierges, composants et abondante doc. Inclus description sur demande motivée. Tél. : 02.31.92.14.80

Débutant en électronique recherche oscilloscope avec un prix raisonnable tél. : 06.66.57.21.20.

INDEX DES ANNONCEURS

ELC – Alimentation	2
COMELEC – Kits du mois	4
SELECTRONIC - Catalogue 2006	8
JMJ – Anciens numéros ELM	22
JMJ – CD-Roms anciens numéros ELM	23
COMELEC – Santé	32
VELLEMAN – Kits	43
PCB POOL – Réalisation de prototypes	51
REF UNION - Salon HAMEXPO	51
COMELEC – Transmission 2.4 GHz et 1.2 GHz	54
SERILEC – Circuits imprimés	67
MICRELEC – Chaîne complète CAO	67
COMELEC – Energie	67
REF UNION - Salon HAMEXPO	68
ARQUIÉ COMPOSANTS – Composants et mat.	77
MULTIPOWER – Autoformation et CAO	77
OPTIMINFO –Liaison Ethernet ou USB	77
COMELEC – Circuit imprimé PNP5	77
JMJ – Bulletin d'abonnement à ELM	78
GRIFO – Contrôle automatisation industrielle	79
ECE/IBC – Matériels et composants	80

ANNONCEZ-VOUS !

VOTRE ANNONCE POUR SEULEMENT 2 TIMBRES* À 0,53 € !

LIGNES	TEXTE : 30 CARACTÈRES PAR LIGNE. VEUILLEZ RÉDIGER VOTRE PA EN MAJUSCULES. LAISSEZ UN BLANC ENTRE LES MOTS.
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

*Particuliers : 2 timbres à 0,53 € - Professionnels : La grille : 90,00 € TTC - PA avec photo : + 30,00 € - PA encadrée : + 8,00 €

Nom ..... **Prénom** .....

Adresse .....

Code postal ..... **Ville** .....

Toute annonce professionnelle doit être accompagnée de son règlement libellé à l'ordre de JMJ éditions. Envoyez la grille, avant le 10 précédent le mois de parution, accompagnée de votre règlement à l'adresse:

JMJ/ELECTRONIQUE • Service PA • BP 20025 • 13720 LA BOUILLADISSE

Directeur de Publication
Rédacteur en chef
J-M MOSCATI
redaction@electronique-magazine.com

Direction - Administration
JMJ éditions
B.P. 20025
13720 LA BOUILLADISSE
Tél.: 0820 820 534
Fax: 0820 820 722

Secrétariat - Abonnements
Petites-annonces - Ventes
A la revue

Vente au numéro
A la revue

Publicité
A la revue

Maquette - Illustration
Composition - Photogravure
JMJ éditions sarl

Impression
SAJIC VIEIRA - Angoulême
Imprimé en France / Printed in France

Distribution
NMPP

Hot Line Technique
0820 000 787*
du lundi au vendredi de 16 h à 18 h

Web
www.electronique-magazine.com

e-mail
info@electronique-magazine.com

* N° INDIGO : 0,12 € / MN

ELECTRONIQUE ET LOISIRS magazine
LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS

EST RÉALISÉ
EN COLLABORATION AVEC :

NUOVA ELETTRONICA
Elettronica In

JMJ éditions
Sarl au capital social de 7800 €
RCS MARSEILLE : 421 860 925
APE 221E
Commission paritaire: 1000T79056
ISSN: 1295-9693
Dépôt légal à parution

I M P O R T A N T

Reproduction, totale ou partielle, par tous moyens et sur tous supports, y compris l'internet, interdite sans accord écrit de l'Editeur. Toute utilisation des articles de ce magazine à des fins de notice ou à des fins commerciales est soumise à autorisation écrite de l'Editeur. Toute utilisation non autorisée fera l'objet de poursuites. Les opinions exprimées ainsi que les articles n'engagent que la responsabilité de leurs auteurs et ne reflètent pas obligatoirement l'opinion de la rédaction. L'Editeur décline toute responsabilité quant à la teneur des annonces de publicités insérées dans le magazine et des transactions qui en découlent. L'Editeur se réserve le droit de refuser les annonces et publicités sans avoir à justifier ce refus. Les noms, prénoms et adresses de nos abonnés ne sont communiqués qu'aux services internes de la société, ainsi qu'aux organismes liés contractuellement pour le routage. Les informations peuvent faire l'objet d'un droit d'accès et de rectification dans le cadre légal.

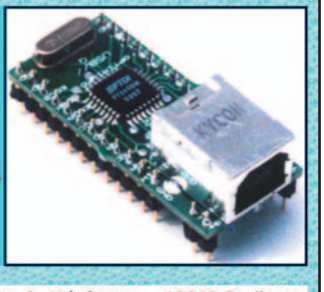
KIT COMMUNICATION

Intégrer une liaison Ethernet ou USB en quelques minutes.

* Convertisseur Ethernet TTL Série, RS232, RS485, RS422.
* Ethernet 10BaseT avec protocole TCP,UDP,ICMP (ping), ARP.
* Aucun composant extérieur
* Communication via ports virtuels ou TCP.
* Exemples en VB, Delphi fournis.
* Modèles disponibles avec protocole HTTP 1.0 et 8 entrées analogiques, programmation JAVA.
* **A partir de 66 € HT.**



* Composant USB 2.0 vers données séries ou parallèles.
* Drivers port virtuel pour Windows, Linux, MAC, ou DLL pour Windows, Linux, MAC gratuits.
* Exemples en C++, VB, Delphi fournis.
* Modèles avec micro PIC, SCENIX ou I/O24
* **Kit de développement à 30.90 € HT.**
* **Support technique gratuit**



optiminfo
Route de Ménétreau · 18240 Boulleret
Tél: 0820 900 021 · Fax: 0820 900 126
Site Web: www.optiminfo.com

Multipower

Des produits innovants pour vos projets

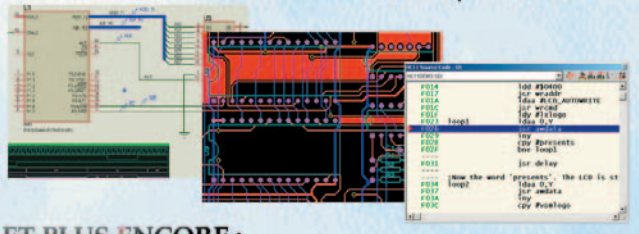
Ressources pédagogiques

Autoformation et cartes pour microprocesseurs PIC



CAO électronique PROTEUS

Schéma, PCB, simulation SPICE et processeurs



ET PLUS ENCORE :

Scopes, analyseurs USB

Cartes pour applications enfouies

Modules d'acquisition de données

PIC, AVR, HC11 et ARM7

NOUVEAU

Plus d'informations à www.multipower.fr

Tél : 01 53 94 79 90 & Fax : 01 53 94 08 51

arquié composants

Rue de écoles 82600 Saint-Sardos France
Tél. 05 63 64 46 91 Fax 05 63 64 38 39
SUR INTERNET http://www.arquie.fr/
e-mail : arquie-composants@wanadoo.fr

Catalogue N°61

Afficheurs.
Alimentations.
Caméras. Capteurs.
Cartes à puces.
Circuits imprimés.
Circuits intégrés.
Coffrets. Condensateurs.
Cellules solaires
Connectique.
Diodes. Fers à souder.
Interrupteurs.
Kits. LEDs.
Microcontrôleurs.
Multimètres.
Oscilloscopes. Outillage.
Programmateurs.
Quartz. Relais.
Résistances. Transformateurs.
Transistors. Etc...



BON pour CATALOGUE FRANCE: GRATUIT (3.00 €,pour: DOM, TOM, UE et autres pays)

Nom:.....Prénom:.....
Adresse:.....
.....
Code Postal:..... Ville:.....
M

COMMENT FABRIQUER FACILEMENT VOS CIRCUITS IMPRIMES ?

Nouveau produit qui arrive tout droit des États-Unis et qui a révolutionné les méthodes de préparation des circuits imprimés réalisés en petites séries :

plus de sérigraphie grâce à une pellicule sur laquelle il suffit de photocopier ou d'imprimer le master...



ET-PNP5
Lot de 5 feuilles au format A4
18,75€
+ port 8,00€

12/2003

COMELEC • CD908 • 13720 BELCODENE • Tél. : 04 42 70 63 90 Fax : 04 42 70 63 95



ABONNEZ VOUS à ELECTRONIQUE

ET LOISIRS magazine

LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS











et profitez de vos privilèges !

RECEVOIR votre revue directement dans votre boîte aux lettres près d'une semaine avant sa sortie en kiosques

BÉNÉFICIER de 50% de remise sur les CD-Rom des anciens numéros**

voir page 77 de ce numéro.

ASSURANCE de ne manquer aucun numéro

RECEVOIR un cadeau* !

* Pour un abonnement de 24 numéros uniquement (délai de livraison : 4 semaines environ). ** Réservé aux abonnés 12 et 24 numéros.

OUI, Je m'abonne à ELECTRONIQUE ET LOISIRS magazine — LE MENSUEL DE L'ÉLECTRONIQUE POUR TOUS

A PARTIR DU N° 76 ou supérieur

E075

Ci-joint mon règlement de _____ € correspondant à l'abonnement de mon choix.

Adresser mon abonnement à : Nom _____ Prénom _____

Adresse _____

Code postal _____ Ville _____

Tél. _____ e-mail _____

❑ chèque bancaire ❑ chèque postal ❑ mandat

❑ Je désire payer avec une carte bancaire
Mastercard – Eurocard – Visa

Date d'expiration :

Cryptogramme visuel :
(3 derniers chiffres du n° au dos de la carte)

*Date, le*_____

Signature obligatoire ▷

Avec votre carte bancaire, vous pouvez vous abonner par téléphone.

TARIFS CEE/EUROPE

❑ **12 numéros** **49€,00**

TARIFS FRANCE

❑ **6 numéros** **22€,00**
au lieu de 27,00 € en kiosque, soit **5,00 € d'économie**

❑ **12 numéros** **41€,00**
au lieu de 54,00 € en kiosque, soit **13,00 € d'économie**

❑ **24 numéros** **79€,00**
au lieu de 108,00 € en kiosque, soit **29,00 € d'économie**

Pour un abonnement 24 numéros, cochez la case du cadeau désiré.

DOM-TOM/HORS CEE OU EUROPE : NOUS CONSULTER

1 CADEAU au choix parmi les 5

POUR UN ABONNEMENT DE 24 numéros

Gratuit :

❑ **Un money-tester**
❑ **Une radio FM / lampe**
❑ **Un testeur de tension**
❑ **Un réveil à quartz**
❑ **Une revue supplémentaire**



Avec 4,00 € uniquement en timbres :

❑ **Un alcootest électronique**

délai de livraison : 4 semaines dans la limite des stocks disponibles

Photos non contractuelles

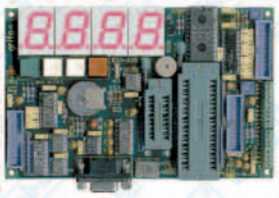
POUR TOUT CHANGEMENT D'ADRESSE, N'OUBLIEZ PAS DE NOUS INDIQUER VOTRE NUMÉRO D'ABONNÉ (INSCRIT SUR L'EMBALLAGE)

Bulletin à retourner à : JMJ – Abo. ELM

B.P. 20025 - 13720 LA BOUILLADISSE – Tél. 0820 820 534 – Fax 0820 820 722

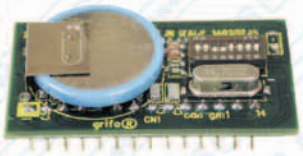
Pour le contrôle et l'automatisation industrielle, une vaste gamme parmi les centaines de cartes professionnelles abaco®

K51 AVR



La carte **K51-AVR** permet d'effectuer une expérimentation complète aussi bien des différents dispositifs pilotables en **I²C-BUS** que des possibilités offertes par les **CPU** de la famille **8051** et **AVR**, surtout accouplés au compilateur **BASCOM**. **Programmeur ISP** incorporé. De très nombreux exemples et des fiches techniques disponibles sur notre site.De nombreux exemples et data-sheet disponibles sur notre site.

CAN GM1



CAN Mini-Module de **28** broches basé sur le **CPU Atmel T89C51CC01** avec **32K FLASH** ; 256 Byte RAM; 1K ERAM; 2K FLASH for Bootloader; 2K EEPROM; 3 Timer-counters et 5 sections de Timer-Counter à haute fonctionnalité (PWM, watch dog, comparaison) ; RTC + 240 Octets RAM, tamponnés par batterie au Lithium ; I2C BUS ; 17 lignes d' E/S TTL ; 8 A/N 10 bits : RS 232 ou TTL ; **CAN** ; 2 DELs de fonctionnement ; Commutateur DIP de configuration ; etc.

GPC® x168



Contrôleur dans la version à Relay comme R168 ou bien à Transistors comme T168. Ils font partie de la **M Type** et comprennent un conteneur pour barre à Omega. 16 entrées optoisolées: 8 Darlington optoisolés de sortie de 3A ou bien Relay de 5A; 4 A/D et 1 D/A converter de 8 bits; ligne sérielle en RS 232, RS 422, RS 485 ou Current Loop; horloge avec batterie au Lithium et RAM tamponnée; E² sérielle; alimentateur switching incorporé; CPU 89C x 51 avec 32K RAM et jusqu'à 64K de FLASH. Opter pour plusieurs tools/instruments de développement du software tels que **BASCOM 8051**, **Ladder-Work**, etc. représente un choix optimal. Disponible également avec un programme de Télécontrôle par l'intermédiaire de **ALB**; on le gère directement à partir de la ligne sérielle du PC. Il contient de nombreux exemples.



GPC® 552

General Purpose Controller 80C552

Aucun système de développement extérieur avec **FM052** on peut de programmer la **FLASH** avec le programme utilisateur. **80C552** de 22MHz ou de 30MHz n'est nécessaire. De très nombreux langages de programmation sont disponibles tels que **BASCOM**, **C**, **BASIC**, **BXC51**, etc. Il est en mesure de piloter directement le Display LCD ou le clavier. Alimentateur incorporé et magasin barre à Omega. 32K RAM; 32K EPROM; socle pour 32K RAM, EPROM ou EEPROM , 44 lignes de I/O TTL; 8 lignes de A/D converter de 10 bits; 2PWM; Counter et Timer; Buzzer; 2 lignes série en RS 232, RS 422, RS 485, Current Loop; Watch-Dog; etc. Il programme directement l'EEPROM de bord avec le programme de l'utilisateur.



QTP 12

Quick Terminal Panel 12 touches

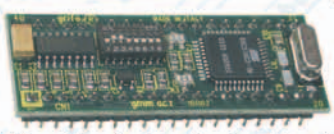
Tableau de commande de l'opérateur, à faible coût, avec boîtier standard **DIN** de **72x144 mm**. Disponible avec écran **LCD Rétroéclairé** ou **Fluorescent** aux formats **2x20** caractères ou Fluorescent **Graphique 140x16** pixels ; Clavier à 12 touches ; communication type RS 232, RS 422, RS 485 ou par Boucle de Courant ; ligne **CAN** ; Vibreur ; E2 interne en mesure de contenir configurations et messages.

GMB HR168



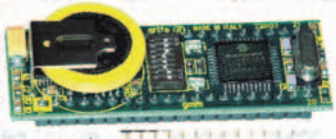
La **GMB HR168** est fondamentalement un module à Barre DIN en mesure d'accueillir une CPU **grifo® Mini-Module** du type GMM à **40** broches. Elle dispose de 16 entrées Galvaniquement isolées pour les signaux **NPN** ou **PNP**; 8 Relais de 5 A; ligne RS 232, RS 422, RS 485 ou Boucle de Courant; diverses lignes TTL et un alimentateur stabilisé.

GMM AC2



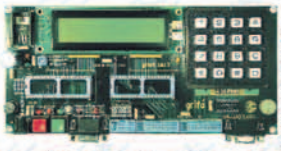
grifo® Mini-Module de **40** broches basée sur la **CPU Atmel T89C51AC2** avec **32K FLASH** ; 256 Bytes RAM ; 1K ERAM ; 2K FLASH pour Programme de lancement ; 2K EEPROM ; 3 Temporisateurs Compteurs et 5 sections de Temporisateur Compteur à haute fonctionnalité (PWM, chien de garde, comparaison) ; 32 lignes d'E/S TTL ; 8 A/N 10 bits ; RS 232 ou TTL ; 2 LEDs d'état ; Commutateur DIP de configuration ; etc.

GMM 4620



grifo® Mini-Module de **40** broches basée sur la **CPU Microchip PIC 18F4620** avec **64K FLASH**; 4K RAM; 1K EEPROM; 3 Timer-counters et 2 sections de Timer-Counter à haute fonctionnalité (PWM, watch dog, comparaison); RTC + 240 Octets RAM, tamponnés par batterie au Lithium; I2C BUS; 33 lignes d' E/S TTL; 13 A/N 10 bits; RS 232 ou TTL; 2 DELs de fonctionnement; Commutateur DIP de configuration; etc.

GMM TST2



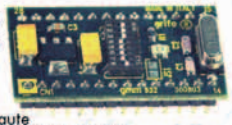
Carte à faible coût pour l'évaluation et l'expérimentation **grifo® Mini-Module** de **28** et de **40** broches type GMM 5115, GMM AC2, GMM 932, GMM AM08, GMM AM32, etc. Elle est dotée de connecteurs rectangulaires **D9** pour la connexion à la ligne sérielle en **RS 232**; connecteurs **10** broches pour la connexion à la **AVR ISP**; clavier à 16 touches; écran LCD rétroéclairé, de 20 caractères pour 2 lignes; Buzzer; connecteurs et sections d'alimentation; touches et LED pour la gestion des E/S numériques ; etc.

GMB HR84



La **GMB HR84** est fondamentalement un module à Barre DIN en mesure d'accueillir une **CPU grifo® Mini-Module** du type **CAN** ou **GMM** à **28** broches. Elle dispose de 8 entrées Galvaniquement isolées pour les signaux **NPN** ou **PNP** ; 4 Relais de 5 A ; ligne RS 232, RS 422, RS 485 ou Boucle de Courant ; ligne **CAN** ; diverses lignes TTL et un alimentateur stabilisé.

GMM 932



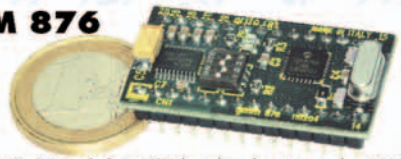
grifo® Mini-Module à **28** broches basée sur la **CPU Philips P89LPC932** avec **8K FLASH** ; 768 Bytes RAM ; 512 Bytes EEPROM ; 3 Temporisateurs Compteurs et 2 sections de Temporisateur Compteur à haute fonctionnalité (PWM, comparaison) ; 2 Comparateurs ; I2C BUS ; 23 lignes d'E/S TTL ; RS 232 ou TTL ; 1 LED d'état ; etc. Alimentation de 2,4V à 5,5V.

GPC® 554



Carte de la **4 Type** de 5x10 cm. Aucun système de développement extérieur n'est nécessaire et avec **FM052** on peut programmer la FLASH avec le programme utilisateur; 80C552 de 22 MHz avec 90K 32K-RAM; sockets pour 32K EPROM et 32K EEPROM, RAM, EPROM ou FLASH; E² en série; connecteur pour batterie au lithium extérieure; 16 lignes de I/O; 6/8 lignes de A/D de 10 bits; 1/2 lignes en série; une RS 232, Watch-dog; timer; counter; connecteur d'expansion pour **Abaco®** I/O BUS, etc. De nombreux tools de développement du logiciel avec des langages de haut niveau comme **BASCOM**, **Assembler**, **BXC-51**, **Compilateur C**, **MCS52**, **SoftICE**, **NoICE**, etc.

GMM 876



grifo® Mini-Module à 28 broches basée sur la **CPU Microchip PIC 16F876A** avec **14,3K FLASH**; 368 Bytes RAM; 256 Bytes EEPROM; 2 Temporisateurs Compteurs et 2 sections de Temporisateur Compteur à haute fonctionnalité (PWM, comparaison); 2 Comparateurs; 5 A/D; I2C BUS; Master/Slave SPI; 22 lignes d'E/S TTL; RS 232 ou TTL; 1 LED d'état; etc.

GMM AM08



grifo® Mini-Module de **28** broches basée sur la **CPU AVR Atmel ATmega 8** avec **8K FLASH** ; 1K RAM ; 512 Bytes EEPROM ; 3 Temporisateurs Compteurs, 3 PWM ; 8 A/N 10/8 bits ; SPI ; Chien de garde Temporisateur ; 23 lignes d'E/S TTL ; RS 232 ou TTL ; 1 LED d'état ; Commutateur DIP de configuration ; etc. Alimentation de 2,7V à 5,5V.

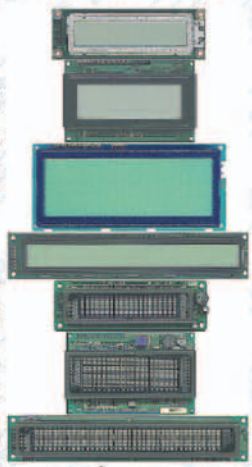
QTP 03

Terminal 3 Touches

Finalement, vous pouvez également équiper vos applications les plus économiques d'un Tableau Commande Opérateur complet. 3 touches; Buzzer; ligne sérielle réglable au niveau TTL ou RS232; E² pouvant contenir jusqu'à 100 messages; etc.

QTP 4x6

Terminal 4x6 Touches



Si vous avez besoin de plus de touches, ou de les connecter sur le réseau, choisissez la version QTP 4x6 qui gère jusqu'à 24 Touches. Quoique ressemblant à des afficheurs série ordinaires, ce sont des Terminaux Vidéo complets. Disponible avec écran **ACL** à **illumination postérieure** ou **Fluorescents** dans les formats 2x20; 4x20 ou 2x40 caractères; clavier 4x6; Buzzer; ligne sérielle réglable RS232; RS422; RS485; Current Loop; E² pouvant contenir jusqu'à 100 message; etc.

GMM PIC-PR



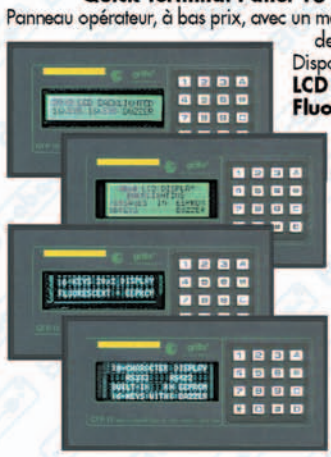
grifo® Mini Module PIC-Programmer

Carte à bas prix dotée de socle **ZIF** pour programmer les **grifo® Mini-Module** de **28** et **40** broches type GMM 876, GMM 4620, CAN PIC ect. La carte est dotée aussi de: connecteur ligne **RS232**; connecteur **D9** pour la connection à la **RJ12** pour **MPLAB**; connecteur à **10** broches pour la connection au Programmateur **MP PIK+**; connecteur pour la section alimentateur; 2 LEDs; ect..

QTP 16

Quick Terminal Panel 16 touches

Panneau opérateur, à bas prix, avec un magasin standard de 96x192 mm. Disponible avec display **LCD Rètroèclairé** ou **Fluorescent** dans les formats **2x20** ou **4x20** caractères; clavier à **16** touches; communication en RS 232, RS 422, RS 485, ou Current Loop; Buzzer; E² capable de contenir jusqu'à 100 messages; 4 entrées optocouplées, que l'on peut acquérir à travers la ligne série et susceptibles de représenter de façon autonome 16 messages différents. même temps jusqu'à 8 dispositifs.



40016 San Giorgio di Piano (BO) - Via dell'Artigiano, 8/6
Tel. +39 051 892052 (4 linee r.a.) - Fax +39 051 893661
Web au site: http://www.grifo.it - http://www.grifo.com

grifo®
ITALIAN TECHNOLOGY

LEXTRONIC

LEXTRONIC
36/40 Rue du Gal de Gaulle
94510 La Queue en Brie
Tel: 01.45.76.83.88 - Fax: 01.45.76.83.88
E-mail: lextronic@lextronic.fr - http://www.lextronic.fr

GPC® abaco® grifo® sont des marques enregistrées de la société grifo®

ESPACE COMPOSANT ELECTRONIQUE

66 Rue de Montreuil 75011 Paris, métro Nation ou Boulet de Montreuil.
Tel : 01 43 72 30 64 / Fax : 01 43 72 30 67 / Mail : ece@ibcfrance.fr
Ouvert le lundi de 10 h à 19 h et du mardi au samedi de 9 h 30 à 19 h

www.ibcfrance.fr Commande sécurisée.

PLUS DE 30.000 REFERENCES EN STOCK

HOT LINE PRIORITAIRE pour toutes vos questions techniques : 08 92 70 50 55 (0.306 € / min)

N°Indigo 0 825 82 59 04

Une gamme très importante de kits et modules en stock

Un extrait de la gamme OFFICE DU KIT

CH 12 Ioniseur électronique	46.52€
CH 14 Detartreur électronique	33,37€
CH 30 Horloge murale parlante	93.17€
CH 31 Truqueur de voix + micro	37.20€
CH 32 Horloge analogique à leds	71.64€
CH 38 Sifflet de dressage pour chien	33,43€
CH 44 Thermomètre mural à LED	44.13€
CH 52 Anamomètre digital	59.20€
CH 78 Alimentation HT pour clôture	35.82€
CH 83 Chasse oiseaux électronique	65.66€
CH 96 Fréquencemètre 27 Mhz CB	65.66€
PL 11 Gradateur de lumière	7.77€
PL 22 Télécommande secteur	27.45€

Un extrait de la gamme VELLEMAN

MK 100 Sapin de noel avec led clignotantes	8.95€
MK 101 Feux coeur à led	13.75€
MK 103 Modulateur à leds	7.50€
MK 104 Coquet électronique	10.50€
MK 105 Générateur de signaux	8.95€
MK 106 Métronome	11.95€
MK 108 Détecteur d'eau	7.50€
MK 109 De électronique	8.95€
MK 115 Sonomètre de poche à leds	7.50€
MK 119 Roulette à led	19.95€
MK 124 Mini journal lumineux	18.25€
MK 126 Simulateur d'alarme de voiture	5.95€
MK 140 Karaoké	14.95€

Un extrait de la gamme ERMES

ER 103 Serrure électr. à transpondeur	42.00€
ER 107 Enregistreur vocal	29.00€
ER 112 Programmateur journalier 5 sorties	35.50€
ER 113 Ampli Mono 1 x 70w	21.00€
ER 114 Mini labo de sust en malette	135.00€
ER 116 Led folie (jeu de 121 leds)	62.00€
ER 117 Mini étoile en leds CMS	24.50€
ER 118 Etoile lumineuse 70 cm	45.50€
ER 119 Enregistreur de températures PC	48.00€
ER 125 Chenillard 8 canaux programmable	49.00€
ER 213 Ampli stéréo 2 x 30 w	25.00€
ER 300 Chargeur pour torche ER301	30.00€
ER 301 Torche à leds étanche rechargeable	45.00€

Les démodulateurs

CI-20E	230.00€
DELTA FTA	59.00€
DM-500S ....dreambox	215.00€
DM-7020 .....disponible tel	476.00€
CLAVIER SANS FIL DREAMBOX	79.00€
DIGIT CIVA.....1 pcmcia + 1 lecteur	169.00€
ICE MM1100	139.00€
K200 KAON	119.00€
KSC520	239.00€
MIRASAT 4000	57.00€
NEOTION3000.....de retour	219.00€
NEOTION501.....avec lecteur	159.00€
REX IV Super emu incorporé..2 + 2	177.00€
SIMBA 202S.....2 lecteurs	229.00€
SKYSTAR USB	115.00€
SKYSTAR 2 PCI	59.50€
SKYSTAR 1 CI.....1 PCMCIA	139.00€
@sat FX-7220	209.00€
@sat FX-6910	184.00€
@sat FX-6915	195.00€
@sat FX-5015	179.00€
@sat FX-5010	149.50€

DIGISAT PRO ACCUS

DIGISAT Pro Accu est contrôlé par microprocesseur ce qui le rend très fiable et précis.Cet instrument est unique car il peut mesurer le signal satellite à partir de deux LNB en même temps L'intensité de réception est représentée graphiquement sur 'afficheur LCD sous forme d'échelles graduées et de nombres de 0 à 99.9.. DIGITAL Pro Accu est alimenté soit par une batterie rechargeable intégrée soit à partir d'un récepteur (à travers un coaxial). 118.00€

Le programmateur de cartes a puces infinity unlimited

Duplicateur de sim gsm inclus, programme ces differentes cartes, Wafercard,Goldcard, Silvercard , Greencard, Greencard2,Bluecard,EmeraldCard, Singlepic, Funcard Funcard2, PrussianCard/Funcard3, PrussianCard2/Funcard4 ,PrussianCard3/Funcards5, PrussianCard4/Funcard6 , PrussianCard5/Funcard6 , JupiterCard JupiterCard2 , FunCard ATmega161, FunCard ATmega163, FunCard ATmega8515/Funkey2, Blackcard, GSM/SIM card, Megapic, Titaniumcard, Basiccard 4.5D, Dragonloader card, Knot card, OPOS card,Toute autre carte compatible Phoenix /Smartmouse à 3.58,,3.68 et 6.00 Mhz

70.50 €

neotion box 501

neotionbox 501 = 159.00 €

La TNT disponible

Télévision Numérique Terrestre

Une gamme tres importante disponible

Mirasat 2006TWM ou Ultima twm	69.00€
Airstar 2 TV....format PCI	69.00€
Ultima T ou Mirasat 2005T	74.00€
Digitmod T1....technisat	99.00€
Televes dir 7287	115.00€
Moditel 1....version PCMCIA	115.00€
Digipal 2....technisat	119.00€
Humax F3fox	139.00€

DIGITAIR

L'intensité de réception est représentée graphiquement sur 'afficheur LCD sous forme d'échelles .
Mesure de 47 mhz à 862 Mhz
Accus integré avec chargeur d'accus livré
beeper 190.00€

Pour regler vos LNB

SATLOOK

DIGITAL NIT..	870.00€
MICRO + =....	520.00€
MARK III =...	399.00€
MARK IV =...	830.00€

Les programmateurs

pour carte

Dynamite=	32.95€
Infinity usb =	28.50€
Infinity phoenix =	41.50€
Mastera v =	75.00€
Mastercrd2=	115.00€
Millenium4+=	19.50€
Mini apollo=	8.00€
Multipro rs232=	34.00€
Multipro usb=	25.00€

pour carte

Cas interface 2

Programme les magic modules et les clones (Matrix -axas - etc) mais aussi d'autre cam de la famille zetacam .Possède en plus un JTAG interface pour la DM7000

Nouveau CAS INTERFACE 3	87.50€
cas interface 2 USB =	48.50€
ADD-ON =	34.00€
Cas interface +port parallele =	26.00€

Nouveauté

Permet d'adapter 3 démodulateurs sur un seul abonnement sans fils

159.00€

Les PCMCIA

Matrix revolution	= 48.30€
Matrix reborn	= 58.30€
Réality cam	= 78.00€
Xcam =	=85.00€
viaccess rouge	=59.00€
freextv jaune	=64.00€
skycrypt	= 149.00€
zetacam blue	=62.00€
dragon twin	=97.00€
dragon twin+loader	= 113.95€

Les cartes a puces

Wafer gold.....16f84 et24lc16	2.30 €
Silver.....16f876/7 et 24lc64	5.40 €
Silver. green..16f876/7 et 24lc128	6.15 €
Atméga.....Atmega163 et 24 lc 256	15.00 €
FUN.....AT90S8515 + 24LC64	6.45 €
FUN 4.....AT90S8515 + 24LC 256	6.50 €
FUN 5.....AT90S8515 + 24LC 512	6.40 €
FUN 6.....AT90S8515 + 24LC 1024	7.40 €
FUN 7.....AT90S8515 + 2*24LC 1024	10.25 €
TITANIUM 2.....Nouvelle titanium	47.50 €
FUNUSB + adaptateur = fun6 en usb	65.00 €
KNOTCARD	19.00 €
KNOTCARD II	38.50 €
TITANCARD2	57.00 €
PLATINUM	95.00 €
OPOS..Equivallent titanium ou knotcard	85.00 €
SCT SATISFACTION.....public averti	139.00 €
redlight 5 chaines 6 mois	89.00€
1 ans .....129.00€	
INXWORLD.....public averti	59.00 €
DRAGON LOAD	19.00 €

Les prix sont donnés à titre indicatif et peuvent être modifiés sans préavis, vérifiez les sur notre site internet pour les ventes par correspondance. Tous nos prix sont TTC. Les produits actifs ne sont ni repris ni échangés. Forfait de port 6.10€ (gratuit à partir de 229€.) sauf colis de plus de 1.5kg, port = 15€. Photo non contractuelles